

UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE INGENIERIA
CARRERA DE INGENIERIA INDUSTRIAL



OPTIMIZAR LA GESTION DE LA PRODUCCION EN LA EMPRESA
“ESPAÑOLA” MEDIANTE LA IMPLEMENTACION DE “SIS-K”

Proyecto de grado para optar al Título de Licenciatura en Ingeniería Industrial

POSTULANTES:

Victor Joaquin Vargas Arce
Dennis Marvin Blanco Aguilar

TUTOR:

Msc. Ing. Oswaldo Terán Modregón

La Paz - Bolivia
julio, 2014

**UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE INGENIERÍA
CARRERA DE INGENIERÍA INDUSTRIAL**

Proyecto de grado:

**OPTIMIZAR LA GESTIÓN DE LA PRODUCCIÓN EN LA EMPRESA
“ESPAÑOLA” MEDIANTE LA IMPLEMENTACIÓN DE “SIS-K”**

Presentada por: Univ. Victor Joaquin Vargas Arce

Univ. Dennis Marvin Blanco Aguilar

Para optar el grado académico de Licenciatura en Ingeniería Industrial

Nota Numeral.....

Nota Literal.....

Ha sido.....

Director de la carrera de Ingeniería Industrial Ing. Msc. Oswaldo Terán Modregon

Tutor: Ing. Msc. Oswaldo Terán Modregon

Tribunal: Ing. Mónica Lino Humerez

Tribunal: Ing. Jorge Velasco Tudela

Tribunal: Ing. José Freddy Gutiérrez Barea

Tribunal: Ing. Javier Cordero Torrez

CONTENIDO

RESUMEN	1
ABSTRACT	3
1 CAPÍTULO 1. PLANTEAMIENTO DEL PROBLEMA	5
1.1 INTRODUCCIÓN	5
1.1.1 <i>Definiciones</i>	5
1.2 BASES DE LA GESTIÓN	5
1.3 PROBLEMÁTICA	7
1.4 FASES DEL ANÁLISIS DE PARETO	9
1.4.1 <i>Lluvia de ideas</i>	9
1.4.2 <i>Diagrama del pescado</i>	11
1.4.3	12
1.4.4 <i>Análisis de Pareto</i>	12
1.4.5 <i>Diagrama de Pareto</i>	13
1.4.6 <i>Soluciones tentativas</i>	13
1.5 OBJETIVOS	14
1.5.1 <i>Objetivo general</i>	14
1.5.2 <i>Objetivos específicos</i>	14
1.5.3 <i>Metodología</i>	14
1.6 CRONOGRAMA DE TRABAJO.....	15
1.7 PROCEDIMIENTOS AUXILIARES	16
1.7.1 <i>Procedimiento para citas y entrada bibliográfica</i>	16
1.7.2 <i>Procedimiento para elaboración de diagramas</i>	17
1.7.3 <i>Procedimiento para la programación del sistema informático</i>	18
2 CAPÍTULO 2. LA EMPRESA	20
2.1 HISTORIA	20
2.2 MISIÓN	20
2.3 VISIÓN	20
2.4 DEFINICIÓN DE LA EMPRESA	21
2.5 DESCRIPCIÓN PROCESO PRODUCTIVO.....	21
2.5.1 <i>Recepción, inspección y almacenamiento de materia prima</i>	21
2.5.2 <i>Selección y troceado</i>	21
2.5.3 <i>Molido y dosificado de especias</i>	21
2.5.4 <i>Mezclado</i>	21
2.5.5 <i>Embutido</i>	22
2.5.6 <i>Secado y ahumado</i>	22
2.5.7 <i>Cocido y enfriado</i>	22
2.5.8 <i>Empaquetado</i>	22
2.5.9 <i>Almacenado</i>	22
2.6 MAQUINARIA.....	23
2.7 PRODUCTOS	25
2.7.1 <i>Embutidos crudos</i>	25
2.7.2 <i>Embutidos escaldados</i>	25
2.7.3 <i>Embutidos cocidos</i>	25
2.7.4 <i>Carnes curadas</i>	25
2.7.5 <i>Productos</i>	25
2.8 INSUMOS.....	26
2.8.1 <i>Materia prima o materiales de elaboración</i>	27

2.8.2	<i>Tripas</i>	27
2.8.3	<i>Sustancias curantes</i>	27
2.8.4	<i>Nitratos y nitritos</i>	27
2.8.5	<i>Aglutinantes y ablandadores</i>	27
2.8.6	<i>Carrageninas</i>	28
2.8.7	<i>Conservantes</i>	28
2.9	ESTRUCTURA ORGANIZACIONAL DE LA EMPRESA.....	28
3	CAPÍTULO 3. SISTEMA DE INFORMACIÓN.....	29
3.1	INTRODUCCIÓN.....	29
3.1.1	<i>Definiciones</i>	29
3.2	BASES PARA LA CREACIÓN DE LA BASE DE DATOS.....	29
3.2.1	<i>La información</i>	29
3.2.2	<i>Base de datos</i>	30
3.2.3	<i>Estructura de una base de datos</i>	31
3.2.4	<i>Ventajas y desventajas de utilizar bases de datos</i>	31
3.2.5	<i>Gestor de base de datos</i>	31
3.2.6	<i>Modelado de datos</i>	33
3.3	FASES PARA LA CREACIÓN DE BASE DE DATOS.....	34
3.3.1	<i>Definición</i>	34
3.3.2	<i>Alcance</i>	34
3.4	TABLAS.....	34
3.4.1	<i>Formulación de las tablas</i>	35
3.5	RELACIONES.....	39
3.5.1	<i>Definiciones</i>	39
3.5.2	<i>Establecer relaciones</i>	39
3.6	CONSULTAS.....	43
3.6.1	<i>Tipos de consulta</i>	43
3.6.2	<i>Como efectuar una consulta en MSOL Server</i>	43
3.6.3	<i>Establecer consultas</i>	44
3.7	ACTUALIZACIÓN DE DATOS.....	45
3.8	MANUAL SQL SERVER 2012.....	45
3.8.1	<i>Introducción</i>	45
3.8.2	<i>Componentes del SQL</i>	45
3.8.3	<i>Componentes del SQL (II)</i>	45
3.8.4	<i>Orden de ejecución de los comandos</i>	46
3.8.5	<i>Consultas de Selección</i>	46
3.8.6	<i>Ordenar los registros</i>	47
3.8.7	<i>Consultas de Acción</i>	48
3.8.8	<i>Insert into</i>	49
3.8.9	<i>Update</i>	49
3.8.10	<i>Referencias cruzadas</i>	50
3.8.11	<i>Criterios de selección</i>	51
3.8.12	<i>El Operador Like</i>	53
3.8.13	<i>Agrupamiento de registros (Count - Max, min - StDev, StDevP)</i>	54
3.8.14	<i>Agrupamiento de registros (Sum - Var, VarP - Compute)</i>	56
3.8.15	<i>COMPUTE de SQL-SERVER</i>	58
3.8.16	<i>Subconsultas</i>	58
3.8.17	<i>Creación de Tablas Nuevas</i>	60
3.8.18	<i>Modificar el Diseño de una Tabla</i>	62
3.8.19	<i>Búsqueda de Registros Duplicados</i>	63
3.8.20	<i>Búsqueda de Registros no Relacionados</i>	64

4	CAPÍTULO 4. DISEÑO DEL SISTEMA INFORMÁTICO SIS-K.....	65
4.1	INTRODUCCIÓN.....	65
4.1.1	<i>Definiciones.....</i>	65
4.2	BASES PARA EL DISEÑO DEL SISTEMA SIS-K.....	65
4.2.1	<i>Diseño de sistemas.....</i>	66
4.2.2	<i>Hardware.....</i>	67
4.2.3	<i>Software.....</i>	67
4.2.4	<i>Ingeniería del Software.....</i>	68
4.2.5	<i>Selección de los componentes del sistema.....</i>	69
4.3	FASES DEL SISTEMA SIS-K.....	72
4.3.1	<i>Administración.....</i>	72
4.3.2	<i>Provisión.....</i>	73
4.3.3	<i>Producción.....</i>	73
4.3.4	<i>Ventas.....</i>	74
4.3.5	<i>Financiera.....</i>	74
4.4	DIAGRAMA UML DE TODO EL SISTEMA.....	75
5	CAPÍTULO 5. DISEÑO DEL MÓDULO ADMINISTRACIÓN DE RECURSOS HUMANOS	76
5.1	INTRODUCCIÓN.....	76
5.2	BASES PARA LA ADMINISTRACIÓN DE RECURSOS HUMANOS.....	77
5.2.1	<i>Manual de funciones.....</i>	77
5.2.2	<i>Conformación Del Manual.....</i>	78
5.3	FASES DEL SISTEMA DE ADMINISTRACIÓN DE RECURSOS HUMANOS.....	81
5.3.1	<i>Gestión de empleados.....</i>	82
5.3.2	<i>Manual de funciones.....</i>	84
5.3.3	<i>Boletas de pago.....</i>	85
5.3.4	<i>Planilla de sueldos y salarios.....</i>	86
6	CAPÍTULO 6. DISEÑO DEL MÓDULO DE PROVISIÓN.....	87
6.1	INTRODUCCIÓN.....	87
6.2	BASES DE LA GESTIÓN DE LA PROVISIÓN.....	87
6.3	FASES DE LA GESTIÓN DE LA PROVISIÓN.....	88
6.3.1	<i>Gestión de Proveedores.....</i>	89
6.3.2	<i>Comprobante de compra.....</i>	90
7	CAPÍTULO 7. DISEÑO DEL MÓDULO DE VENTAS.....	92
7.1	INTRODUCCIÓN.....	92
7.2	BASES PARA LA GESTIÓN DE VENTAS.....	92
7.2.1	<i>Gestión de ventas.....</i>	92
7.2.2	<i>Elementos fundamentales de la gestión de ventas.....</i>	92
7.2.3	<i>El cliente.....</i>	93
7.2.4	<i>Proceso de ventas.....</i>	94
7.2.5	<i>Fuerza de ventas.....</i>	95
7.2.6	<i>Metodología para la gestión del equipo de ventas.....</i>	95
7.2.7	<i>Estrategias para la gestión de ventas.....</i>	96
7.2.8	<i>Sistema de ventas.....</i>	97
7.2.9	<i>El merchandising.....</i>	97
7.3	FASES DE LA GESTIÓN DE VENTAS.....	99
7.3.1	<i>Gestión de clientes.....</i>	99
7.3.2	<i>Registro de venta.....</i>	100
7.3.3	<i>Factura.....</i>	100

8	CAPÍTULO 8. DISEÑO DEL MÓDULO FINANZAS	102
8.1	INTRODUCCIÓN.....	102
8.2	BASES PARA LA GESTIÓN FINANCIERA	102
8.2.1	<i>Funciones</i>	<i>103</i>
8.2.2	<i>Herramientas.....</i>	<i>103</i>
8.2.3	<i>Técnicas del análisis financiero</i>	<i>104</i>
8.2.4	<i>Razones financieras.....</i>	<i>104</i>
8.3	FASES DE LA GESTIÓN FINANCIERA.....	106
8.3.1	<i>Registros contables.....</i>	<i>106</i>
8.3.2	<i>Gestor de cuentas.....</i>	<i>106</i>
8.3.3	<i>Índices financieros.....</i>	<i>108</i>
8.3.4	<i>Balance general.....</i>	<i>108</i>
8.3.5	<i>Estado de resultados.....</i>	<i>109</i>
9	CAPÍTULO 9. DISEÑO DEL MÓDULO PRODUCCIÓN	111
9.1	INTRODUCCIÓN.....	111
9.2	BASES DE LA GESTIÓN DE LA PRODUCCIÓN	111
9.2.1	<i>Sistema de agotamiento de depósito.....</i>	<i>111</i>
9.2.2	<i>Sistema empujar o MRP</i>	<i>125</i>
9.2.3	<i>Sistema Jalar o Justo a tiempo JIT.....</i>	<i>129</i>
9.2.4	<i>Enfoque a cuellos de botella o teoría de las restricciones</i>	<i>138</i>
9.3	FASES DE LA GESTIÓN DE LA PRODUCCIÓN	139
9.3.1	<i>Inventario de insumos.....</i>	<i>140</i>
9.3.2	<i>Inventario de productos.....</i>	<i>141</i>
9.3.3	<i>Programa maestro de producción.....</i>	<i>142</i>
10	CAPÍTULO 10. INGENIERÍA DEL PROYECTO	144
10.1	INTRODUCCIÓN.....	144
10.1.1	<i>Definiciones.....</i>	<i>144</i>
10.2	BASES PARA LA INGENIERÍA DEL PROYECTO	144
10.2.1	<i>Objetivo.....</i>	<i>144</i>
10.2.2	<i>Modelo de ingeniería del proyecto.....</i>	<i>144</i>
10.3	FASES DE LA MEDICIÓN DEL TRABAJO.....	145
10.3.1	<i>Definiciones.....</i>	<i>145</i>
10.4	FASES DE LA MEDICIÓN DE TIEMPOS IMPRODUCTIVOS	145
10.4.1	<i>Medición de tiempos improductivos.....</i>	<i>146</i>
10.5	FASES DE LA MEDICIÓN DE TIEMPOS DE OPERACIÓN	160
10.5.1	<i>Medición del tiempo de los elementos de las tareas</i>	<i>161</i>
10.5.2	<i>Medición de los tiempos de elaboración de los productos.....</i>	<i>177</i>
10.5.3	<i>Medición de los rendimientos.....</i>	<i>188</i>
10.6	PRONÓSTICO DE LA DEMANDA.....	189
10.6.1	<i>Bases para el pronóstico de la demanda.....</i>	<i>189</i>
10.6.2	<i>Fases del pronóstico de la demanda</i>	<i>189</i>
10.6.3	<i>Definición de la variable endógena</i>	<i>190</i>
10.6.4	<i>Medición de los valores históricos.....</i>	<i>190</i>
10.6.5	<i>Definición de las causas.....</i>	<i>191</i>
10.6.6	<i>Modelo causal.....</i>	<i>191</i>
10.6.7	<i>Modelo no causal</i>	<i>193</i>

11	CAPÍTULO 11. DISEÑO DE MÉTODOS	202
11.1	INTRODUCCIÓN.....	202
11.2	FASES DE LA PROGRAMACIÓN LINEAL	202
11.2.1	<i>Definición.....</i>	202
11.2.2	<i>Alcance.....</i>	202
11.2.3	<i>Estructura del sistema de información.....</i>	203
11.3	MODELACIÓN	203
11.3.1	<i>Definición de la función objetivo</i>	204
11.3.2	<i>Definición de las restricciones.....</i>	204
11.3.3	<i>Validación del modelo.....</i>	207
11.3.4	<i>Solución del modelo</i>	207
12	CAPÍTULO 12. DISEÑO DE LA INTERFAZ	210
12.1	INTRODUCCIÓN.....	210
12.1.1	<i>Definiciones.....</i>	210
12.1.2	<i>Bases para el desarrollo de la aplicación.....</i>	210
12.1.3	<i>Entorno de desarrollo integrado.....</i>	210
12.1.4	<i>Lenguaje de programación</i>	210
12.1.5	<i>Editor de texto.....</i>	211
12.1.6	<i>Compilador</i>	211
12.1.7	<i>Interprete.....</i>	212
12.1.8	<i>Depurador.....</i>	213
12.1.9	<i>Cliente</i>	214
12.1.10	<i>Interfaz gráfica de usuario.....</i>	215
12.1.11	<i>Metodología para el diseño de interfaz del sistema.....</i>	215
12.2	CÓDIGO FUENTE	217
12.3	DISEÑO DE LA BASE DE DATOS.....	217
12.4	INTERFAZ PARA EL USUARIO.....	218
12.4.1	<i>Administración</i>	219
12.4.2	<i>Provisión</i>	222
12.4.3	<i>Producción.....</i>	222
12.4.4	<i>Ventas.....</i>	225
12.4.5	<i>Contabilidad.....</i>	226
12.5	SALIDAS DEL SISTEMA	229
12.6	SEGURIDAD DEL SISTEMA	232
13	CAPÍTULO 13. MANUAL DEL USUARIO.....	234
13.1	INGRESO AL SISTEMA.....	234
13.2	ENTORNO DEL SISTEMA	234
13.3	MÓDULO DE ADMINISTRACIÓN	235
13.3.1	<i>Gestión de empleados.....</i>	236
13.3.2	<i>Entorno del gestor de empleados.....</i>	237
13.3.3	<i>Crear nuevo registro</i>	237
13.3.4	<i>Modificar un registro</i>	238
13.3.5	<i>Eliminar un registro</i>	238
13.3.6	<i>Modo de búsqueda</i>	239
13.3.7	<i>Manual de funciones</i>	240
13.3.8	<i>Boletas de pago.....</i>	241
13.3.9	<i>Planilla de sueldos personal</i>	242
13.4	MÓDULO DE PROVISIÓN	242
13.4.1	<i>Gestión de proveedores.....</i>	243
13.4.2	<i>Entorno del administrador de proveedores.....</i>	243

13.4.3	<i>Modificar un registro</i>	244
13.4.4	<i>Eliminar un registro</i>	244
13.4.5	<i>Modo de búsqueda</i>	244
13.4.6	<i>Comprobante de compra</i>	245
13.5	MÓDULO DE PRODUCCIÓN	245
13.5.1	<i>Entorno de administrador de productos terminados</i>	246
13.5.2	<i>Crear nuevo registro</i>	246
13.5.3	<i>Modificar un registro</i>	246
13.5.4	<i>Eliminar un registro</i>	246
13.5.5	<i>Modo de búsqueda</i>	247
13.5.6	<i>Entorno administrador de insumos</i>	247
13.5.7	<i>Crear nuevo registro</i>	247
13.5.8	<i>Modificar un registro</i>	247
13.5.9	<i>Eliminar un registro</i>	248
13.5.10	<i>Modo de búsqueda</i>	248
13.5.11	<i>Programa maestro de producción</i>	248
13.6	MÓDULO DE VENTAS	250
13.6.1	<i>Entorno gestión de clientes</i>	251
13.6.2	<i>Crear nuevo registro</i>	251
13.6.3	<i>Modificar un registro</i>	251
13.6.4	<i>Eliminar un registro</i>	252
13.6.5	<i>Modo de búsqueda</i>	252
13.6.6	<i>Registro de ventas</i>	252
13.7	MODULO FINANCIERO	254
13.7.1	<i>Registros contables</i>	255
13.7.2	<i>Entorno del administrador de cuentas</i>	255
13.7.3	<i>Crear nuevo registro</i>	256
13.7.4	<i>Modificar un registro</i>	256
13.7.5	<i>Eliminar un registro</i>	256
13.7.6	<i>Modo de búsqueda</i>	256
13.7.7	<i>Indicadores financieros</i>	256
13.7.8	<i>Balance general</i>	257
13.7.9	<i>Estado de resultados</i>	258
14	CAPÍTULO 14. EVALUACIÓN ECONÓMICA Y FINANCIERA	260
14.1	DEFINICIÓN	260
14.1.1	<i>Bases para la evaluación financiera</i>	260
14.1.2	<i>Fases para la evaluación financiera</i>	260
14.2	BENEFICIOS DEL SISTEMA	260
14.3	COSTOS DE OPORTUNIDAD	261
14.3.1	<i>Acceso rápido a la información</i>	261
14.3.2	<i>Menor redundancia de la información</i>	261
14.3.3	<i>Mayor seguridad en los datos</i>	261
14.3.4	<i>Se tienen datos más documentados</i>	261
14.3.5	<i>Toma de decisión más eficiente</i>	261
14.3.6	<i>Menor espacio de almacenamiento</i>	262
14.3.7	<i>Mayor control en almacenes</i>	262
14.3.8	<i>Evita pérdida de tiempo en la recopilación de la información</i>	262
14.3.9	<i>Menor error en la planeación de la producción</i>	262
14.3.10	<i>Proporciona, facilita y ejecuta automáticamente procesos</i>	262
14.3.11	<i>Mayor capacidad de análisis con los nuevos reportes</i>	262

14.4	BENEFICIO TOTAL	262
14.5	COSTO DE IMPLEMENTAR EL SISTEMA.....	263
14.6	FLUJO DE FONDOS.....	263
14.6.1	<i>Tasa de oportunidad</i>	263
14.6.2	<i>Indicador de rentabilidad</i>	263
	CONCLUSIONES	265
	RECOMENDACIONES	267
	BIBLIOGRAFÍA	268



ÍNDICE DE FIGURAS

FIGURA 1-1 SIS-K y La española. Cronograma de trabajo, 2014.....	15
FIGURA 1-2 Conceptos de lenguaje VB.NET, 2014.....	19
FIGURA 2-1 La española. Organigrama de la empresa, 2014.....	28
FIGURA 3-1 Modelo de creación de base de datos, 2014	33
FIGURA 3-2 La española. Formulario para la formulación de tablas, 2014	36
FIGURA 3-3 La española. Formulario para el diseño de tablas, 2014	38
FIGURA 3-4 SIS-K DB. Diagrama de relación para el sistema de facturación, 2014.....	41
FIGURA 3-5 SIS-K DB. Diagrama de relación para el sistema de contabilidad, 2014	42
FIGURA 3-6 La española. Ejemplo de código y reporte, 2014	44
FIGURA 4-1 SIS-K. Sistema de información designado, 2014.....	65
FIGURA 4-2 SIS-K. Sistema técnico de la información designado, 2014	66
FIGURA 4-3 SIS-K. Diagrama de red, 2014.....	70
FIGURA 4-4 La española. Evaluación de experiencia en Windows, 2014	71
FIGURA 5-1 La española. Ejemplo de reporte para la gestión de empleados, 2014	83
FIGURA 5-2 La española. Ejemplo de reporte del manual de empleados, 2014	84
FIGURA 5-3 La española. Ejemplo de reporte del boleto de pago, 2014.....	85
FIGURA 5-4 La española. Ejemplo de reporte de planilla de pago, 2014.....	86
FIGURA 6-1 La española. Ejemplo de reporte para la gestión de proveedores, 2014.....	90
FIGURA 6-2 La española. Ejemplo de reporte para el registro de compra, 2014	91
FIGURA 7-1 La española. Ejemplo de registro para la gestión de clientes, 2014.....	101
FIGURA 8-1 La española. Ejemplo de reporte de cuentas, 2014	107
FIGURA 8-2 La española. Ejemplo de reporte de indicadores financieros, 2014.....	108
FIGURA 8-3 La española. Ejemplo de reporte de balance general, 2014	109
FIGURA 8-4 La española. Ejemplo de reporte de balance general, 2014	110
FIGURA 9-1 La española. Ejemplo de reporte de insumos, 2014	140
FIGURA 9-2 La española. Ejemplo de reporte de producto terminado, 2014.....	141
FIGURA 10-1 La española: Formulario de observación de estudio de tiempos.2014	175
FIGURA 10-2 La española: Modelo de Holt- Winters para la salchicha.2014	199
FIGURA 10-3 La española: Cuadro de dialogo SOLVER para la salchicha, 2014	200
FIGURA 12-1 La española: Tabla de empleados, 2014.....	217
FIGURA 12-2 La española: Desarrollo del módulo principal del sistema, 2014	218
FIGURA 12-3 La española: Modulo principal en depuración, 2014	219
FIGURA 12-4 La española: Gestión de empleados, modulo administración, 2014.....	220
FIGURA 12-5 La española: Manual de funciones, modulo administración, 2014.....	220
FIGURA 12-6 La española: Boletas de Pago, modulo administración, 2014.....	221
FIGURA 12-7 La española: Planilla de sueldos patronales, 2014	221
FIGURA 12-8 La española: Gestión de proveedores, modulo provisión, 2014	222
FIGURA 12-9 La española: Compras, modulo provisión, 2014.....	222
FIGURA 12-10 La española: Productos terminados, modulo producción, 2014	223
FIGURA 12-11 La española: Insumos, modulo producción, 2014.....	223
FIGURA 12-12 La española: Pronostico, Programa maestro de prod., mod. Prod., 2014.....	224
FIGURA 12-13 La española: Programación lineal, Programa maestro de prod., mod. Prod., 2014 ..	224
FIGURA 12-14 La española: MRP, 2014.....	225
FIGURA 12-15 La española: Gestión de clientes, modulo ventas, 2014.....	225
FIGURA 12-16 La española: Ventas, modulo ventas, 2014	226
FIGURA 12-17 La española: Registro, modulo contabilidad, 2014	226
FIGURA 12-18 La española: Cuentas, modulo contabilidad, 2014.....	227
FIGURA 12-19 La española: Indicadores, modulo contabilidad, 2014.....	227
FIGURA 12-20 La española: Balance General, contabilidad, 2014	228

FIGURA 12-21 La española: Estado de Resultados, modulo contabilidad, 2014	228
FIGURA 12-22 La española: Reporte boleta de pago, modulo administración, 2014	229
FIGURA 12-23 La española: Planilla de sueldos y salarios, modulo administración, 2014.....	229
FIGURA 12-24 La española: Registro de compra, modulo provisión, 2014.....	230
FIGURA 12-25 La española: Registro de venta, modulo ventas, 2014	230
FIGURA 12-26 La española: Indicadores, modulo contabilidad, 2014.....	231
FIGURA 12-27 La española: Estado de Resultados, modulo contabilidad, 2014.....	231
FIGURA 12-28 La española: Balance General, modulo contabilidad, 2014.....	232
FIGURA 12-29 La española: Autenticación, 2014.....	233
FIGURA 13-1 La española: Ingreso al Sistema, 2014.....	234
FIGURA 13-2 La española: Entorno del Sistema, 2014.....	235
FIGURA 13-3 La española: Modulo de Administración, 2014	235
FIGURA 13-4 La española: Modulo de Administración, 2014	236
FIGURA 13-5 La española: Modulo de Administración, 2014	237
FIGURA 13-6 La española: Modulo de Administración- Nuevo Registro, 2014	238
FIGURA 13-7 La española: Modulo de Administración-Búsqueda de registros, 2014.....	239
FIGURA 13-8 La española: Modulo de Administración-Manual de Funciones, 2014.....	240
FIGURA 13-9 La española: Modulo de Administración-Boletas de Pago, 2014.....	241
FIGURA 13-10 La española: Modulo de Administración-Planilla de Sueldos, 2014.....	242
FIGURA 13-11 La española: Modulo de Provisión, 2014	242
FIGURA 13-12 La española: Modulo de Provisión-Ingreso, 2014.....	243
FIGURA 13-13 La española: Modulo de Provisión-Entorno, 2014.....	243
FIGURA 13-14 La española: Modulo de Provisión-Filtros de Búsqueda, 2014	244
FIGURA 13-15 La española: Modulo de Provisión-Comprobantes de compra, 2014.....	245
FIGURA 13-16 La española: Modulo de Producción, 2014	245
FIGURA 13-17 La española: Modulo de Producción-Entorno, 2014.....	246
FIGURA 13-18 La española: Modulo de Producción-Filtros de búsqueda, 2014.....	247
FIGURA 13-19 La española: Modulo de Producción, 2014	247
FIGURA 13-20 La española: Modulo de Producción-Filtros de búsqueda Insumos, 2014	248
FIGURA 13-21 La española: Modulo de Producción-Pronostico, 2014	249
FIGURA 13-22 La española: Modulo de Producción-Entorno Programación Lineal, 2014	249
FIGURA 13-23 La española: Modulo de Producción-Modo de uso, 2014	250
FIGURA 13-24 La española: Modulo de Producción-MRP, 2014	250
FIGURA 13-25 La española: Modulo de Ventas, 2014	251
FIGURA 13-26 La española: Modulo de Ventas-Entorno, 2014.....	251
FIGURA 13-27 La española: Modulo de Ventas-Filtros de búsqueda, 2014	252
FIGURA 13-28 La española: Modulo de Ventas-Registro de venta 1-3, 2014	252
FIGURA 13-29 La española: Modulo de Ventas-Registro de venta 2-3, 2014	253
FIGURA 13-30 La española: Modulo de Ventas-Registro de venta 3-3, 2014	254
FIGURA 13-31 La española: Modulo Financiero, 2014	254
FIGURA 13-32 La española: Modulo Financiero-Registros Contables, 2014.....	255
FIGURA 13-33 La española: Modulo Financiero-Entorno Administrador de cuentas, 2014.....	255
FIGURA 13-34 La española: Modulo Financiero-Filtros de Búsqueda, 2014	256
FIGURA 13-35 La española: Modulo Financiero-Indicadores Financieros, 2014.....	257
FIGURA 13-36 La española: Modulo Financiero-Balance general, 2014.....	258
FIGURA 13-37 La española: Modulo Financiero-Estado de resultados, 2014	259

ÍNDICE DE DIAGRAMAS

DIAGRAMA 1-1 Sistema de información, 2014.....	5
DIAGRAMA 1-2 Sistema de procesamiento, 2013.....	6
DIAGRAMA 1-3 Sistema de gestión empresarial ideal, 2014	6
DIAGRAMA 1-4 Fases del sistema de información, 2014.....	7
DIAGRAMA 1-5 Fases del análisis de Pareto, 2014.....	9
DIAGRAMA 1-6 La española. Diagrama del pescado para la inadecuada administración, 2014....	11
DIAGRAMA 1-7 La española. Diagrama de Pareto, 2014.....	13
DIAGRAMA 2-1 La española. Diagrama de flujo del proceso, 2014	23
DIAGRAMA 3-1 Fases de la creación de la base de datos, 2014.....	34
DIAGRAMA 4-1 SIS-K. Sistema de gestión empresarial, 2014	72
DIAGRAMA 4-2 SIS-K. Subsistema de Administración de recursos humanos, 2014	73
DIAGRAMA 4-3 SIS-K. Subsistema de Provisión, 2014	73
DIAGRAMA 4-4 SIS-K. Subsistema de producción, 2014.....	74
DIAGRAMA 4-5 SIS-K. Subsistema de Ventas, 2014.....	74
DIAGRAMA 4-6 SIS-K. Sistema de Finanzas, 2014	75
DIAGRAMA 4-7 SIS-K. Diagrama UML de todo el sistema, 2014	75
DIAGRAMA 5-1 Sistema de administración de recursos humanos, 2014.....	82
DIAGRAMA 6-1 Gestión de proveedores, 2014.....	88
DIAGRAMA 6-2 SIS-K: Sistema de provisión, 2014	89
DIAGRAMA 7-1 Objetivos del Merchandising, 2014	97
DIAGRAMA 7-2 Gestión de ventas, 2014	99
DIAGRAMA 8-1 SIS-K. Sistema financiero, 2014.....	106
DIAGRAMA 9-1 Sistema de gestión de la producción, 2014	139
DIAGRAMA 9-2 SIS-K: Sistema de producción, 2014	139
DIAGRAMA 10-1 Fases de la medición del trabajo, 2014	145
DIAGRAMA 10-2 Fases de medición de tiempos improductivos, 2014.....	145
DIAGRAMA 10-3 La Española: Proceso de fabricación general, 2014.....	146
DIAGRAMA 10-4 Fases de la medición de tiempos de operación, 2014	160
DIAGRAMA 10-5 Procedimiento para calcular los suplementos por sección, 2014.....	163
DIAGRAMA 10-6 La española: Diagrama de trozado, 2014.....	167
DIAGRAMA 10-7 La española: Diagrama de molido, 2014	168
DIAGRAMA 10-8 La española: Diagrama de preparación de aditivos, 2014.....	169
DIAGRAMA 10-9 La española: Diagrama de mezclado, 2014.....	170
DIAGRAMA 10-10 La española: Diagrama de embutido, 2014.....	171
DIAGRAMA 10-11 La española: Diagrama de ahumado, 2014	172
DIAGRAMA 10-12 La española: Diagrama de cocinado, 2014	173
DIAGRAMA 10-13 La española: Diagrama de enfriado, 2014.....	174
DIAGRAMA 10-14 La española: Diagrama hombre-máquina para Carne molida, 2014.....	182
DIAGRAMA 10-15 La española: Diagrama Hombre-máquina para Salchicha, 2014	183
DIAGRAMA 10-16 La española: Diagrama hombre-máquina para Chorizo, 2014.....	184
DIAGRAMA 10-17 La española: Diagrama hombre-máquina para Mortadela, 2014	185
DIAGRAMA 10-18 La española: Diagrama hombre-máquina para Jamón, 2014	186
DIAGRAMA 10-19 La española: Diagrama hombre-máquina para Salame, 2014.....	187
DIAGRAMA 10-20 Pronostico de la demanda, 2014	189
DIAGRAMA 10-21 Método de Holt-Winters, 2014	194
DIAGRAMA 11-1 Fases de la programación entera, 2014	202
DIAGRAMA 14-1 Fases para la evaluación financiera, 2014.....	260

ÍNDICE DE CUADROS

CUADRO 1-1 La española. Análisis de Pareto filtrado por incidencia, 2014	12
CUADRO 1-2 La española. Soluciones tentativas, 2014.....	13
CUADRO 1-3 Símbolos para la elaboración de diagramas, 2014.....	18
CUADRO 2-1 La española. Características de la empresa, 2014.....	21
CUADRO 2-2 La española. maquinaria y equipo, 2014.....	23
CUADRO 2-3 La española. Consumo de agua – promedio anual, 2008-2012.....	24
CUADRO 2-4 La española. Consumo de energía eléctrica, 2012.....	24
CUADRO 2-5 la española: consumo de energía eléctrica, 2008-2012.....	24
CUADRO 2-6 La española: consumo de combustible, 2012.....	25
CUADRO 3-1 Atributos de la información, 2014.....	30
CUADRO 3-2 Aptitudes de los gestores, 2014.....	32
CUADRO 3-3 DB SIS-K. Funciones de agregado, 2014.....	46
CUADRO 3-4 Referencia cruzada ACCESS ,2014.....	51
CUADRO 3-5 Sintaxis de los criterios de selección, 2014.....	52
CUADRO 3-6 Ejemplo de Sintaxis de los criterios de selección, 2014.....	52
CUADRO 3-7 Operador LIKE, 2014.....	53
CUADRO 4-1 Aptitudes de los IDE, 2014.....	72
CUADRO 5-1 La española. Ejemplo de registro para la gestión de empleados, 2014.....	82
CUADRO 6-1 La española. Ejemplo de registro para la gestión de proveedores, 2014.....	89
CUADRO 7-1 La española. Ejemplo de registro para la gestión de clientes, 2014.....	99
CUADRO 8-1 Razones financieras, 2014.....	105
CUADRO 8-2 La española. Ejemplo de registro de cuentas, 2014.....	107
CUADRO 10-1 La Española: Tamaño de muestra anticipado, 2014.....	147
CUADRO 10-2 La Española: Estadístico de Puebla, 2014.....	148
CUADRO 10-3 La Española. Estimacion de Parametros, 2014.....	149
CUADRO 10-4 La Española. Estadístico de prueba, 2014.....	149
CUADRO 10-5 La Española: Estimacion de Parametros, 2014.....	152
CUADRO 10-6 La Española: Estadístico de prueba, 2014.....	152
CUADRO 10-7 La Española: Estadístico de prueba, 2014.....	154
CUADRO 10-8 La Española: Estimacion de los Parametros, 2014.....	155
CUADRO 10-9 La Española: Estadístico de prueba, 2014.....	155
CUADRO 10-10 La Española: Estimacion de los Parametros, 2014.....	158
CUADRO 10-11 La Española: Estadístico de prueba, 2014.....	158
CUADRO 10-12 La española: Resumen de estudio de paradas, 2014.....	160
CUADRO 10-13 La española: Tiempos anticipados al Estudio de tiempos y movimientos, 2014.....	161
CUADRO 10-14 La española: Tamaño de muestra real para la operación de trozado, 2014.....	162
CUADRO 10-15 La española: Tamaño de muestra para las operaciones, 2014.....	163
CUADRO 10-16 Suplementos recomendados por la OIT.....	165
CUADRO 10-17 La española: Resumen de suplementos por sección, 2014.....	166
CUADRO 10-18 La española: Resumen de suplementos por dificultad de producto, 2014.....	166
CUADRO 10-19 La española: Resumen de estudio de tiempos de tareas, 2014.....	176
CUADRO 10-20 La española: Elaboración de Salchicha, 2014.....	177
CUADRO 10-21 La española: Elaboración de Chorizo, 2014.....	178
CUADRO 10-22 La española: Elaboración de Mortadela, 2014.....	179
CUADRO 10-23 La española: Elaboración de Jamón, 2014.....	180
CUADRO 10-24 La española: Elaboración de Salame, 2014.....	181
CUADRO 10-25 La española: Tiempos de elaboración por producto [Hr], 2014.....	187
CUADRO 10-26 La española: Composición de los tiempos de elaboración [Hr], 2014.....	188
CUADRO 10-27 La española: Rendimiento de los productos según horario [Hr/Kg], 2014.....	189

CUADRO 10-28 La española, Demanda histórica [Kg], 2012 - 2013.....	190
CUADRO 10-29 La española, Estimación de Parámetros 2012 - 2013.....	193
CUADRO 10-30 La española: Salchicha, parámetro de estacionalidad, 2014	197
CUADRO 10-31 La española: Parámetros Iniciales del modelo Holt-Winters, 2014	198
CUADRO 10-32 La española: Coeficientes del modelo Holt-Winters, 2014.....	200
CUADRO 10-33 La española: Pronostico de la demanda [Kg], 2014.....	201
CUADRO 11-1 La española: Coeficientes tecnológicos [Bolivianos/Kg], 2014	204
CUADRO 11-2 Solución de las variables y satisfacción de las restricciones, 2014.....	208
CUADRO 11-3 La española: Análisis de sensibilidad, 2014	208
CUADRO 14-1 La española. Beneficios del sistema [Bolivianos], 2014.....	263
CUADRO 14-2 La española. Cotización de la instalación [Bolivianos], 2014	263
CUADRO 14-3 La española. Tasa de oportunidad de la inversión, 2014	263
CUADRO 14-4 La española. Flujo de fondos [Bolivianos], 2014.....	264



Resumen

En el Planteamiento del problema hemos logrado plantear el problema como un sistema de gestión empresarial. Para las fases de un sistema de información este sistema propuesto está en la etapa de madurez, es decir que tiene la forma final de un sistema de gestión empresarial. A este sistema lo hemos denominado (SIS-K).

SIS-K es un sistema informático empresarial creado por los mismos autores (Vargas Arce y Blanco Aguilar 2014), este sistema es creado como un estándar para todos los sectores empresariales. Este sistema tiene principal aplicación en las empresas de diseño y producción flexible.

SIS-K es un ERP, pero este no está diseñado como uno. El diseño se basa en la creatividad de los autores, no se ha usado ningún proyecto base o plantilla para la codificación de este sistema.

En la empresa “La española”, hemos reunido los datos básicos de la empresa para tener una referencia de los campos que tenemos que incluir en el modelo de datos de la entidad (Entity Data Model, EDM) y de los parámetros de ingeniería que tenemos que calcular.

En el Sistema de información, hemos diseñado la base de datos (Data Base, DB) y su diagrama de entidad-relación (EDM). Lo cual es nuestra base para el sistema de información empresarial. Esta base de datos se denomina (SIS-K DB). El motor de base de datos está creada en la herramienta (Microsoft SQL Server 2012).

En el Diseño del sistema informático SIS-K, hemos designado el sistema de gestión empresarial, describiendo los módulos que lo integran. El sistema está diseñado en el entorno de desarrollo integrado (Integrated Device Electronics, IDE) de (Microsoft Visual Studio 2012) en el lenguaje VB.NET. La arquitectura es la que presenta por defecto el programa, no nos hemos enfocado en este tema. Todo el diseño está en el diagrama de clases (Unified Modeling Language, UML) para su mejor comprensión.

En el Diseño del Módulo Administración de recursos humanos, Diseño del Módulo Provisión, Diseño del Módulo Ventas, Diseño del Módulo Finanzas y Diseño del Módulo Producción, hemos diseñado los módulos componentes del sistema, mostrando los requerimientos que necesitan y sus procedimientos.

En Ingeniería del proyecto, se generaron los parámetros de rendimiento y pronóstico de la demanda requerido por el programa. Los parámetros de trabajo van relacionados con los tiempos improductivos, tiempos de operación, suplementos y rendimiento de producción. El pronóstico de la demanda se basa en una metodología propia de los autores, que se adapta a todo tipo de caso.

En el Diseño de Métodos, hemos desarrollado el modelo de programación lineal (MPL) para optimizar la producción, también hemos conectado con un complemento COM para el algoritmo de optimización del modelo. Para terminar hemos transformado el modelo de programación lineal (MPL) en un modelo informático, con este capítulo se cumple con el objetivo general.

En el Diseño de la interfaz, basado en el enfoque sistémico se siguió la metodología del prototipo de sistemas para desarrollar la aplicación informática. Esta metodología consta de 5 etapas: en la primera se determinan los requerimientos que debe satisfacer la aplicación; en la segunda se desarrolla un modelo de trabajo, se determina el lenguaje de programación a utilizar, el diseño de la base de datos, la interfaz para el usuario y las salidas del sistema a través de un diagrama de clases en UML; en la tercera se utiliza el prototipo con un estudio de caso de la UML, se evalúa su características de operación utilizándolas bajo situaciones reales; en la cuarta se revisa el prototipo, se realizan cambios que aumenten la eficiencia de la aplicación; y la quinta repite las dos anteriores etapas las veces que sean necesarias hasta obtener el sistema idóneo. Para la modelación hemos utilizado el programa (Microsoft Visual Studio 2012), con el lenguaje de codificación VB.NET, el motor de base de datos utilizado es (Microsoft SQL Server 2012).

En el Manual del Usuario, se muestra mediante impresiones de pantalla como se debe utilizar el sistema para realizar sus diversas funciones y explica los procedimientos que se deben seguir para utilizar algunas herramientas del sistema.

En la Evaluación Financiera, en este capítulo se transformaron todos los beneficios del sistema en beneficios financieros a través del costo de oportunidad, se creó un flujo de fondos para un horizonte de 5 años y finalmente se realizó la evaluación final de la implementación del programa a través del índice VAN.

Palabras clave: Ingeniería de sistemas, Sistema informático empresarial, Sistema de gestión estratégica, Pronóstico, Programación de la producción, Desarrollo de métodos, Base de datos, modelo de datos de la entidad, ERP, MPL, UML, SQL, IDE.

Abstract

In the Proposal of the problem, we have managed to raise the issue like a system of management. For the phases of an information system this proposed, the system is in the maturity stage, which means that it has the final way of a system of management. We have called this system (SIS K).

SIS K is a software enterprise system created by the same authors (Vargas Arce y Blanco Aguilar 2014), this system is created like a standard for all the business sectors. This system has main application in enterprise with design and flexible production.

SIS-K is an ERP, but this is not designed like one. The design is based on the creativity of the authors, it has not done use any base project o template for the codification of this system.

In The Enterprise “La Española”, we have gathered the basic company data to have a reference of the fields that we have to include in the Entity Data Model (EDM) and of the parameters of engineering that we have to calculate.

In the Information system, we design the data base (DB) and its diagram of entity relation (EDM). Which is our base for the enterprise information system. This data base called itself (SIS K DB). The engine data base this created in the program (Microsoft SQL Server 2012).

In the Information-technology systems Design, we have designed the system of management, describing the moduli that integrate it. The system is designed by the Integrated Device Electronics (IDE) by (Microsoft Visual Studio 2012) in the language VB.NET. Architecture is the one that presents by default the program, we have not focused on this section. Whole design is in the diagram of classes (UML) for their better understanding.

In the Design of the module Human Resources Management, Design of the module Provision, Design of the module Sales, Design of the module Finance, Design of the module Production have designed the component moduli of the system, showing the requests that they need and his procedures.

In the Engineering of the project, they generated the parameters of performance and forecast of the request required by the program themselves. The parameters of work go once unproductive times were related to, uptimes, supplements and performance of production. The forecast of the request is based on a methodology of its own of the authors, which he adapts to every kind of forecast.

In the Design of methods, we have developed the linear programming (LP) to analyze the optimize production, we have also connected with the add-in for the algorithm of the model's optimization. In order to finish we have turned the linear programming (LP) into a computer model, with this chapter it comes true with the objective general.

In the Design of interface, once the systems approach was based on the methodology of the prototype of systems to develop the information-technology application ensued. This methodology consists of five stages: for the first they determine the requests that the application should satisfy; for the second a model of work develops, the programming language is determined to use, the design of the data base, the interface for the user and logoffs through by diagram of classes; for the third class we used prototype with the case study from UML, it is evaluated his characteristics of operation using them under real situations; In the quarter he checks his prototype, changes come true that they increase the efficiency of the application; And the country house repeats the two previous stages the times that are necessary to obtain the ideal system. For the modelation we have used the program (Microsoft Visual Studio 2012), with the coding language VB.NET, the base motor of data used is (Microsoft SQL Server 2012).

In the Manual of User, shows up by means of impressions of screen as the system to accomplish his various functions should be used and explain the procedures that should ensue to use some tools of the system.

In the Financial evaluation, all the benefits of the system turned into financial benefits through the opportunity cost, created for himself a flow of funds for a horizon of 5 years and finally came true the final evaluation of the implementation of the program through the indicator VPN.

Key Words: Systems engineering, Information-technology enterprise system, Business management system, Forecasting, Programming of production, Development of methods, Data base, Entity data model, ERP, LP, UML, SQL, IDE.

CAPÍTULO 1. PLANTEAMIENTO DEL PROBLEMA

1.1 Introducción

El objetivo de este capítulo es hablar de la gestión y por qué se constituye en el problema de fondo de la empresa para ello se dan varias nociones de lo que es las bases de la gestión.

1.1.1 Definiciones

Gestión, es la administración integral. De esta palabra deriva el gerenciamiento.

Sistema; es un conjunto de elementos interrelacionados entre sí para lograr un objetivo final.

Informática, es la disciplina que lidia con el tratamiento racional y sistémico de la información (por medios automáticos)¹.

1.2 Bases de la gestión

El objetivo final de un Ingeniero es la de gestionar, entonces yendo hacia ese lado, el gerente o científico debe mejorar continuamente su gestión.

La teoría de la información, indica que una fuente emite un mensaje a través de un emisor el cual codifica el mensaje y luego este es transportado hasta un receptor el cual decodifica el mensaje y lo muestra en destino.



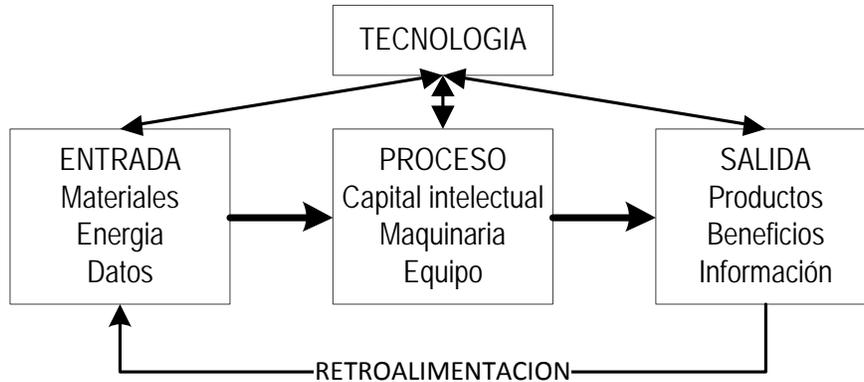
Fuente: (Pando Viamont 2013)

La teoría de sistemas; indica que dada una entrada, ejecutada por un proceso, obtiene una salida, donde puede existir una retroalimentación.

¹ Esta definición es de (Chiavenato 2006, 367)

DIAGRAMA 1-2

Sistema de procesamiento, 2013



Fuente: (Pando Viamont 2013)

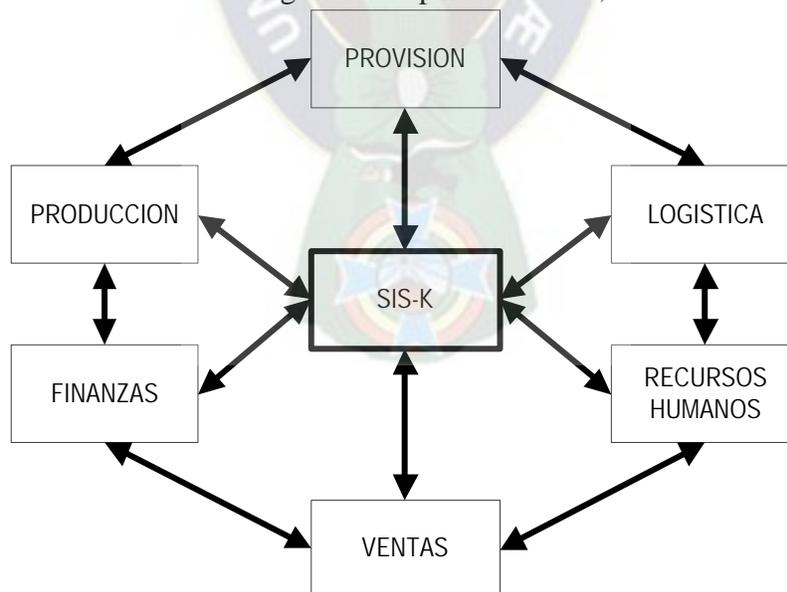
La teoría general de sistemas según (Hurtado Carmona 2011) indica que mientras más grande sea el sistema más control tendremos sobre el mismo.

El enfoque de sistemas, considera un análisis integral e interdisciplinario de cada una de las funciones de la empresa con visión globalizadora. Se resalta por utilizar las bondades de las corrientes anteriores.

Entonces vamos a definir nuestro sistema ideal llamado SIS-K, para intentar llegar a consolidarlo.

DIAGRAMA 1-3

Sistema de gestión empresarial ideal, 2014

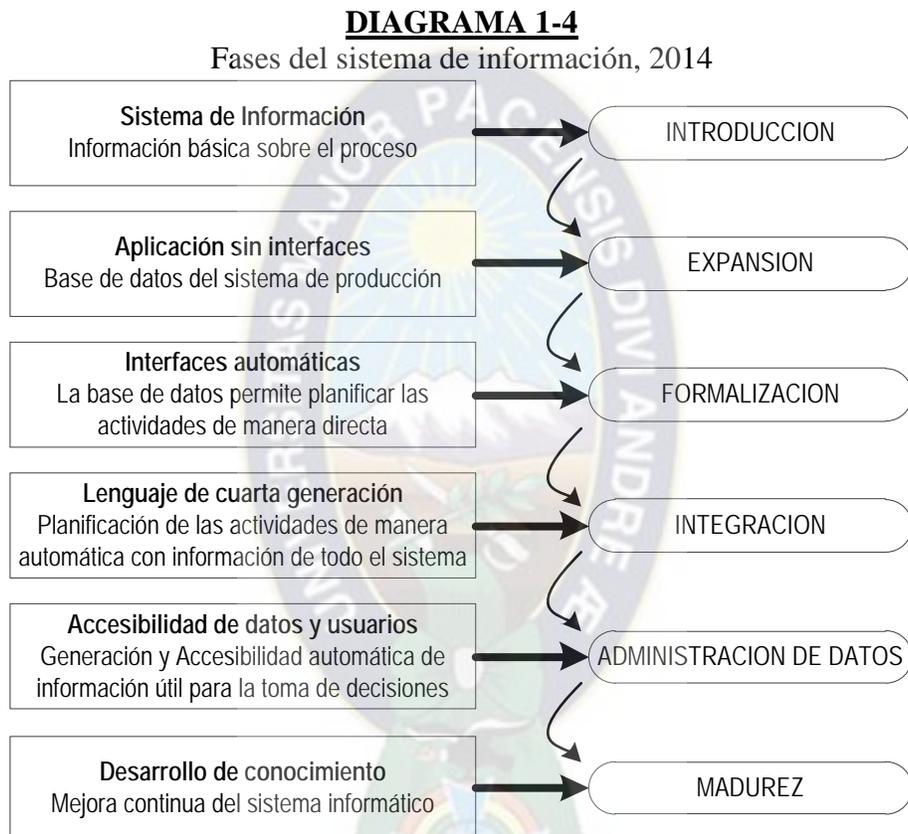


Fuente: Elaborado en base a (Pando Viamont 2013)²

² El sistema propuesto por (Pando Viamont), propone que los componentes del sistema son: Provisión, Compras, Logística, Producción, Distribución y Comercialización. Para nuestro razonamiento comprendemos que el subsistema de Provisión, Logística y Distribución; va referido a un sistema de información geográfico y los demás a un sistema no geográfico.

Para nuestro sistema de base el subsistema de Logística engloba la Logística de Aprovechamiento (Provisión), Producción y Distribución. Y este es un sistema de información geográfico, el cual lo vamos a obviar por la complejidad de los sistemas de información geográfico. Entonces esto da paso a más módulos en el sistema como ser el módulo de Administración del personal y Finanzas que son de gran ayuda a empresas del sector Pequeño, Mediano y Grande; pero no a los del sector Micro³.

En las fases de desarrollo del sistema de información, se tienen seis etapas que se muestran en el siguiente gráfico.



Fuente: (Pando Viamont 2013)

El sistema SIS-K que vamos a concebir como aplicación de la empresa “La española”, se encuentra en la fase cinco Administración de datos. Para que este sistema llegue a etapa de madurez, se requiere por lo menos una retroalimentación, lo cual se dará en el futuro.

1.3 Problemática

El sistema de procesamiento o sistema técnico-social de la empresa está en la primera fase de un sistema de información ideal, esto afecta enormemente a la gestión empresarial y para que se tenga una mejor comprensión de ello se lo ilustrará con un ejemplo. Entramos

³ Para el sector Micro es recomendable solo los módulos de Ventas, Provisión y Producción para el segmento Productivo.

en el supuesto que la solución sea plantear un sistema de agotamiento de depósitos el cual explicaremos más adelante, la solución estará planteada como la combinación lineal de los costos de inventario y su existencia, para esto requerimos de un servicio de análisis de datos el cual puede ser MS Office Excel, también supongamos que el motor de datos no requiere de un diseño de campos, entonces podemos analizar los datos, como existe la probabilidad de que los costos de la empresa cambian constantemente e inesperadamente, entonces tenemos que cambiar la producción en función de un nuevo costo, dado que el Libro de MS Office Excel abarca un gran número de hojas y celdas por hoja, entonces al tener tanta información no se puede procesarla con eficiencia, esta ineficiencia repercute en otros departamentos generando grandes pérdidas de dinero a la empresa, las cuales pueden llegar a 7000 [bolivianos/hora].

Aquí otro caso similar entramos en el supuesto que la solución sea plantear un sistema de jalar o justo a tiempo, la solución estará planteada como un sistema de tarjetas “KANBAN” que generan el programa de producción, estas tarjetas son calculadas manualmente por una persona de la empresa, entonces llega un pedido el cual requiere de un plazo de entrega mínimo entregado en frontera, pero el cliente esta tan desesperado que quiere una hora de entrega bastante pronta, entonces nos da los requerimientos para calcularle el tiempo de entrega, dado lo complejo de los cálculos de las tarjetas se tarda en el cálculo y entonces el cliente se enoja y declina del negocio, entonces se deben tomar medidas para evitar este error, que tiene un costo de oportunidad de la ganancia que hubiese generado ese negocio.

Ahora vamos a decir que estos casos más el problema de la calidad son los problemas más comunes dentro de cualquier empresa. El problema del primer caso es el uso inadecuado de un motor de datos y servicio de análisis de datos. El problema no es el usuario sino el sistema. Entonces si se hubiese tenido un sistema con interfaz sencilla y secuencial, que solo muestre los campos que puedan ser editados, se hubiera resuelto el problema de la producción sin tomar tanto tiempo y sin la necesidad de poner tanta concentración en una pantalla. El segundo caso podría ser resuelto con el sistema del primer caso, pero ya demostramos que el primer sistema no es el más adecuado, entonces la solución primaria es la misma.

Si volvemos al primer caso y sin querer cambiamos algo del análisis en el Libro de MS Office Excel y salimos del libro sin guardar cambios, existe la probabilidad de que ese cambio erróneo se quede en el libro. Tenemos toda una serie de cálculos que se realizaran erróneamente, corregir el sistema nos llevara al menos una semana resolverlo. Después de

revisar todo este embrollo, concluimos que lo mejor sería aplicar un sistema de interfaz sencillo.

En el primer caso existe la probabilidad de un robo de información en segundo plano, es decir que existe la probabilidad de que modifiquen o roben el libro que nos costó 180 [horas] de edición sin darnos cuenta.

Después de revisar este ejemplo se concluye que el problema es llevar el sistema de información de la fase de Introducción en el que se encuentra la empresa a la fase de Administración de datos.

Idealmente requerimos de un sistema informático que vuelva efectivos y automáticos todos los métodos de Ingeniería Industrial, además que protejan el gestor de datos, además que tengan procesos automáticos de análisis.

Entonces ya habiendo definido nuestra problemática vamos a emplear el análisis de Pareto para inducir la solución a la problemática planteada.

1.4 Fases del análisis de Pareto



Fuente: Elaborado en base a (Niebel y Freidvalds 2004) y (Organización Internacional del Trabajo 1996)

1.4.1 Lluvia de ideas

- Registros manuales
- Registros inadecuados de compras y ventas
- Falta de Administración de personal
- Falta Manual de funciones
- Falta de Organización jerárquica
- Ningún control de la información
- Falta de comunicación entre áreas

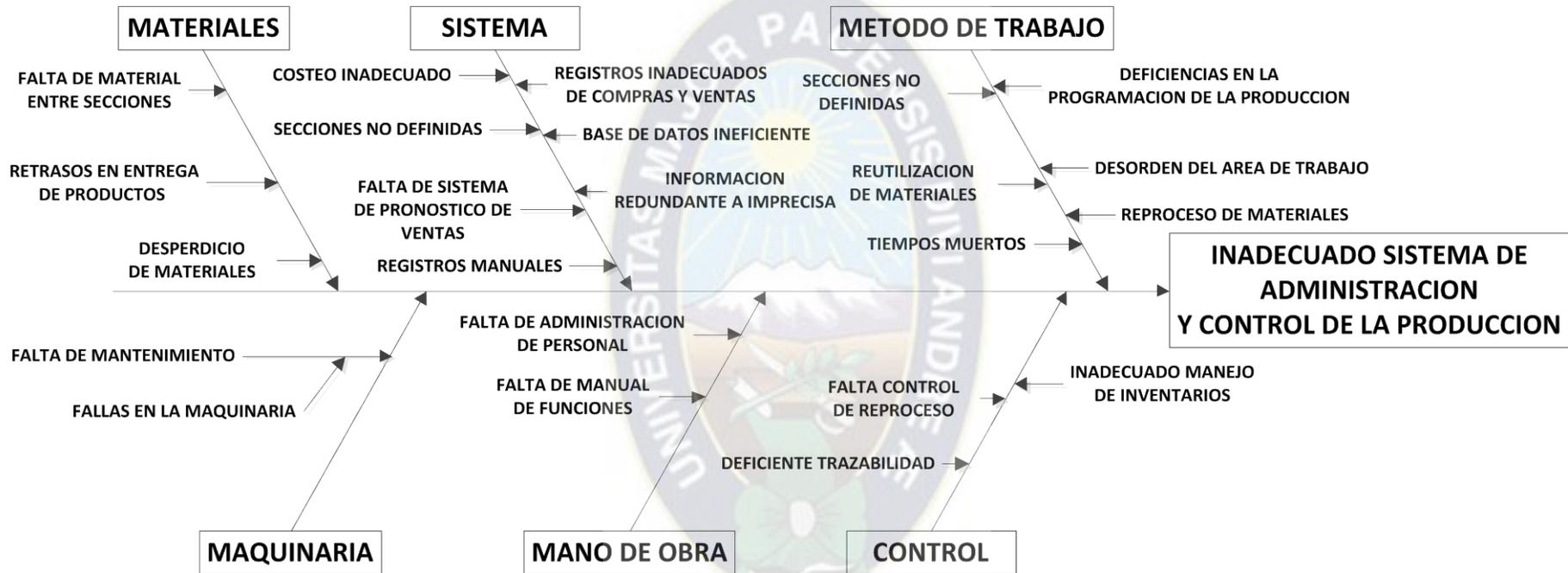
- Deficiencias en la programación de la producción
- Base de datos ineficiente
- Inadecuado manejo de inventarios
- Información redundante e imprecisa
- Costeo inadecuado
- Falta de sistema de pronóstico de ventas
- Retrasos en la entrega de productos
- Tiempos muertos
- Ninguna planeación de pedidos
- Desorden del área de trabajo
- Falta de mantenimiento
- Falta de material entre las secciones
- Deficiente trazabilidad
- Reproceso de materiales
- Reutilización de materiales
- Desperdicio de materiales



1.4.2 Diagrama del pescado

DIAGRAMA 1-6

La española. Diagrama del pescado para la inadecuada administración, 2014



Fuente: Elaborado en base a la lluvia de ideas y Gerencia General

1.4.3 Análisis de Pareto

CUADRO 1-1

La española. Análisis de Pareto filtrado por incidencia, 2014

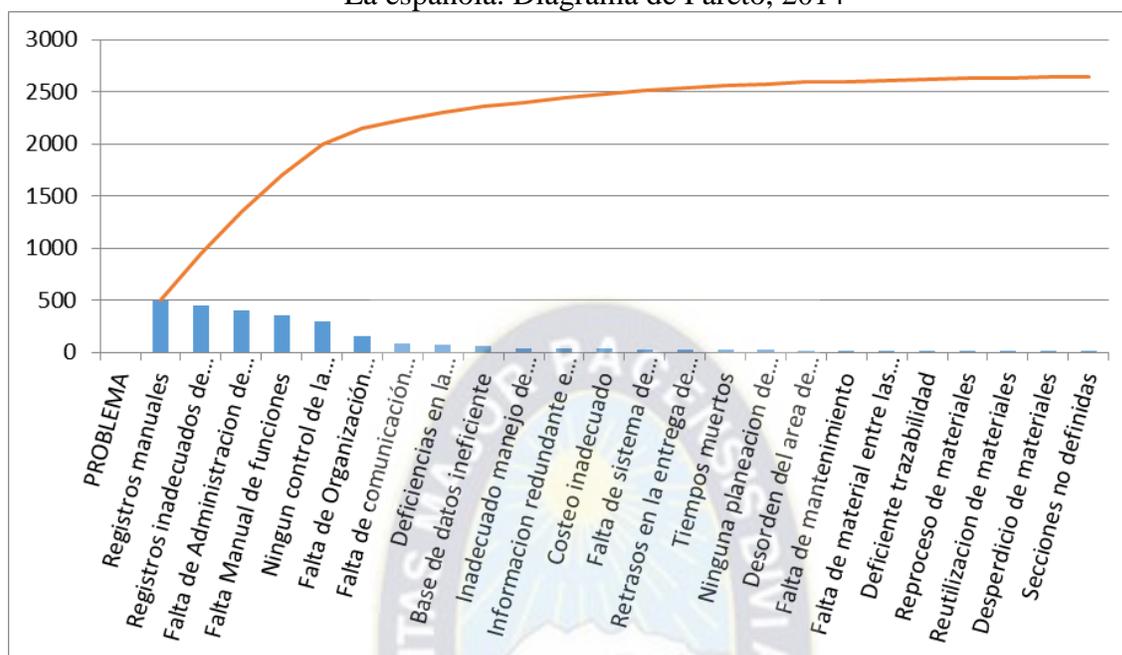
PROBLEMA	INCIDENCIA	PORCENTAJE ACUMULADO
Registros manuales	500	18,90%
Registros inadecuados de compras y ventas	450	35,92%
Falta de Administración de personal	400	51,04%
Falta Manual de funciones	350	64,27%
Ningún control de la información	300	75,61%
Falta de Organización jerárquica	150	81,29%
Falta de comunicación entre áreas	80	84,31%
Deficiencias en la programación de la producción	70	86,96%
Base de datos ineficiente	60	89,22%
Inadecuado manejo de inventarios	40	90,74%
Información redundante e imprecisa	40	92,25%
Costeo inadecuado	40	93,76%
Falta de sistema de pronóstico de ventas	30	94,90%
Retrasos en la entrega de productos	25	95,84%
Tiempos muertos	20	96,60%
Ninguna planeación de pedidos	20	97,35%
Desorden del área de trabajo	15	97,92%
Falta de mantenimiento	10	98,30%
Falta de material entre las secciones	10	98,68%
Deficiente trazabilidad	10	99,05%
Reproceso de materiales	10	99,43%
Reutilización de materiales	5	99,62%
Desperdicio de materiales	5	99,81%
Secciones no definidas	5	100,00%

Fuente: Elaborado en base al diagrama de pescado y Gerencia de Producción

1.4.4 Diagrama de Pareto

DIAGRAMA 1-7

La española. Diagrama de Pareto, 2014



Fuente: Elaborado en el programa (Microsoft Office Excel 2013) en base al CUADRO 1-1

Según el análisis de Pareto, si resolvemos los primeros cinco problemas entonces vamos a resolver el 80% de los problemas.

1.4.5 Soluciones tentativas

La solución planteada para los problemas planteados se refleja en la siguiente tabla.

CUADRO 1-2

La española. Soluciones tentativas, 2014

PROBLEMA	SOLUCIÓN	
Registros manuales	Herramienta informática	
Registros inadecuados de compras y ventas	Herramienta informática	
Falta de Administración de personal	Herramienta informática	
Falta Manual de funciones	Herramienta informática	Manual de funciones
Ningún control de la información	Herramienta informática	

Fuente: Elaborado con base en sugerencia de los autores, Gerencia General y Gerencia de producción

Entonces la solución final para resolver el problema es crear una herramienta informática. Para tal caso necesitamos reunir primero parámetros y limitaciones de las dimensiones que

tiene el alcance nuestro estudio. Entonces tenemos que hacer un estudio del trabajo y un estudio de la capacidad.

1.5 Objetivos

1.5.1 Objetivo general

Mejorar el sistema de administración y control de la producción utilizando el “SIS-K” para optimizar la gestión empresarial.

1.5.2 Objetivos específicos

- Transformar la base de datos física en una base de datos digital.
- Generar los parámetros de los procesos de producción y comercialización.
- Desarrollar los métodos requeridos por la empresa: programación de la producción, gestión de inventarios, gestión del personal, generador de reportes financieros.
- Desarrollar los módulos requeridos por la empresa: producción, administración de recursos humanos, provisión, ventas y finanzas.
- Transformar el método para programar la producción en un método informático.
- Desarrollo de un sistema de seguridad de la información.
- Desarrollo de la aplicación informática.
- Evaluar el proyecto de forma económica y financiera.

1.5.3 Metodología

El método a utilizar empieza con un planteamiento del problema según el Análisis de Pareto, del cual encontraremos las posibles soluciones. Dentro del proyecto se evaluará el problema de producción según la programación entera, que tendrá las dimensiones: recursos mano de obra directa, recursos materiales, recursos monetarios, recursos de maquinaria.

Luego se creará la base de datos digital, se crearán las tablas según la teoría del cubo multidimensional. Realizar la transformación lineal de reales de dimensión n a reales de dimensión dos e actualizar los datos en las tablas, finalmente armando nuestro diagrama de entidad relación, con esto ya tenemos nuestra base de datos consolidado.

Se aplica una consultoría para reunir los parámetros y requerimientos de la empresa en el desarrollo de métodos y las consultas potencialmente requeridas. Luego transformar los métodos y consultas en forma digital, se realizará la consulta externa para validar los métodos, los parámetros, y la validación del programa final.

Luego se crean los métodos que solucionarían los problemas dentro del sistema, entonces transformaremos estos procedimientos en análisis informático para implementar directamente al sistema.

Finalmente se creará el sistema informático integral, en MS Visual Studio 2012, con todo lo anterior mencionado.

Por último se redactará una memoria técnica paralelamente redactando todo lo anterior que se mencionó, al final redactaremos un manual para el usuario y finalmente el análisis según la perspectiva financiera-económica.

Cabe recalcar que el sistema informático no soporta sistemas de información geográfica, por lo cual no vamos a intervenir en la ingeniería logística (aprovisionamiento, distribución de planta y distribución de producto). Entonces lo que vamos a hacer en el estudio de procesos es dar opciones e integrarlos al sistema de maximizar el valor agregado.

1.6 Cronograma de trabajo

FIGURA 1-1
SIS-K y La española. Cronograma de trabajo, 2014



Fuente: Elaborado en el programa (Microsoft Project 2010) en base a experiencia previa

1.7 Procedimientos auxiliares

Para la elaboración de este proyecto se tienen tres procedimientos auxiliares uno para citas bibliográficas, para diagramas y para la programación del sistema informático que se desglosan a continuación.

1.7.1 Procedimiento para citas y entrada bibliográfica

Para mencionar las citas y bibliografía del documento vamos a basarnos únicamente en la norma APA (sexta edición). La información recomendada que requerimos de las fuentes son:

Autor.- Es el nombre del o de los autores o también se puede introducir un nombre corporativo.

Título del capítulo o entrada.- Es el nombre de la sección de la publicación.

Título de la publicación.- Es el nombre literal de la publicación.

Fecha.- Es la fecha de publicación o lectura de la publicación en caso de documentos electrónicos.

Ciudad.- Es la ciudad de publicación.

Provincia o estado.- Es la provincia o estado de la publicación.

País o región.- Es el país o región de la publicación.

Editorial.- Es el grupo editorial de la publicación.

La información extra que requerimos de las fuentes son:

Páginas.- Son el grupo de páginas donde se muestra la extracción esencial de la información.

Traductor.- Es la persona autorizada por los autores y la editorial para traducir el documento del idioma original a otro.

Edición.- Es el número de edición del documento, esto sirve para evitar confusión entre ediciones.

URL.- Es la dirección en la red de dominio público del documento.

La información recomendada es estándar para todo tipo de documentos. En la sección de ANEXO A están los requerimientos extra según el tipo de documento y formato según la norma. Lo que hicimos en esta sección es solo conceptualizar la información de la norma.

1.7.2 Procedimiento para elaboración de diagramas

Para elaborar los diagramas de flujo vamos a elaborar un procedimiento propio basado en la norma ISO 9000 y ANSI para diagramas de flujo⁴. Los conceptos que requerimos se basan en diagramas de tres tipos de arquitectura.

- De línea
- De anillo
- De estrella

Considerando los requerimientos de la empresa, que tiene una política de mejora continua pero no así de un sistema de gestión estratégica vamos a utilizar la arquitectura de anillo para los procesos de la consultoría.

Los conceptos principales que necesitamos para la elaboración de diagramas son:

Material (Dato).- Es el insumo principal de un proceso de producción, este siempre deben ir junto a su información o especificación técnica (metadato⁵).

Proceso.- Es la ejecución de una operación en la cual debe intervenir la acción manual.

Control.- Es la consulta de una operación en la cual debe intervenir la acción mental.

Decisión.- Es la consulta para filtrar el flujo de operaciones.

Optimización.- Es un sistema de retroalimentación que tiene por objetivo optimizar un proceso.

Transporte.- Es el movimiento de materiales de una sección a otra⁶.

Almacén.- Es la base de materiales y datos.

Documento.- Es la salida de información que no es necesariamente un dato para almacenar.

Los símbolos de estos conceptos se muestran a en el siguiente gráfico.

⁴ La base más importante del procedimiento es la diagramación común de lenguaje de programación.

⁵ Aquí estamos haciendo una analogía del uso de datos. Un material es para una empresa es lo que un dato es para un programa, y una información técnica es para una empresa lo que un metadato es para un programa.

Empresa → Material → Información técnica
Programa → Dato → Metadato

⁶ En el documento llamaremos a este transporte (almacén intermedio).

CUADRO 1-3

Símbolos para la elaboración de diagramas, 2014

	Material
	Proceso
	Control
	Decisión
	Optimización
	Transporte
	Almacén
	Documento

Fuente.- Elaborado en base a la norma ISO 9000 y ANSI para diagramas de Flujo.

1.7.3 Procedimiento para la programación del sistema informático

Para la programación del sistema informático vamos a basarnos en el lenguaje VB.NET.

Los códigos básicos para programar el sistema son:

FIGURA 1-2

Conceptos de lenguaje VB.NET, 2014

```
'Definición de clase
PublicClassForm1
'Definición de proceso
PublicSub Codigos_basicos()
'Dimensionamiento y valor inicial de una variable
Dim variable As String = "texto"
Dim numeroentero As Integer = 0
Dim numerodecimal As Double = 0
'Proceso de calculo
    numerodecimal = numeroentero + 1.3
'Proceso de decisión simple
If numerodecimal < 2 Then numerodecimal = 4
'Procesos de decisión complejo
SelectCase variable
Case"texto"
variable = "contenido 1"
Case""
    variable = "contenido 2"
Case Else
    variable = "contenido 3"
EndSelect
'Proceso cíclico sin condición
For numeroentero = 0 To 10 Step 2
    numerodecimal = numerodecimal + 1.3
Next
'Proceso cíclico con condición
DoWhile numerodecimal = 10
    numerodecimal = numerodecimal - 0.1
Loop
EndSub
EndClass
```

Fuente: Elaborado en base a (Microsoft Visual Studio 2012)

Este código fuente es elaborado para una mejor comprensión de los conceptos básicos de codificación. Las líneas de código de color negro son introducidas por el usuario, las líneas de código de color azul son escritas por la herramienta de modelación, las líneas de color verde no intervienen en el proceso o clase y son de ayuda al usuario.

Una clase es un conjunto de procesos, un proceso es un conjunto de operaciones. El dimensionamiento es la forma de una variable a la cual se le va a asignar valores según forma. Un proceso de decisión es un filtro que se le hacen a los datos. Un proceso cíclico es un conjunto de operaciones que se le hacen a los datos.

CAPÍTULO 2. LA EMPRESA

2.1 Historia

Fiambres y Embutidos “La Española” da sus inicios halla por los años 70 con su mentor el Sr. Pablo Mendoza B.; con la elaboración de fiambres y embutidos, cuidando siempre de dar un producto de excelente calidad a un precio económico.

En los años 90 da inicio la actividad de panificación y se da un repunte en las actividades de la empresa prueba clara es la variedad de productos que se elaboran con tecnología de punta. Actualmente está a cargo del señor Julio Mendoza, hijo del iniciador del negocio.

La empresa está ubicada en la ciudad de El Alto, zona Brasil (Rio Seco), calle Villa Bella N° 447. El terreno es de superficie cuadrada de 1600 (m²) y el área construida donde funciona tanto la administración como la planta productiva es de 580 (m²).

La producción de embutidos alcanza aproximadamente a 53.600 (Kg.) mensuales.

“La Española” es una empresa, que elabora principalmente salchichas, destinado al consumo interno y como cualquier otra empresa “La Española” se preocupa por la mejora de sus procesos y está en constante lucha por controlar sus inventarios o planear su producción diaria debido a que la competitividad exige una mejora continua día tras día.

2.2 Misión

Elaborar productos de alta calidad que satisfagan las necesidades de sus clientes para ello cuenta con una absoluta higiene en el proceso productivo y trabaja con materiales e insumos de alta calidad rigurosamente examinados antes de procesarlos.

2.3 Visión

Ser la empresa líder de su ramo en el mercado nacional a través de la captación e incremento de su cartera de clientes, satisfaciendo las necesidades y exigencias del cliente, con productos de alta calidad a precios competitivos, para lograr ser una empresa altamente rentable.

2.4 Definición de la empresa

CUADRO 2-1

La española. Características de la empresa, 2014

Nombre o razón social	Industria de alimentos de fiambres y embutidos “La Española”
Registro en FUNDEMPRESA	00005858
Clasificación CIU	3111-Matanza de ganado y preparación de carne
Actividad Principal	Elaboración de productos cárnicos

Fuente: (<http://www.fundempresa.org.bo/> 2011)

2.5 Descripción proceso productivo

De acuerdo a varias visitas técnicas realizadas a las instalaciones de La Española, se pudo constatar que el proceso de producción sigue una secuencia de producción que se describe a continuación.

2.5.1 Recepción, inspección y almacenamiento de materia prima

Para la zona de descarga del camión, se cuenta con personal destinado a realizar el trabajo, además de evaluar la calidad de la materia prima tomando en cuenta parámetros como el PH y la proporción de grasa, luego de efectuar el control de calidad y el control de peso las carnes se refrigeran, quedando almacenados hasta que puedan ser empleados.

2.5.2 Selección y troceado

La materia prima principal que es la carne de cerdo, res y pollo es seleccionada, donde se la separa de huesos, grasa, cuero y nervios, para posteriormente ser troceada ahí se corta la carne de cerdo en pedazos pequeños y se la almacene en contenedores pequeños.

2.5.3 Molido y dosificado de especias

Se introduce la carne troceada en la tolva de la moladora, se dosifica las especias, condimentos y aditivos en los recipientes donde se encuentra la carne molida

2.5.4 Mezclado

Las diferentes carnes son transportadas hasta la cutter, esta máquina realiza la mezcla homogénea de las carnes con las especias, condimentos y aditivos, en el proceso se generan altas temperaturas que puede llegar a descomponer la carne, por lo que durante el proceso se va dosificando pequeñas cantidades de hielo para que esto no ocurra, al finalizar el proceso se obtiene una pasta de carne.

2.5.5 Embutido

Se lleva la pasta de carne hacia la embudidora y se la introduce en su tolva para posteriormente embutir la carne en las tripas, los calibres de las tripas son especificadas en la orden de producción remitida por el jefe de producción. Una vez embutida la carne se procede a un atado en forma manual que da la forma característica al producto.

2.5.6 Secado y ahumado

Se introduce la carne embutida y atada en los hornos para reducir la cantidad de agua que hasta el momento tiene, esto hace que la pasta de carne madure y obtenga su sabor y olor.

2.5.7 Cocido y enfriado

La carne embutida y ahumada es sumergida en agua caliente en un tanque, obteniendo una carne pre cocida, después se introduce esta carne en otro tanque lleno de agua fría, para bajar la temperatura del producto de manera gradual.

2.5.8 Empaquetado

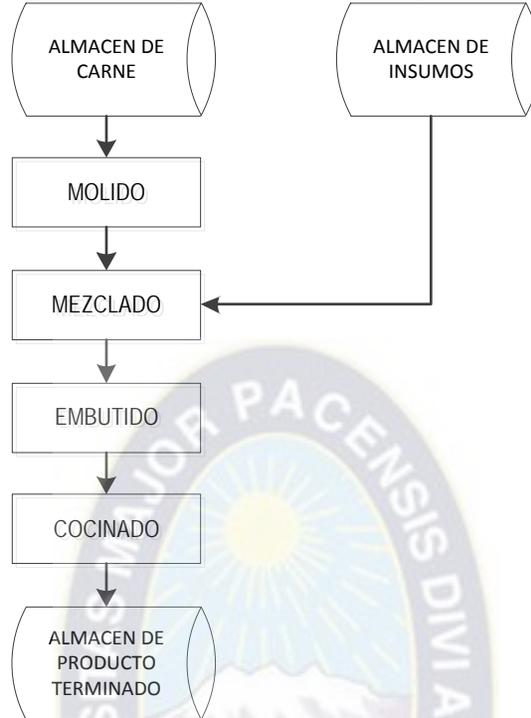
Se introduce la carne embutida pre cocido y frío en la empaquetadora donde se sella el producto en contenedores plásticos y la misma realiza la codificación del producto.

2.5.9 Almacenado

El producto final es almacenado en un refrigerador industrial a una temperatura de -1°C y listo para ser despachado.

DIAGRAMA 2-1

La española. Diagrama de flujo del proceso, 2014



Fuente: Elaborado con base en (Mendoza Gutierrez 2014)

2.6 Maquinaria

La maquinaria con la cuenta la empresa esta detallada en el siguiente cuadro:

CUADRO 2-2

La española. maquinaria y equipo, 2014

Cantidad	Maquinaria o Equipo	Marca	Observaciones
1	Moedor de carne	Rex Maschinen	0.5-11 KW
2	Mezcladora Cutter	Alexanderwerk (220 POL)	Cap – 120 L
1	Embutidora de salchichas	Alexanderwerk	Cap – 40-50 Kg
1	Embutidora para mortadela	Alexanderwerk	Cap – 150 Kg
1	Masajeadora para jamón	Rawer	Cap – 50 L
5	Hornos para secado	Nacional	Cap – 120Kg

Fuente: Elaboración con base en visitas realizadas a la planta de elaboración.

Análisis de flujos de materiales

El flujo de materiales se detalla en los siguientes cuadros.

CUADRO 2-3

La española. Consumo de agua – promedio anual, 2008-2012

Proceso y Operación	Agua utilizada m3	Perdida %	Agua residual m3
Proceso de producción y limpieza de planta	5161,79	85	4387,52
Evaporación	538,34	100	0,00
Sistema sanitario	440,26	100	440,26
Uso domestico	67,87	100	67,87
CONSUMO TOTAL	6208,25		4895,65

Fuente: Elaboración con base a información proporcionada por la empresa

CUADRO 2-4

La española. Consumo de energía eléctrica, 2012

Mes 2012	Consumo Kw/h
Ene	3349,6
Feb	3179,2
Mar	3670,0
Abr	3428,8
May	3240,9
Jun	4248,3
Jul	3318,7
Ago	3430,7
Sep	3787,1
Oct	3606,7
Nov	3320,3
Dic	3129,2

Fuente: Elaboración con base a información proporcionada por la empresa.

CUADRO 2-5

la española: consumo de energía eléctrica, 2008-2012

ÁREA	%	Kw/h
Producción – Planta	93	38789,84
Administración	7	2919,66
Total Consumo	100	41709,5

Fuente: Elaboración con base a información proporcionada por la empresa

CUADRO 2-6

La española: consumo de combustible, 2012

Proceso	Maquinaria	Cantidad	GLP (Garrafas/día)	GLP (Garrafas/año)
Secado y ahumado	Hornos	5	5	1400
Cocimiento	Ollas de cocimiento	9	9	2520
TOTAL		14	3920	

Fuente: Elaboración con base a información proporcionada por la empresa

2.7 Productos

Una forma de clasificarlos desde el punto de vista de la práctica de elaboración, reside en referir al estado de la carne al incorporarse al producto.

2.7.1 Embutidos crudos.

Son aquellos productos elaborados con carnes y grasa crudos, sometidos a un ahumado o maduración. Por ejemplo chorizo, salames.

2.7.2 Embutidos escaldados.

Son aquellos cuya pasta finamente triturada es incorporada cruda, sufriendo el tratamiento térmico (cocción), luego de ser embutidos. Por ejemplo mortadelas, salchichas, jamones, etc.

2.7.3 Embutidos cocidos.

Son aquellos cuando la totalidad de la pasta o parte de ella se cuece antes de incorporarla a la masa. Por ejemplo queso de choncho, etc.

2.7.4 Carnes curadas.

Son aquellas que reciben un tratamiento especial, o sea el curado, que consiste en someter en salmuera, esto permite mejorar la capacidad de conservación, por un tiempo prolongado. Por ejemplo; jamón crudo, tocino, etc.

2.7.5 Productos.

Los productos que ofrece La Española a sus clientes son:

- Salchicha especial con/ sin piel
- Salchicha de coctel
- Salchicha especial de 21 cm
- Chorizo de freír
- Chorizo coctel

- Mortadela liza
- Mortadela cazador
- Mortadela Jamonada
- Mortadela con cuero
- Queso de chanco
- Pasta de hígado
- Jamón
- Salame
- Chorizo español
- Tocino

2.8 Insumos

La empresa emplea para la fabricación de productos principalmente carne fresca en sus diferentes cortes y anchuras, tanto de ganado vacuno, porcino y carne de pollo. También emplea diferentes sustancia para mejorar la presentación y sabor del producto final.

Las materias primas y otras sustancias alimenticias utilizadas para la elaboración de cárnicos se pueden clasificar en:

- a. Carne vacuna
- b. Carne de cerdo
- c. Carne de pollo

La carne es el tejido muscular de los animales, para la alimentación se la utiliza en forma directa o procesada.

La calidad organoléptica de la carne engloba dos aspectos: La calidad del músculo y la calidad de la grasa. Los criterios de importancia comercial de calidad del músculo son: PH, ternera, color, capacidad de retención de agua (CRA), textura y contenido en grasa intramuscular. En la calidad de la grasa los criterios son: consistencia, sabor, aroma y color.

El sabor y la textura de la carne depende exclusivamente de las condiciones ambientales en al que el animal se ha desarrollado, también del tipo de alimentación, sexo y edad.

Las materias primas generalmente la adquieren de la ciudad de Santa Cruz. La empresa adquiere también carne por corte especial ya que en la actualidad es lo más conveniente debido a que el resto del animal no es necesario.

2.8.1 Materia prima o materiales de elaboración.

Los insumos que se utilizan en el proceso de producción son:

2.8.2 Tripas.

Las tripas que utiliza “La Española” son tripas naturales como también tripas sintéticas llamadas colágeno. Las tripas naturales son de ganado ovino y de res. La española las adquiere del mercado local o de proveedores conocidos., son económicas y dan un aspecto artesanal. Las tripas sintéticas tienen la ventaja de que se las puede conservar por periodos largos, tienen un calibre uniforme, son resistentes al ataque bacteriano, por su impermeabilidad se tienen cero merma, se pueden poner el logotipo de la empresa, se utiliza en procesos automáticos, no son tóxicas, las tripas sintéticas de colágeno son comestibles, facilitan el pelado.

2.8.3 Sustancias curantes.

Las sustancias curantes son aquellas que causan alteraciones positivas en la carne, tanto para mejorar su sabor, color, aroma y consistencia. Debido a la capacidad fijadora que tienen estas sustancias mejora el rendimiento de la carne respecto a su peso. Estas sustancias pueden ser: Sal común, Nitratos y Nitritos, Aglutinantes y Ablandadores y Sustancias Aromatizadores. La sal frena el desarrollo de algunos microorganismos, disminuyendo la cantidad de agua disponible para su multiplicación.

2.8.4 Nitratos y nitritos.

El nitrito es el responsable del enrojecimiento, ya que se reduce a NO, que reacciona con la mioglobina para dar nitrosomioglobina. Al añadir nitrito a un producto cárnico se genera un aroma típico, debido a la reacción del nitrito con algunos componentes de la carne.

2.8.5 Aglutinantes y ablandadores.

Se utiliza estos productos ya que permiten la homogenización y ligazón de los productos cárnicos. Estos pueden ser harinas, soja, leche y gelificantes. La harina que se utiliza es el almidón ya que este permite geles, especialmente la amilosa y el amilopectina. La soja da la posibilidad de aumentar el contenido proteico de los productos cárnicos, tanto la soja como sus derivados forman una dispersión que favorece la retención de agua, aumenta la capacidad emulsionante, son ligantes y son muy solubles. La leche se utiliza en la fabricación de embutidos crudos, ya que permite la maduración y a la vez es un agente ligante.

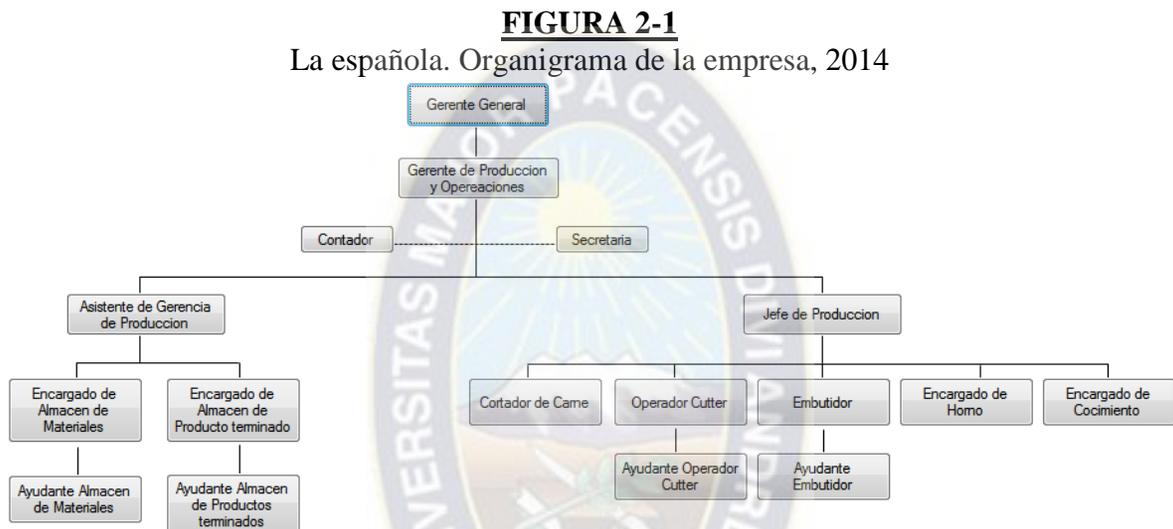
2.8.6 Carrageninas.

Las Carrageninas son utilizadas en la producción de pasta de hígado para las salmueras también se las adquiere del exterior.

2.8.7 Conservantes.

Son los que aumentan el tiempo de vida de los productos almacenados todos estos son a base de potasio.

2.9 Estructura organizacional de la empresa.



Fuente: Elaborado en base a (Mendoza Gutierrez 2014)

CAPÍTULO 3. SISTEMA DE INFORMACIÓN

3.1 Introducción

El sistema de información tiene dos componentes principales: la base de datos y la aplicación de interfaz usuario.

3.1.1 Definiciones

Base de datos, es un conjunto de datos sistematizados según una dimensión la cual alimenta los métodos de alguna aplicación.

Información, los datos que fueron transformados a un contexto significativo.

Cubo multidimensional.- La teoría del cubo multidimensional propone que si se tienen más dimensiones es posible ser más precisos. Entonces si tenemos cualquier tabla con n dimensiones tenemos que hacer una transformación lineal a una tabla con dos dimensiones, donde la primera dimensión es la de orden numérico, y la segunda de parámetros. Y la transformación está definida probablemente por un sistema binario (booleano).

$$T: \mathbb{R}^n \rightarrow \mathbb{R}^2 \text{ DEFINIDA POR } T(n) = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

3.2 Bases para la creación de la base de datos

3.2.1 La información

La información es útil para los usuarios finales, consiste en dar a los datos un valor agregado a través de su procesamiento, los atributos de calidad de la información se enfocan en tres dimensiones tiempo contenido y forma.

CUADRO 3-1

Atributos de la información, 2014

DIMENSIÓN	CUALIDADES DE LA INFORMACIÓN
TIEMPO	
Oportunidad	La información debe suministrarse en el momento que sea necesario
Actualidad	La información debe ser reciente al momento de suministrarse
Frecuencia	La información debe suministrarse con la frecuencia que sea necesario
Periodo	La información puede suministrarse sobre periodos pasados, presentes y futuros.
CONTENIDO	
Exactitud	La información debe estar libre de errores
Pertinencia	La información debe estar relacionada con las necesidades de un destinatario específico y para una determinada situación
Integridad	Debe suministrarse toda la información que sea necesaria
Brevidad	Debe proporcionar solo la información que se necesite, cuando se necesite
Alcance	La información puede tener un alcance amplio o estrecho, o un enfoque interno o externo.
Desempeño	La información puede revelar el desempeño, al medir las actividades logradas, el progreso alcanzado o los recursos acumulados
FORMA	
Claridad	La información debe suministrarse en una forma que sea fácil de entender
Detalle	La información puede presentarse en un formato detallado o resumido
Orden	La información se puede ordenar en una secuencia predeterminada
Presentación	La información puede presentarse en forma narrativa, numérica, grafica u otras formas
Medios	La información puede proporcionarse en forma de documentos de papel impresos, presentaciones de video u otros medios.

Fuente: Elaborado en base a (Levaggi 1999)

3.2.2 Base de datos

Es una colección de archivos interrelacionados, es un conjunto de datos pertenecientes a un mismo contexto y almacenados sistemáticamente para su posterior uso, son creados con un sistema gestor de base de datos. El contenido de una base de datos engloba a la información concerniente (almacenadas en archivos) de una organización o empresa, de tal manera que los datos estén disponibles para los usuarios, una finalidad de la base de datos es eliminar la redundancia de la información o al menos minimizarla. Los tres componentes principales de un sistema de base de datos son el hardware, el software y los datos a manejar así como el personal encargado de manejo del sistema.

3.2.3 Estructura de una base de datos

Las bases de datos están compuestas de datos y metadatos. Los datos son un conjunto de caracteres con algún significado y los metadatos son datos que sirven para especificar la estructura de la base de datos. De este modo se producen dos visiones de la base de datos.

- **Estructura Lógica.** Indica la composición y distribución teórica de la base de datos, la estructura lógica sirve para que las aplicaciones puedan utilizar los elementos de la base de datos sin saber realmente como están almacenados. Sus elementos son objetos, entidades, nodos, relaciones y enlaces que realmente no tienen presencia real en la física del sistema por ello para acceder a los datos tiene que haber una posibilidad de traducir la estructura lógica en una estructura física.
- **Estructura física.** Es la estructura de los datos tal cual se almacenan en las unidades de disco. La correspondencia entre la estructura lógica y la física se almacena en la base de datos.

3.2.4 Ventajas y desventajas de utilizar bases de datos

Ventajas

- Menor redundancia de la información
- Mayor seguridad en los datos
- Se tienen datos más documentados
- Acceso a los datos más eficiente
- Menor espacio de almacenamiento

Desventajas

- Instalación costosa
- Requiere personal calificado
- Implantación larga y difícil

3.2.5 Gestor de base de datos

Un sistema gestor de base de datos es la herramienta que permite a los usuarios procesar, describir, administrar, y recuperar los datos almacenados en una base de datos. El éxito del SGBD reside en mantener la seguridad integridad de los datos también debe proporcionar varias herramientas a los distintos usuarios como:

- Herramientas para la creación de datos así como la estructura de la base de datos

- Herramientas para administrar y crear la estructura física requerida en las unidades de almacenamiento.
- Herramientas para la manipulación de datos para añadir modificar o suprimir consultas de datos.
- Herramientas para recuperación de datos en caso de desastre.
- Herramientas para la creación de copias de seguridad.
- Herramientas para la gestión de la comunicación de la base de datos.

Para gestionar la base de datos vamos a realizar un pequeño análisis empírico y abstracto, sin base científica, para elegir el sistema de gestor de base de datos que vamos a utilizar. Los programas que posiblemente nos puedan ayudar son: MS SQL Server 2012, MS Office Access 2013, Oracle Data Warehouse 2010. De manera anticipada podemos mencionar que el mejor gestor elegido por los autores es MS SQL Server 2012, pero vamos a confirmarlo con un pequeño análisis de aptitudes.

CUADRO 3-2

Aptitudes de los gestores, 2014

APTITUD	MS SQL SERVER 2012	MS OFFICE ACCESS 2013	ORACLE DATA WAREHOUSE 2010
CREAR MOTOR DE BASE DE DATOS	X	X	X
LEGUAJE PROPIO PARA EFECTUAR CONSULTAS	X		X
MEJOR COMPATIBILIDAD CON MS VISUAL STUDIO 2012	X		
SERVICIO DE ANÁLISIS DE DATOS	X		
SERVICIO DE REPORTES	X		X
COMPATIBLE CON ALGORITMOS DE CONEXIÓN	X	X	X
TOTAL	6	2	4

Fuente: Elaborado en base a experiencia de uso de los gestores

Tal como lo mencionamos anticipadamente, el gestor elegido es MS SQL SERVER 2012. Ahora el dilema está en aprender a usar el gestor, lo cual es tarea de los autores.

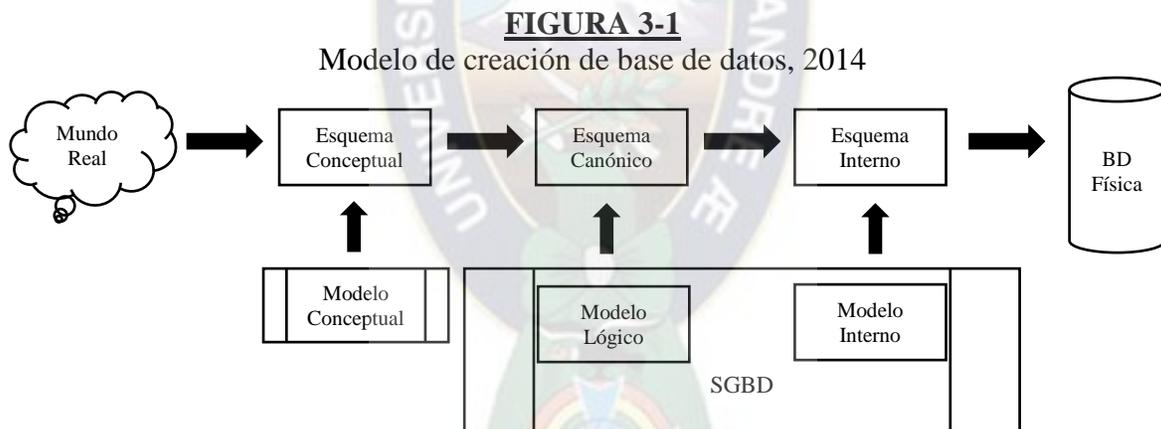
El gestor de base de datos tiene un sistema de motor de base de datos, servicio de análisis de datos (minería de datos) y servicio de reporte de datos (consulta). En este capítulo solo utilizaremos la base de datos. También el gestor tiene un sistema de protección de la información la cual vamos a utilizar por requerimiento de la empresa.

El servicio de análisis es deficiente para todos los gestores de base de datos, las funciones predeterminadas son básicas y no cumplen con los requerimientos que se tiene en una especialidad, es complejo codificar un análisis que cumpla con requerimientos específicos. Explicado de mejor manera si tengo una base de datos con datos de la demanda de los últimos años, es sumamente complejo elaborar un procedimiento que pueda pronosticar, si se tienen los parámetros es posible, pero si no se tienen los parámetros es prácticamente imposible crear un procedimiento heurístico, lo cual es necesarios para resolver un modelo aditivo de (Holt y Winters) por ejemplo.

3.2.6 Modelado de datos

La finalidad de los modelos es la de simbolizar una parte del mundo real de forma que sea más fácil de manipular. Es un esquema mental en el que se intenta reproducir las características de una realidad específica.

El caso de los modelos de datos lo que intenta reproducir es una información real que deseamos almacenar en un sistema informático y se representa de la siguiente forma:

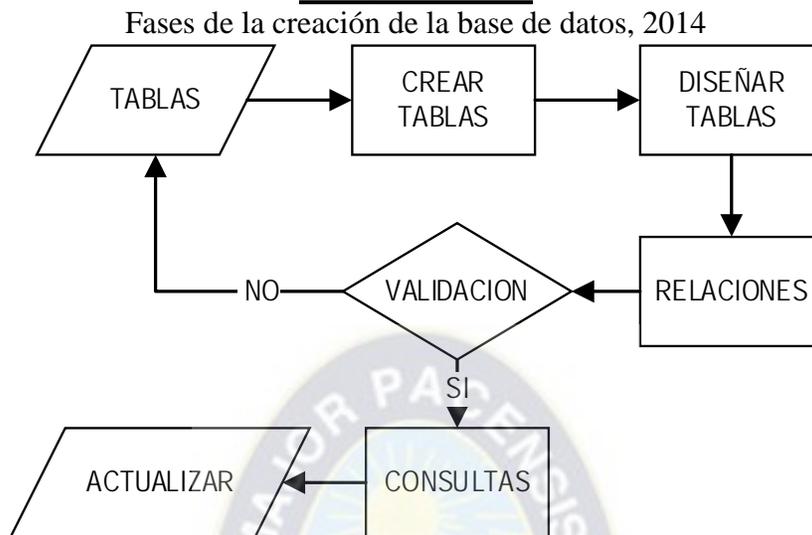


Fuente: Elaborado en base a (Microsoft SQL Server 2012) y (Levaggi 1999)

- **Mundo real.** Contiene l información tal cual la percibimos, es el punto de partida.
- **Esquema conceptual.** Representa el modelo de datos de forma independiente del sistema gestor de base de datos que se utilice.
- **Esquema canónico.** Representa los datos en un formato más cercano al del ordenador
- **Esquema interno.** Representa los datos según el modelo concreto de un sistema gestor de base de datos.
- **Base de datos física.** Son los datos tal cual almacenados en disco.

3.3 Fases para la creación de base de datos

DIAGRAMA 3-1



Fuente: Elaboración en base al tutorial interactivo (Video2Brain 2013)

3.3.1 Definición

Sistema de información, es un conjunto de elementos de datos procesados que ejecutan métodos específicos para lograr un objetivo.

3.3.2 Alcance

Las tablas tienen las perspectivas del planteamiento del problema. Para la perspectiva financiera las tablas son: Activos fijos, Contabilidad, Facturas, Compras y Ventas. Para la perspectiva de mercado las tablas son: Clientes, Ventas, Proveedores, Compras, Precio y Demanda. Para la perspectiva de personal las tablas son: Empleados y Contabilidad. Para la perspectiva de procesos las tablas son: Activos fijos, Demanda, Insumos, Productos y Empleados.

Como se podrá comprender hay tablas que están dentro de más de una perspectiva, esto se normaliza con las relaciones de las tablas.

3.4 Tablas

En el alcance ya definimos las tablas, ahora los vamos a validar con el procedimiento del bucle de las fases.

En los siguientes subtítulos vamos a usar formularios, las copias en imagen digitalizada del formulario original están en el trabajo principal y los formularios para llenar están la parte de ANEXO B.

3.4.1 Formulación de las tablas

Para formular las tablas necesitamos establecer los requerimientos que tiene la empresa para establecer la consultoría. Estos requerimientos los introducimos a través de un formulario de formulación de tablas. Los formularios que vamos a establecer son creados en la aplicación MS Office InfoPath 2013.⁷



⁷ Aquí se genera un gran problema, el programa MS Office InfoPath 2013 no es compatible con la impresora PDF, XPS y OneNote 2013; pero si lo es con el formato Web mht, entonces lo que se hizo para resolver el problema es convertir al formato Web mht desde MS Office InfoPath Filler 2013 y luego de allí importar con OneNote 2013 como impresión de archivo y así obtenemos nuestro formulario en formato de imagen.

FIGURA 3-2

La española. Formulario para la formulación de tablas, 2014



FORMULACION DE TABLAS

1. DATOS BASICOS

EMPRESA:

SECTOR: CIU:

2. REQUERIMIENTOS GENERALES

TABLA	PERSPECTIVAS	APLICA
1. Activos Fijos	Financiera, Procesos	<input checked="" type="checkbox"/>
2. Clientes	Financiera, Mercado	<input checked="" type="checkbox"/>
3. Compras	Financiera, Mercado	<input checked="" type="checkbox"/>
4. Cuentas	Financiera, Personal	<input checked="" type="checkbox"/>
5. Demanda	Mercado, Procesos	<input checked="" type="checkbox"/>
6. Empleados	Personal, Procesos	<input checked="" type="checkbox"/>
7. Facturas	Financiera	<input checked="" type="checkbox"/>
8. Insumos	Procesos	<input checked="" type="checkbox"/>
9. Precio	Mercado	<input checked="" type="checkbox"/>
10. Proceso	Procesos	<input checked="" type="checkbox"/>
11. Productos	Procesos	<input checked="" type="checkbox"/>
12. Proveedores	Mercado	<input checked="" type="checkbox"/>
13. Sueldos	Financiera, Procesos	<input checked="" type="checkbox"/>
14. Transacciones	Financiera	<input checked="" type="checkbox"/>
15. Ventas	Financiera, Mercado	<input checked="" type="checkbox"/>

3. REQUERIMIENTOS ESPECIFICOS

TABLA	PERSPECTIVAS	APLICA
APROBACION	<input type="text" value="03/05/2014"/>	<input type="text" value="03/05/2014"/>
	CONSULTOR	RESPONSABLE EMPRESA

Fuente: Elaborado en (Microsoft InfoPath Design 2013) en base a Gerencia General y experiencia de los autores

El formulario de formulación de tablas está diseñado en tres secciones. La primera sección es dedicada el título, sus componentes son la fecha de llenado, nombre del formulario y logo de la empresa. En la segunda sección tenemos tres subsecciones la primera está

dedicada a los datos generales de la empresa, sus componentes son nombre o razón social como “empresa”, el sector y la clasificación CIIU con seis dígitos; la siguiente subsección está dedicada a las tablas que aplican según el consultor con una columna para la aplicación de la perspectiva y otra para validar por parte de la empresa; y la tercera subsección está dedicada a los requerimientos específicos que tenga la empresa para la formulación de tablas siendo el validador en esta ocasión el consultor. La tercera sección está dedicada a la validación quirográfica de las partes, en esta sección se tiene la rúbrica de las partes con la fecha y el nombre del responsable.

Ya tenemos bien definidas nuestras tablas, entonces tenemos que diseñarlas. Ahora vamos a ejecutar el programa MS SQL Server 2012 con la cuenta de servidor ya creada, pulsamos el botón opuesto del ratón en la pestaña “databases” y seleccionamos la opción “new database”. Luego le asignamos el nombre de la empresa “LA ESPAÑOLA” e inmediatamente el programa nos crea la base de datos, pero si no ocurre así pulsamos el mouse sobre el botón actualizar. Y se actualizan las bases de datos.⁸

Luego de creada la base de datos creamos nuestras tablas una por una. Expandimos nuestra base de datos creada y en la pestaña tablas pulsamos el botón opuesto del ratón y seleccionamos la opción “new table”. Luego de eso en la ventana central del programa nos aparece una tabla de diseño donde vamos a introducir el nombre de los campos, el tipo de dato del campo, la permisión de vacíos de los campos y la llave principal que en todos los casos es el primer campo.

Ahora vamos a establecer los requerimientos para las tablas ya elegidas. Estos requerimientos los introducimos a través de un formulario de diseño de tablas. Los formularios nuevamente son creados en el programa MS Office InfoPath 2013.

⁸ Es importante no alterar las otras bases de datos del programa ya que esas bases de datos alimentan el sistema operativo.



01/01/2014

DISEÑO DE TABLAS

1. DATOS BASICOS

Empresa: Embutidos "La Española" S.R.L.

Sector: Procesamiento y conservación de carne

CIIU: 1010

2. REQUERIMIENTOS GENERALES

Clientes		T	N	A	Producto		T	N	A	Ventas		T	N	A	Compras		T	N	A	Proveedores		T	N	A	Empleados		T	N	A	Facturas		T	N	A															
Código	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_producto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_venta	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_compra	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_prov	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_empleado	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_factura	<input checked="" type="checkbox"/>																		
Empresa	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Description	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Fecha_pedido	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Fecha_pedido	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Empresa	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Nombre	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Fecha	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Import_Ped	<input checked="" type="checkbox"/>													
Dirección	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Precio	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Fecha_entrega	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Fecha_entrega	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Dirección	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Dirección	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_cliente	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_cliente	<input checked="" type="checkbox"/>													
Teléfono	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Ficha_tecnica	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cant_pedv	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cant_pedc	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Teléfono	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Teléfono	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_venta	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_venta	<input checked="" type="checkbox"/>													
CL_NIT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Existencia_fisica	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Description	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Description	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	CL_NIT	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cargo	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_venta	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_venta	<input checked="" type="checkbox"/>													
Cod_empleado	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Saldo_inyv	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	precio	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	precio	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_empleado	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Departamento	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_prov	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_prov	<input checked="" type="checkbox"/>													
Saldo_cpc	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Import_pedv	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	import_pedc	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Saldo_cpp	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Saldo_cpc	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Fecha_Contr	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_compra	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_compra	<input checked="" type="checkbox"/>													
Saldo_cliente	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_cliente	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_producto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Saldo_prov	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Saldo_prov	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Salario_basico	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_compra	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_compra	<input checked="" type="checkbox"/>													
Cons_final	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_producto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_insumo	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_insumo	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_insumo	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Foto	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_compra	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Cod_compra	<input checked="" type="checkbox"/>													
	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	Personal_fijo	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

3. REQUERIMIENTOS ESPECIFICOS

Nombre de tabla
Nombre de campo

LEYENDA

- I: INTEGER
- S: NVARCHAR
- M: MONEY
- D: DATE
- B: BIT
- O: IMAGE
- R: REAL

APROBACION

01/01/2014

01/01/2014

CONSULTOR

RESPONSABLE EMPRESA

Fuente: Elaborado en (Microsoft InfoPath Design 2013) en base a Gerencia General y experiencia de los autores

El formulario diseño de tablas está diseñado de manera en tres secciones. La primera sección es dedicada el título, sus componentes son la fecha de llenado, nombre del formulario y logo de la empresa. En la segunda sección tenemos tres subsecciones la primera está dedicada a los datos generales de la empresa, sus componentes son nombre o razón social como “empresa”, el sector y la clasificación CIU con seis dígitos; la siguiente subsección está dedicada al diseño de tablas que aplican según el consultor con una columna para el nombre, para el tipo de dato, para la aprobación de nullos y otra para validar por parte de la empresa; y la tercera subsección está dedicada a los requerimientos específicos que tenga la empresa para la formulación de tablas siendo el validador en esta ocasión el consultor. La tercera sección está dedicada a la validación quirografía de las partes, en esta sección se tiene la rúbrica de las partes con la fecha y el nombre del responsable.

Como podemos ver los dos formularios han sido validados por las dos partes. También cabe recalcar que no se llenó la sección de requerimientos específicos, esto puede ser debido a que la empresa tiene un conocimiento reducido de los sistemas de información antes de efectuar la consultoría o que los consultores son muy acertados en la concepción del sistema de información. En caso de que no hubieran validado las partes el tendríamos que efectuar una nueva corrida al bucle para optimizar el sistema. Ahora se revisará el sistema cada dos años para verificar que el sistema siga en óptimo.⁹

3.5 Relaciones

Las relaciones se explican a continuación.

3.5.1 Definiciones

Llave principal, es una columna o combinación de columnas que contiene valores que identifican cada fila en la tabla.

Llave foránea, es una columna o combinación de columnas que se usa para establecer e implementar un vínculo en los datos de dos tablas diferentes.

3.5.2 Establecer relaciones

Para establecer las relaciones debemos recurrir a la formulación de tablas, al formulario de diseño de tablas. Para fines prácticos vamos a definir la primera columna como llave

⁹ Vamos a aclarar la razón por la que hacemos más de un formulario para todas las tablas, la razón es sencilla en el diagrama de fases definimos los procesos de manera secuencial, primero tenemos que definir las tablas luego las diseñamos, no podemos diseñar una tabla sin que este definida o sin que esta no haya sido aprobada Lo correcto es hacer un formulario para cada etapa del proceso.

principal. Y las llaves foráneas son las columnas de llave principal en otras tablas. La manera de representar las relaciones para los programas de gestión de base de datos es un gráfico.

Modelo entidad relación (E/R)

Se trata de un modelo que sirve para crear esquemas conceptuales de bases de datos, con el transcurso del tiempo se volvió un estándar para realizar esta tarea. Se llama modelo E/R justamente porque crea asociaciones entre objetos o elementos (real o abstracto) que es lo que se busca en un esquema conceptual.

Ahora vamos a graficar las relaciones de la base de datos usando la plantilla de (Microsoft Office Visio 2010).

Esta plantilla nos da las formas básicas para elaborar un diagrama de entidad relación.¹⁰

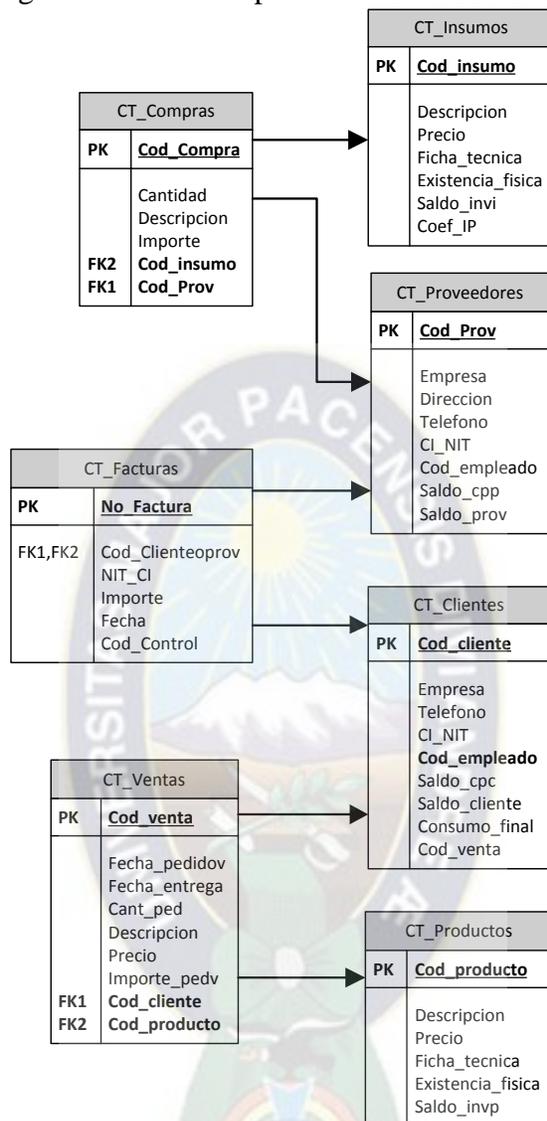
Siguiendo la perspectiva de mercado el sistema de transacciones tiene el siguiente diagrama de relaciones.



¹⁰ El gestor de base de datos nos permite elaborar un diagrama de la base de datos en la pestaña server\databases\database diagram\ hacer click derecho y elegir la opción new database diagram, pero este diagrama no es posible transformar en una imagen es por eso que usamos la plantilla de MS Office Visio 2010.

FIGURA 3-4

SIS-K DB. Diagrama de relación para el sistema de transacciones, 2014

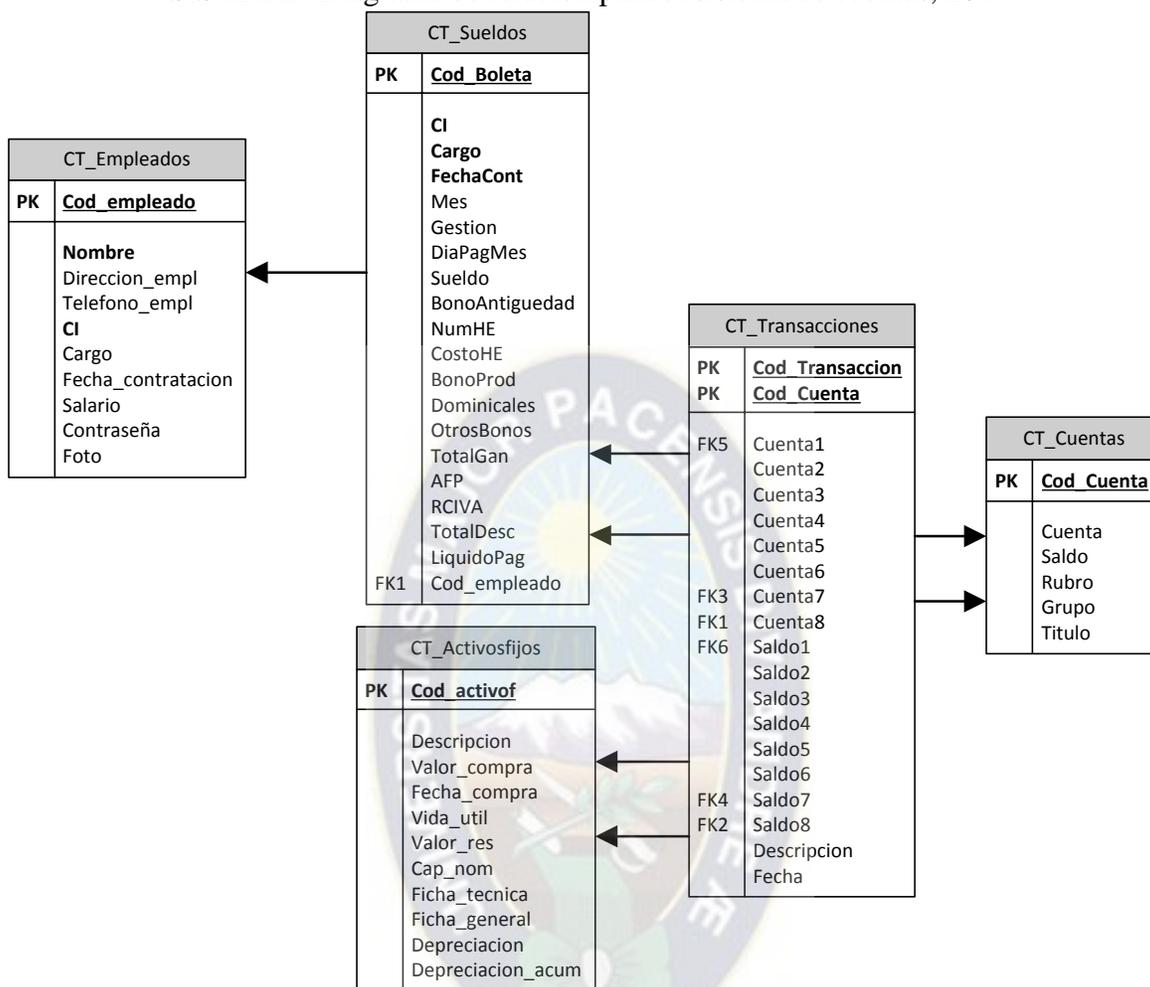


Fuente: Elaborado en la plantilla database diagram de (Microsoft Office Visio 2010) en base a la FIGURA 3-3

Siguiendo la perspectiva financiera el sistema de gestión de cuentas tiene el siguiente diagrama de relaciones, considerando que las cuentas que no tienen foránea, son cuentas para otras transacciones.

FIGURA 3-5

SIS-K DB. Diagrama de relación para el sistema de cuentas, 2014



Fuente: Elaborado en la plantilla database diagram de (Microsoft Office Visio 2010) En base al FIGURA 3-1

Siguiendo la perspectiva de procesos no hemos creado ningún diagrama de relaciones, esta perspectiva requiere de análisis de datos. El gestor de base de datos tiene el servicio de análisis, este servicio de análisis contiene muy escasas funciones, por lo que no vamos a usar este gestor para el servicio de base de datos y el servicio de análisis de datos.

Para este procesos de análisis de datos podemos tomar dos caminos: el primero es crear la base de datos en MS SQL Server 2012, modelar nuestros propios métodos y conectarlos con consultas; el segundo es migrar todo el sistema de producción a MS Office Excel 2013, Diseñar los datos y parámetros del sistema, analizar en el complemento SOLVER de MS

Office Excel 2013 ejecutando por debajo del sistema el proceso de análisis y conectar con vista de los datos.

En el sistema de producción se tienen dos análisis principales: el primero es el análisis de la programación de la producción, según el alcance de la empresa lo vamos a formular como un modelo de programación lineal; el segundo es el análisis del modelo de pronóstico de la demanda, según el alcance de la empresa lo vamos a formular como un modelo no causal, que se analiza mediante una simulación de Gradiente conjugado o Evolucionista, por la complejidad del modelo vamos a utilizar la simulación Evolucionista.¹¹

Ahora mencionado lo anterior vamos a optar por la segunda opción sin dejar de lado la primera opción para evolucionar en el desarrollo del sistema. Este capítulo solo es de modelación de base de datos, por tanto no podemos incluir métodos en este capítulo. Para no descuidar el sistema de producción lo vamos a retomar el diseño del sistema en el **CAPÍTULO 12. DISEÑO DE LA INTERFAZ.**

3.6 Consultas

Son solicitudes específicas de filtrado que se le hace a la base de datos, la cual devuelve la solicitud en un informe para analizar de mejor manera la información.

3.6.1 Tipos de consulta

Las consultas pueden ser:

- De orden, esta consulta da un orden o criterio de orden a las tablas consultadas.
- De filtro, esta consulta filtra las tablas consultadas según un o varios criterios.

3.6.2 Como efectuar una consulta en MS SQL Server

MS SQL Server utiliza el lenguaje “Transact-SQL”, para efectuar un consulta primero debemos seleccionar la tablas o tablas requeridas. Para esto debemos escribir SELECT seguido del nombre de los campos en el orden que se requiere y si es para todos los campos

¹¹ Aquí vamos a explicar brevemente los métodos de resolución de problemas de SOLVER de MS Office Excel 2013. El primer método es el algoritmo SIMPLEX que aparece como “SIMPLEX LP”, este método solo es aplicable a la programación lineal, el método itera siguiendo los puntos factibles de un espacio de soluciones. El segundo es la simulación del GRADIENTE CONJUGADO que aparece como “GRG Nonlinear”, este método aplica a todo los modelos matemáticos de programación, el método itera entre los puntos de una red de espacios perfectamente separados de un espacio de posibles soluciones, mientras más discreción se tenga entre los puntos de la red más precisa será la simulación, lo cual también es una limitante del método ya que mientras más puntos se tienen en la red más tiempo de ejecución requeriremos para resolver el problema. El tercero es el método EVOLUCIONISTA que aparece como “Evolutionary”, este método aplica a todos los modelos matemáticos de programación, el método itera siguiendo un crecimiento en el objetivo. Este proceso requiere de menor tiempo de ejecución que el anterior por lo cual es el más eficiente.

podemos simplificar con *, por ejemplo, SELECT id, precio, producto. Luego en la siguiente línea escribimos FROM seguido de la tabla a la que nos referimos, por ejemplo, FROM DBO.PRODUCTOS. Para ordenar la petición en la siguiente línea escribimos ORDER BY seguido del nombre del campo, esto nos da un orden ascendente por defecto, para conseguir el orden descendente debemos escribir DESC después de lo último, por ejemplo, ORDER BY producto DESC. Para filtrar la petición en la siguiente línea escribimos WHERE entre paréntesis el criterio de filtrado, por ejemplo, WHERE (Precio<100).

Para terminar con este ejercicio vamos a mostrar la consulta completa y el reporte respectivo como ejemplo desde luego.

FIGURA 3-6

La española. Ejemplo de código y reporte, 2014

```
SELECT TOP (100) PERCENT Cod_producto AS id, Descripción AS Producto, Precio
FROM dbo.CT_Productos
WHERE (Precio > 30)
ORDER BY Precio DESC
```

	id	Producto	Precio
▶	205	Salame	49,0000
	204	Jamon	46,0000
	203	Mortadela	41,0000
	202	Chorizo	40,0000
	201	Salchicha viena	35,7000
*	NULL	NULL	NULL

Fuente: Elaborado en (Microsoft SQL Server 2012) en base al tutorial interactivo de (Video2Brain 2013)

Este gráfico es elaborado en el gestor elegido, en la parte superior está el código escrito por el usuario y en la parte inferior es el reporte que presenta el programa.

Este lenguaje es de gran ventaja respecto a otros gestores, ya que se pueden hacer un sinnúmero de peticiones de lógica booleana.

3.6.3 Establecer consultas

Para establecer las consultas tenemos que referirnos al sistema informático y a los métodos que vamos a utilizar, por ahora no vamos a poder realizar todas las consultas, vamos a volver al tema cuando tengamos bien definido nuestros métodos los cuales tienen los requerimientos específicos de nuestra base de datos. Esto no afecta la concepción de nuestra base de datos.

3.7 Actualización de datos

Es la transformación de los datos reales en datos físicos. Para proteger la información de la empresa de Embutidos La Española. Hemos decidido conjuntamente publicar parcialmente los datos de la empresa. Los cuales se muestran en la parte de ANEXO D.

3.8 Manual SQL Server 2012

3.8.1 Introducción

El lenguaje de consulta estructurado (SQL) es un lenguaje de base de datos normalizado, utilizado por los diferentes motores de bases de datos para realizar determinadas operaciones sobre los datos o sobre la estructura de los mismos.

Pero como sucede con cualquier sistema de normalización hay excepciones para casi todo; de hecho, cada motor de bases de datos tiene sus peculiaridades y lo hace diferente de otro motor, por lo tanto, el lenguaje SQL normalizado (ANSI) no nos servirá para resolver todos los problemas, aunque si se puede asegurar que cualquier sentencia escrita en ANSI será interpretable por cualquier motor de datos.

3.8.2 Componentes del SQL

El lenguaje SQL está compuesto por comandos, cláusulas, operadores y funciones de agregado. Estos elementos se combinan en las instrucciones para crear, actualizar y manipular las bases de datos. Comandos

Existen dos tipos de comandos SQL:

- DDL que permiten crear y definir nuevas bases de datos, campos e índices.
- DML que permiten generar consultas para ordenar, filtrar y extraer datos de la base de datos.

3.8.3 Componentes del SQL (II)

Operadores de Comparación

Operador Uso

< Menor que

> Mayor que

<> Distinto de

<= Menor o igual que

>= Mayor o igual que

= Igual que

BETWEEN Utilizado para especificar un intervalo de valores.

LIKE Utilizado en la comparación de un modelo

IN Utilizado para especificar registros de una base de datos

Funciones de Agregado

Las funciones de agregado se usan dentro de una cláusula *SELECT* en grupos de registros para devolver un único valor que se aplica a un grupo de registros.

CUADRO 3-3

DB SIS-K. Funciones de agregado, 2014

Función	Descripción
<i>AVG</i>	Utilizada para calcular el promedio de los valores de un campo determinado
<i>COUNT</i>	Utilizada para devolver el número de registros de la selección
<i>SUM</i>	Utilizada para devolver la suma de todos los valores de un campo determinado
<i>MAX</i>	Utilizada para devolver el valor más alto de un campo especificado
<i>MIN</i>	Utilizada para devolver el valor más bajo de un campo especificado

Fuente: Elaborado en base a (Video2Brain 2013)

3.8.4 Orden de ejecución de los comandos

Dada una sentencia SQL de selección que incluye todas las posibles cláusulas, el orden de ejecución de las mismas es el siguiente:

- Cláusula FROM
- Cláusula WHERE
- Cláusula GROUP BY
- Cláusula HAVING
- Cláusula SELECT
- Cláusula ORDER BY

3.8.5 Consultas de Selección

Las consultas de selección se utilizan para indicar al motor de datos que devuelva información de las bases de datos, esta información es devuelta en forma de conjunto de registros que se pueden almacenar en un objeto recordset.

Este conjunto de registros puede ser modificable. Consultas básicas

La sintaxis básica de una consulta de selección es la siguiente:

SELECT

Campos

FROM

Tabla

En donde campos es la lista de campos que se deseen recuperar y tabla es el origen de los mismos, por ejemplo:

SELECT

Nombre, Teléfono

FROM

Cientes

Esta sentencia devuelve un conjunto de resultados con el campo nombre y teléfono de la tabla clientes.

En determinadas ocasiones nos puede interesar incluir una columna con un texto fijo en una consulta de selección, por ejemplo, supongamos que tenemos una tabla de empleados y deseamos recuperar las tarifas semanales de los electricistas, podríamos realizar la siguiente consulta:

SELECT

*Empleados.Nombre, 'Tarifa semanal: ', Empleados.TarifaHora * 40*

FROM

Empleados

WHERE

Empleados.Cargo = 'Electricista'

3.8.6 Ordenar los registros

Adicionalmente se puede especificar el orden en que se desean recuperar los registros de las tablas mediante la cláusula ORDER BY Lista de Campos.

En donde Lista de campos representa los campos a ordenar.

Ejemplo:

SELECT

CodigoPostal, Nombre, Telefono

FROM

Cientes

ORDER BY

Nombre

Esta consulta devuelve los campos *CodigoPostal*, *Nombre*, *Telefono* de la tabla *Cientes* ordenados por el campo *Nombre*.

Se pueden ordenar los registros por más de un campo, como por ejemplo:

SELECT

CodigoPostal, Nombre, Telefono

FROM

Cientes

ORDER BY

CodigoPostal, Nombre

Incluso se puede especificar el orden de los registros: ascendente mediante la cláusula (*ASC* - se toma este valor por defecto) o descendente (*DESC*)

SELECT

CodigoPostal, Nombre, Telefono

FROM

Cientes

ORDER BY

CodigoPostal DESC, Nombre ASC

3.8.7 Consultas de Acción

Las consultas de acción son aquellas que no devuelven ningún registro, son las encargadas de acciones como añadir y borrar y modificar registros.

Tanto las sentencias de actualización como las de borrado desencadenarán (según el motor de datos) las actualizaciones en cascada, borrados en cascada, restricciones y valores por defecto definidos para los diferentes campos o tablas afectadas por la consulta. **DELETE**

Crea una consulta de eliminación que elimina los registros de una o más de las tablas listadas en la cláusula *FROM* que satisfagan la cláusula *WHERE*.

Esta consulta elimina los registros completos, no es posible eliminar el contenido de algún campo en concreto. Su sintaxis es: **DELETE FROM Tabla WHERE criterio**

Una vez que se han eliminado los registros utilizando una consulta de borrado, no puede deshacer la operación.

Si desea saber qué registros se eliminarán, primero examine los resultados de una consulta de selección que utilice el mismo criterio y después ejecute la consulta de borrado.

Mantenga copias de seguridad de sus datos en todo momento. Si elimina los registros equivocados podrá recuperarlos desde las copias de seguridad.

DELETE

FROM

Empleados

WHERE

Cargo = 'Vendedor'

3.8.8 Insert into

Agregar un registro en una tabla. Se la conoce como una consulta de datos añadidos.

Esta consulta puede ser de dos tipos: Insertar un único registro o Insertar en una tabla los registros contenidos en otra tabla. Para insertar un único Registró:

En este caso la sintaxis es la siguiente:

INSERT INTO Tabla (campo1, campo2, ..., campoN)

VALUES (valor1, valor2, ..., valorN)

Esta consulta graba en el campo1 el valor1, en el campo2 y valor2 y así sucesivamente.

Para seleccionar registros e insertarlos en una tabla nueva

En este caso la sintaxis es la siguiente:

SELECT campo1, campo2, ..., campoN INTO nuevatabla

FROM tablaorigen [WHERE criterios]

3.8.9 Update

Crea una consulta de actualización que cambia los valores de los campos de una tabla especificada basándose en un criterio específico. Su sintaxis es:

UPDATE Tabla SET Campo1=Valor1, Campo2=Valor2, CampoN=ValorN

WHERE Criterio

UPDATE es especialmente útil cuando se desea cambiar un gran número de registros o cuando éstos se encuentran en múltiples tablas.

Puede cambiar varios campos a la vez.

El ejemplo siguiente incrementa los valores Cantidad pedidos en un 10 por ciento y los valores Transporte en un 3 por ciento para aquellos que se hayan enviado al Reino Unido.:

UPDATE

Pedidos

SET

*Pedido = Pedidos * 1.1,*

*Transporte = Transporte * 1.03*

WHERE

PaisEnvío = 'ES'

UPDATE no genera ningún resultado. Para saber qué registros se van a cambiar, hay que examinar primero el resultado de una consulta de selección que utilice el mismo criterio y después ejecutar la consulta de actualización.

UPDATE

Empleados

SET

Grado = 5

WHERE

Grado = 2

Si en una consulta de actualización suprimimos la cláusula WHERE todos los registros de la tabla señalada serán actualizados.

UPDATE

Empleados

SET

*Salario = Salario * 1.1*

3.8.10 Referencias cruzadas

Referencias Cruzadas ACCESS

Una consulta de referencias cruzadas es aquella que nos permite visualizar los datos en filas y en columnas, estilo tabla, por ejemplo:

CUADRO 3-4

Referencia cruzada ACCESS ,2014

Producto / Año	1996	1997
Pantalones	1.250	3.000
Camisas	8.560	1.253
Zapatos	4.369	2.563

Fuente: Elaborado en base a (Video2Brain 2013)

Si tenemos una tabla de productos y otra tabla de pedidos, podemos visualizar en total de productos pedidos por año para un artículo determinado, tal y como se visualiza en la tabla anterior. La sintaxis para este tipo de consulta es la siguiente:

*TRANSFORM función agregada instrucción select PIVOT campo pivot
[IN (valor1[, valor2[, ...]])]*

En donde:

Función agregada Es una función SQL agregada que opera sobre los datos seleccionados.

Instrucción select Es una instrucción SELECT.

Campo pivot Es el campo o expresión que desea utilizar para crear las cabeceras de la columna en el resultado de la consulta.

valor1, valor2 Son valores fijos utilizados para crear las cabeceras de la columna.

Para resumir datos utilizando una consulta de referencia cruzada, se seleccionan los valores de los campos o expresiones especificadas como cabeceras de columnas de tal forma que pueden verse los datos en un formato más compacto que con una consulta de selección.

3.8.11 Criterios de selección

En el apartado anterior se vio la forma de recuperar los registros de las tablas, las formas empleadas devolvían todos los registros de la mencionada tabla. A lo largo de este apartado se estudiarán las posibilidades de filtrar los registros con el fin de recuperar solamente aquellos que cumplan unas condiciones preestablecidas.

Antes de comenzar el desarrollo de este apartado hay que recalcar tres detalles de vital importancia. El primero de ellos es que cada vez que se desee establecer una condición referida a un campo de texto la condición de búsqueda debe ir encerrada entre comillas simples; la segunda es que no es posible establecer condiciones de búsqueda en los campos memo y; la tercera y última hace referencia a las fechas. A día de hoy no he sido capaz de

encontrar una sintaxis que funcione en todos los sistemas, por lo que se hace necesario particularizarlas según el banco de datos:

CUADRO 3-5

Sintaxis de los criterios de selección, 2014

Banco de Datos	Sintaxis
SQL-SERVER	Fecha = #mm-dd-aaaa#
ORACLE	Fecha = to_date('YYYYDDMM','aaaammdd',)
ACCESS	Fecha = #mm-dd-aaaa#

Fuente: Elaborado en base a (Kort, Silverschatz y Sudarshan 1998)

Ejemplo

CUADRO 3-6

Ejemplo de Sintaxis de los criterios de selección, 2014

Banco de Datos	Ejemplo (para grabar la fecha 18 de mayo de 1969)
SQL-SERVER	Fecha = #05-18-1969# o Fecha = 19690518
ORACLE	Fecha = to_date('YYYYDDMM', '19690518')
ACCESS	Fecha = #05-18-1969#

Fuente: Elaborado en base al CUADRO 3-5

Referente a los valores lógicos True o False cabe destacar que no son reconocidos en ORACLE, ni en este sistema de bases de datos ni en SQL-SERVER existen los campos de tipo "SI/NO" de ACCESS; en estos sistemas se utilizan los campos BIT que permiten almacenar valores de 0 o 1. Internamente, ACCESS, almacena en estos campos valores de 0 o -1, así que todo se complica bastante, pero aprovechando la coincidencia del 0 para los valores FALSE, se puede utilizar la sintaxis siguiente que funciona en todos los casos: si se desea saber si el campo es falso "... CAMPO = 0" y para saber los verdaderos "CAMPO <> 0".

3.8.12 El Operador Like

Se utiliza para comparar una expresión de cadena con un modelo en una expresión SQL. Su sintaxis es:

Expresión Like modelo

En donde expresión es una cadena modelo o campo contra el que se compara expresión. Se puede utilizar el operador Like para encontrar valores en los campos que coincidan con el modelo especificado. Por modelo puede especificar un valor completo (Ana María), o se puede utilizar una cadena de caracteres comodín como los reconocidos por el sistema operativo para encontrar un rango de valores (Like An*).

El operador Like se puede utilizar en una expresión para comparar un valor de un campo con una expresión de cadena. Por ejemplo, si introduce Like C* en una consulta SQL, la consulta devuelve todos los valores de campo que comiencen por la letra C. En una consulta con parámetros, puede hacer que el usuario escriba el modelo que se va a utilizar.

El ejemplo siguiente devuelve los datos que comienzan con la letra P seguido de cualquier letra entre A y F y de tres dígitos:

Like 'P[A-F]###'

Este ejemplo devuelve los campos cuyo contenido empiece con una letra de la A hasta la letra D seguidas de cualquier cadena.

Like '[A-D]'*

En la tabla siguiente se muestra cómo utilizar el operador Like
SQL SERVER 2012

CUADRO 3-7
Operador LIKE, 2014

Ejemplo	Descripción
LIKE 'A%'	Todo lo que comience por A
LIKE '_NG'	Todo lo que comience por cualquier carácter y luego siga NG
LIKE '[AF]%'	Todo lo que comience por A o F
LIKE '[A-F]%'	Todo lo que comience por cualquier letra comprendida entre la A y la F
LIKE '[A^B]%'	Todo lo que comience por A y la segunda letra no sea una B

Fuente: Elaborado en base a (Video2Brain 2013)

En determinados motores de bases de datos, esta cláusula, no reconoce el asterisco como carácter comodín y hay que sustituirlo por el carácter tanto por ciento (%).

3.8.13 Agrupamiento de registros (Count - Max, min - StDev, StDevP)

Count

Calcula el número de registros devueltos por una consulta. Su sintaxis es la siguiente:

Count(expr)

En donde expr contiene el nombre del campo que desea contar. Los operandos de expr pueden incluir el nombre de un campo de una tabla, una constante o una función (la cual puede ser intrínseca o definida por el usuario pero no otras de las funciones agregadas de SQL).

Puede contar cualquier tipo de datos incluso texto.

Aunque expr puede realizar un cálculo sobre un campo, Count simplemente cuenta el número de registros sin tener en cuenta qué valores se almacenan en los registros. La función Count no cuenta los registros que tienen campos null a menos que expr sea el carácter comodín asterisco (*).

Si utiliza un asterisco, Count calcula el número total de registros, incluyendo aquellos que contienen campos null. Count(*) es considerablemente más rápida que Count(Campo). No se debe poner el asterisco entre dobles comillas ('*').

SELECT

Count() AS Total*

FROM

Pedidos

Si expr identifica a múltiples campos, la función Count cuenta un registro sólo si al menos uno de los campos no es Null.

Si todos los campos especificados son Null, no se cuenta el registro. Hay que separar los nombres de los campos con ampersand (&).

SELECT

Count(FechaEnvío & Transporte) AS Total

FROM

Pedidos

Podemos hacer que el gestor cuente los datos diferentes de un determinado campo

SELECT

Count(DISTINCT Localidad) AS Total

FROM

Pedidos

Max, Min

Devuelven el mínimo o el máximo de un conjunto de valores contenidos en un campo específico de una consulta. Su sintaxis es:

Min(expr)

Max(expr)

En donde expr es el campo sobre el que se desea realizar el cálculo.

Expr puede incluir el nombre de un campo de una tabla, una constante o una función (la cual puede ser intrínseca o definida por el usuario pero no otras de las funciones agregadas de SQL).

SELECT

Min(Gastos) AS ElMin

FROM

Pedidos

WHERE

Pais = 'España'

SELECT

Max(Gastos) AS ElMax

FROM

Pedidos

WHERE

Pais = 'España'

StDev, StDevP

Devuelve estimaciones de la desviación estándar para la población (el total de los registros de la tabla) o una muestra de la población representada (muestra aleatoria). Su sintaxis es:

StDev(expr)

StDevP(expr)

En donde expr representa el nombre del campo que contiene los datos que desean evaluarse o una expresión que realiza un cálculo utilizando los datos de dichos campos.



Los operandos de *expr* pueden incluir el nombre de un campo de una tabla, una constante o una función (la cual puede ser intrínseca o definida por el usuario pero no otras de las funciones agregadas de SQL).

StDevP evalúa una población, y *StDev* evalúa una muestra de la población.

Si la consulta contiene menos de dos registros (o ningún registro para *StDevP*), estas funciones devuelven un valor Null (el cual indica que la desviación estándar no puede calcularse).

SELECT

StDev(Gastos) AS Desviación

FROM

Pedidos

WHERE

País = 'España'

SELECT

StDevP(Gastos) AS Desviación

FROM

Pedidos

WHERE

País = 'España'



3.8.14 Agrupamiento de registros (Sum - Var, VarP - Compute)

Sum

Devuelve la suma del conjunto de valores contenido en un campo específico de una consulta. Su sintaxis es:

Sum(expr)

En donde *expr* representa el nombre del campo que contiene los datos que desean sumarse o una expresión que realiza un cálculo utilizando los datos de dichos campos.

Los operandos de *expr* pueden incluir el nombre de un campo de una tabla, una constante o una función (la cual puede ser intrínseca o definida por el usuario pero no otras de las funciones agregadas de SQL).

SELECT

*Sum(PrecioUnidad * Cantidad) AS Total*

FROM

DetallePedido

Var, VarP

Devuelve una estimación de la varianza de una población (sobre el total de los registros) o una muestra de la población (muestra aleatoria de registros) sobre los valores de un campo.

Su sintaxis es:

Var(expr)

VarP(expr)

VarP evalúa una población, y Var evalúa una muestra de la población. Expr el nombre del campo que contiene los datos que desean evaluarse o una expresión que realiza un cálculo utilizando los datos de dichos campos.

Los operandos de expr pueden incluir el nombre de un campo de una tabla, una constante o una función (la cual puede ser intrínseca o definida por el usuario pero no otras de las funciones agregadas de SQL)

Si la consulta contiene menos de dos registros, Var y VarP devuelven Null (esto indica que la varianza no puede calcularse).

Puede utilizar Var y VarP en una expresión de consulta o en una Instrucción SQL.

SELECT

Var(Gastos) AS Varianza

FROM

Pedidos

WHERE

País = 'España'

SELECT

VarP(Gastos) AS Varianza

FROM

Pedidos

WHERE

País = 'España'

3.8.15 COMPUTE de SQL-SERVER

Esta cláusula añade una fila en el conjunto de datos que se está recuperando, se utiliza para realizar cálculos en campos numéricos.

COMPUTE actúa siempre sobre un campo o expresión del conjunto de resultados y esta expresión debe figurar exactamente igual en la cláusula SELECT y siempre se debe ordenar el resultado por la misma o al menos agrupar el resultado.

Esta expresión no puede utilizar ningún ALIAS.

SELECT

IdCliente, Count(IdPedido)

FROM

Pedidos

GROUP BY

IdPedido

HAVING

Count(IdPedido) > 20

COMPUTE

Sum(Count(IdPedido))

SELECT

*IdPedido, (PrecioUnidad * Cantidad - Descuento)*

FROM

[Detalles de Pedidos]

ORDER BY

IdPedido

COMPUTE

*Sum((PrecioUnidad * Cantidad - Descuento)) // Calcula el Total*

BY IdPedido // Calcula el Subtotal

3.8.16 Subconsultas

Una subconsulta es una instrucción SELECT anidada dentro de una instrucción SELECT, SELECT...INTO, INSERT...INTO, DELETE, o UPDATE o dentro de otra subconsulta.

Puede utilizar tres formas de sintaxis para crear una subconsulta:

Comparación [ANY | ALL | SOME] (instrucción sql)



Expresión [NOT] IN (instrucción sql)

[NOT] EXISTS (instrucción sql)

En donde:

Comparación Es una expresión y un operador de comparación que compara la expresión con el resultado de la subconsulta.

Expresión Es una expresión por la que se busca el conjunto resultante de la subconsulta.

Instrucción SQL Es una instrucción SELECT, que sigue el mismo formato y reglas que cualquier otra instrucción SELECT. Debe ir entre paréntesis.

Se puede utilizar una subconsulta en lugar de una expresión en la lista de campos de una instrucción SELECT o en una cláusula WHERE o HAVING.

En una subconsulta, se utiliza una instrucción SELECT para proporcionar un conjunto de uno o más valores especificados para evaluar en la expresión de la cláusula *WHERE* o *HAVING*.

Se puede utilizar el predicado ANY o SOME, los cuales son sinónimos, para recuperar registros de la consulta principal, que satisfagan la comparación con cualquier otro registro recuperado en la subconsulta. El ejemplo siguiente devuelve todos los productos cuyo precio unitario es mayor que el de cualquier producto vendido con un descuento igual o mayor al 25 por ciento:

```
SELECT *  
FROM  
Productos  
WHERE  
PrecioUnidad  
ANY  
(SELECT  
PrecioUnidad  
FROM  
DetallePedido  
WHERE  
Descuento = 0.25)
```

El predicado ALL se utiliza para recuperar únicamente aquellos registros de la consulta principal que satisfacen la comparación con todos los registros recuperados en la subconsulta.

Si se cambia ANY por ALL en el ejemplo anterior, la consulta devolverá únicamente aquellos productos cuyo precio unitario sea mayor que el de todos los productos vendidos con un descuento igual o mayor al 25 por ciento. Esto es mucho más restrictivo.

El predicado IN se emplea para recuperar únicamente aquellos registros de la consulta principal para los que algunos registros de la subconsulta contienen un valor igual.

El ejemplo siguiente devuelve todos los productos vendidos con un descuento igual o mayor al 25 por ciento:

```
SELECT *
FROM
  Productos
WHERE
  IDProducto
  IN
  (SELECT
  IDProducto
  FROM
  DetallePedido
  WHERE
  Descuento = 0.25)
```

Inversamente se puede utilizar NOT IN para recuperar únicamente aquellos registros de la consulta principal para los que no hay ningún registro de la subconsulta que contenga un valor igual.

3.8.17 Creación de Tablas Nuevas

Estructuras de las Tablas

Una base de datos en un sistema relacional está compuesta por un conjunto de tablas, que corresponden a las relaciones del modelo relacional. En la terminología usada en SQL no se alude a las relaciones, del mismo modo que no se usa el término atributo, pero sí la palabra columna, y no se habla de tabla, sino de línea.

Creación de Tablas Nuevas

```
CREATE TABLE tabla (  
campo1 tipo (tamaño) índice1,  
campo2 tipo (tamaño) índice2,... ,  
índice multicampo , ... )
```

En donde:

Tabla: Es el nombre de la tabla que se va a crear.

Campo#: Es el nombre del campo o de los campos que se van a crear en la nueva tabla. La nueva tabla debe contener, al menos, un campo.

Tipo: Es el tipo de datos de campo en la nueva tabla. (Ver Tipos de Datos)

Tamaño: Es el tamaño del campo sólo se aplica para campos de tipo texto.

Índice#: Es una cláusula CONSTRAINT que define el tipo de índice a crear. Esta cláusula es opcional.

Índice multicampos. Es una cláusula CONSTRAINT que define el tipo de índice multicampos a crear. Un índice multicampo es aquel que está indexado por el contenido de varios campos. Esta cláusula es opcional.

Ejemplo:

```
CREATE TABLE  
Empleados ( Nombre TEXT (25),  
Apellidos TEXT (50) )
```

(Crea una nueva tabla llamada Empleados con dos campos, uno llamado Nombre de tipo texto y longitud 25 y otro llamado apellidos con longitud 50)

Creación de Índices

Si se utiliza el motor de datos Jet de Microsoft sólo se pueden crear índices en bases de datos del mismo motor. La sintaxis para crear un índice en una tabla ya definida es la siguiente:

```
CREATE [ UNIQUE ] INDEX índice  
ON Tabla (campo [ASC|DESC][, campo [ASC|DESC], ...])  
[WITH { PRIMARY | DISALLOW NULL | IGNORE NULL }]
```

En donde:

Índice: Es el nombre del índice a crear.

Tabla: Es el nombre de una tabla existente en la que se creará el índice.

Campo: Es el nombre del campo o lista de campos que constituyen el índice.

ASC|DESC: Indica el orden de los valores de los campos ASC indica un orden ascendente (valor predeterminado) y DESC un orden descendente.

UNIQUE: Indica que el índice no puede contener valores duplicados.

DISALLOW NULL Prohíbe valores nulos en el índice

IGNORE NULL: Excluye del índice los valores nulos incluidos en los campos que lo componen.

PRIMARY: Asigna al índice la categoría de clave principal, en cada tabla sólo puede existir un único índice que sea "Clave Principal". Si un índice es clave principal implica que no puede contener valores nulos ni duplicados.

Ejemplo:

CREATE INDEX

MiIndice

ON

Empleados (Prefijo, Telefono)

(Crea un índice llamado MiIndice en la tabla empleados con los campos Prefijo y Teléfono.)

3.8.18 Modificar el Diseño de una Tabla

Modifica el diseño de una tabla ya existente, se pueden modificar los campos o los índices existentes. Su sintaxis es:

ALTER TABLE tabla {ADD {COLUMN tipo de campo[(tamaño)]

[CONSTRAINT índice]

CONSTRAINT índice multicampo} |

DROP {COLUMN campo I CONSTRAINT nombre del índice}}

En donde:

Tabla: Es el nombre de la tabla que se desea modificar.

Campo: Es el nombre del campo que se va a añadir o eliminar.

Tipo: Es el tipo de campo que se va a añadir.

Tamaño: Es el tamaño del campo que se va a añadir (sólo para campos de texto).

Índice: Es el nombre del índice del campo (cuando se crean campos) o el nombre del índice de la tabla que se desea eliminar.

Índice multicampo:	Es el nombre del índice del campo multicampo (cuando se crean campos) o el nombre del índice de la tabla que se desea eliminar.
Operación	Descripción
ADD COLUMN:	Se utiliza para añadir un nuevo campo a la tabla, indicando el nombre, el tipo de campo y opcionalmente el tamaño (para campos de tipo texto).
ADD:	Se utiliza para agregar un índice de multicampos o de un único campo.
DROP COLUMN:	Se utiliza para borrar un campo. Se especifica únicamente el nombre del campo.
DROP:	Se utiliza para eliminar un índice. Se especifica únicamente el nombre del índice a continuación de la palabra reservada CONSTRAINT.

Ejemplo:

ALTER TABLE

Empleados

ADD COLUMN

Salario CURRENCY

(Agrega un campo Salario de tipo Moneda a la tabla Empleados.)

3.8.19 Búsqueda de Registros Duplicados

Para generar este tipo de consultas lo más sencillo es utilizar el asistente de consultas de Access, editar la sentencia SQL de la consulta y pegarla en nuestro código. No obstante este tipo de consulta se consigue de la siguiente forma:

SELECT DISTINCT Lista de Campos a Visualizar FROM Tabla

WHERE CampoDeBusqueda In

(SELECT CampoDeBusqueda FROM Tabla As psudónimo

GROUP BY CampoDeBusqueda HAVING Count() > 1)*

ORDER BY CampoDeBusqueda

Un caso práctico, si deseamos localizar aquellos empleados con igual nombre y visualizar su código correspondiente, la consulta sería la siguiente:

SELECT DISTINCT

```
Empleados.Nombre, Empleados.IdEmpleado
FROM
Empleados
WHERE
Empleados.Nombre
In ( SELECT Nombre FROM Empleados As Tmp GROUP BY Nombre HAVING Count(*)
> 1)
ORDER BY Empleados.Nombre
```

3.8.20 Búsqueda de Registros no Relacionados

Este tipo de consulta se emplea en situaciones tales como saber que productos no se han vendido en un determinado periodo de tiempo:

```
SELECT DISTINCT
Productos.IdProducto, Productos.Nombre
FROM
Productos LEFT JOIN Pedidos ON
Productos.IdProducto = Pedidos.IdProducto
WHERE
(Pedidos.IdProducto Is Null)
AND
(Pedidos.Fecha Between #01-01-1998# And #01-30-1998#)
```

La sintaxis es sencilla, se trata de realizar una unión interna entre dos tablas seleccionadas mediante un LEFT JOIN, estableciendo condición que el campo relacionado de la segunda sea NULL.

CAPÍTULO 4. DISEÑO DEL SISTEMA INFORMÁTICO SIS-K

4.1 Introducción

Como mencionamos en el planteamiento del problema hemos mostrado el enfoque sistémico general ahora vamos a escribirlo aplicado a la empresa y vamos a evaluar el sistema que queremos en las fases del sistema de información.

4.1.1 Definiciones

Diseño, es la estructura prefigurada.

Sistema informático, es un sistema que permite almacenar y procesar la información en una central de procesos y gestión los usuarios.

4.2 Bases para el diseño del sistema SIS-K

Siguiendo la teoría de la información vamos a mostrar un ejemplo del flujo de la información que se va a tener en el sistema de información gerencial.



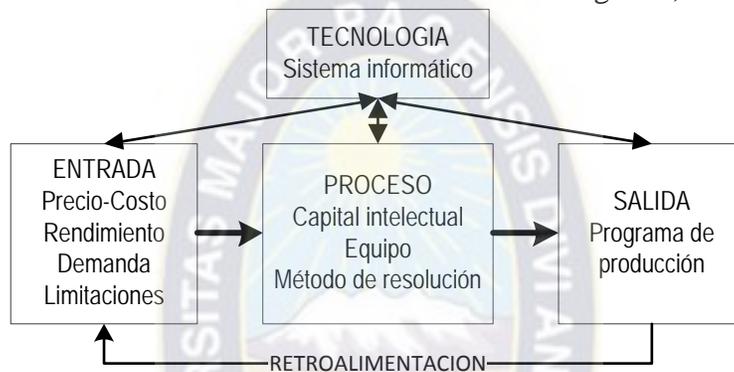
Fuente: Elaborado en base al FIGURA 1 - 2

Este sistema está muy claro, lo explicamos. El Gerente General o responsable cambia el precio de un producto, entonces nuestro sistema ya no es óptimo (sistema de producción), por qué cambiaron los coeficientes de la función objetivo, entonces es posible que la solución siga siendo óptima o que no sea óptima. Sin redundar mucho en el tema, el sistema se conecta con el Servidor de Análisis de datos, resuelve el problema de programación lineal y genera un nuevo programa de producción, este programa es

mostrado en pantalla. Finalmente estos resultados se actualizan en el Servidor de Base de Datos y toda la empresa tiene acceso a este programa, empezando por el usuario que cambio el precio del producto.

Siguiendo la teoría de sistemas vamos a mostrar el sistema de producción que queremos integrar al sistema actual, como ejemplo para luego mostrarlo modificado en el **CAPÍTULO 9. DISEÑO DEL MÓDULO PRODUCCIÓN** como subsistema de todo el sistema SIS-K.

FIGURA 4-2
SIS-K. Sistema técnico de la información designado, 2014



Fuente: Elaborado en base al CUADRO 1-2

El Hardware¹² incluye computadoras o cualquier tipo de dispositivo electrónico inteligente, que consisten en procesadores, memoria y sistemas de almacenamiento externo. El Software¹³ incluye al sistema operativo, firmware y aplicaciones, siendo especialmente importante los sistemas de gestión de bases de datos. Por último el Capital intelectual¹⁴ incluye al personal técnico que crean y mantienen el sistema (analistas, programadores y operarios) y a los usuarios que lo utilizan. Todo esto integra el subsistema de producción.

4.2.1 Diseño de sistemas

El diseño de sistemas es el arte de definir la arquitectura del Hardware, Software, componentes, módulos y datos de un sistema de cómputo para satisfacer ciertos requerimientos. Es la etapa posterior al análisis de sistemas.

¹² La Real Academia de la Lengua Española reconoce a la palabra Hardware que se puede interpretar como equipo físico.

¹³ La Real Academia de la Lengua Española reconoce a la palabra Software que se puede interpretar como soporte lógico.

¹⁴ También llamado Soporte humano

El diseño de sistemas tiene un rol más respetado y crucial en la industria de procesamiento de datos. La importancia del software multiplataforma ha incrementado la ingeniería de software a costa de los diseños de sistemas.

Los métodos de análisis y diseño orientado a objetos se están volviendo en los métodos más ampliamente utilizados para el diseño de sistemas. El UML se ha vuelto un estándar en el Análisis y diseño orientado a objetos. Es ampliamente utilizado para el modelado de sistemas de software y se ha incrementado su uso para el diseño de sistemas que no son software así como organizaciones.

4.2.2 Hardware

El Hardware se refiere a todas las partes tangibles de un sistema informático; sus componentes son: eléctricos, electrónicos, electromecánicos y mecánicos. Son cables, gabinetes o cajas, periféricos de todo tipo y cualquier otro elemento físico involucrado; contrariamente, el soporte lógico es intangible y es llamado software. El término es propio del idioma inglés (literalmente traducido: partes duras), su traducción al español no tiene un significado acorde, por tal motivo se la ha adoptado tal cual es y suena; la Real Academia Española lo define como «Conjunto de los componentes que integran la parte material de una computadora».2 El término, aunque sea lo más común, no solamente se aplica a las computadoras; del mismo modo, también un robot, un teléfono móvil, una cámara fotográfica o un reproductor multimedia poseen hardware (y software).3 4 La historia del hardware de computador se puede clasificar en cuatro generaciones, cada una caracterizada por un cambio tecnológico de importancia. Una primera delimitación podría hacerse entre hardware básico, el estrictamente necesario para el funcionamiento normal del equipo, y complementario, el que realiza funciones específicas.

Un sistema informático se compone de una unidad central de procesamiento (UCP/CPU), encargada de procesar los datos, uno o varios periféricos de entrada, los que permiten el ingreso de la información y uno o varios periféricos de salida, los que posibilitan dar salida (normalmente en forma visual o auditiva) a los datos procesados.

4.2.3 Software

Se conoce como software al equipamiento lógico o soporte lógico de un sistema informático, que comprende el conjunto de los componentes lógicos necesarios que hacen

posible la realización de tareas específicas, en contraposición a los componentes físicos que son llamados hardware.

Los componentes lógicos incluyen, entre muchos otros, las aplicaciones informáticas; tales como el procesador de texto, que permite al usuario realizar todas las tareas concernientes a la edición de textos; el llamado software de sistema, tal como el sistema operativo, que básicamente permite al resto de los programas funcionar adecuadamente, facilitando también la interacción entre los componentes físicos y el resto de las aplicaciones, y proporcionando una interfaz con el usuario.

4.2.4 Ingeniería del Software

Ingeniería de software es la aplicación de un enfoque sistemático, disciplinado y cuantificable al desarrollo, operación y mantenimiento de software, y el estudio de estos enfoques, es decir, la aplicación de la ingeniería al software. Es la aplicación de la ingeniería al software, ya que integra matemáticas, ciencias de la computación y prácticas cuyos orígenes se encuentran en la ingeniería.

Se pueden citar otras definiciones enunciadas por prestigiosos autores:

Ingeniería de software es el estudio de los principios y metodologías para el desarrollo y mantenimiento de sistemas software (Zelkowitz, 1978).

Ingeniería de software es la aplicación práctica del conocimiento científico al diseño y construcción de programas de computadora y a la documentación asociada requerida para desarrollar, operar y mantenerlos. Se conoce también como desarrollo de software o producción de software (Bohem, 1976).

La ingeniería de software trata del establecimiento de los principios y métodos de la ingeniería a fin de obtener software de modo rentable, que sea fiable y trabaje en máquinas reales (Bauer, 1972).

En 2004, la U. S. Bureau of Labor Statistics (Oficina de Estadísticas del Trabajo de Estados Unidos) contó 760 840 ingenieros de software de computadora. El término "ingeniero de software", sin embargo, se utiliza de manera genérica en el ambiente empresarial, y no todos los que se desempeñan en el puesto de ingeniero de software poseen realmente títulos de ingeniería de universidades reconocidas.

Algunos autores consideran que "desarrollo de software" es un término más apropiado que "ingeniería de software" para el proceso de crear software. Personas como Pete McBreen

(autor de "Software Craftmanship") cree que el término IS implica niveles de rigor y prueba de procesos que no son apropiados para todo tipo de desarrollo de software.

Indistintamente se utilizan los términos "ingeniería de software" o "ingeniería del software"; aunque menos común también se suele referenciar como "ingeniería en software". En Hispanoamérica los términos más comúnmente usados son los dos primeros.

La creación del software es un proceso intrínsecamente creativo y la ingeniería del software trata de sistematizar este proceso con el fin de acotar el riesgo del fracaso en la consecución del objetivo, por medio de diversas técnicas que se han demostrado adecuadas en base a la experiencia previa.

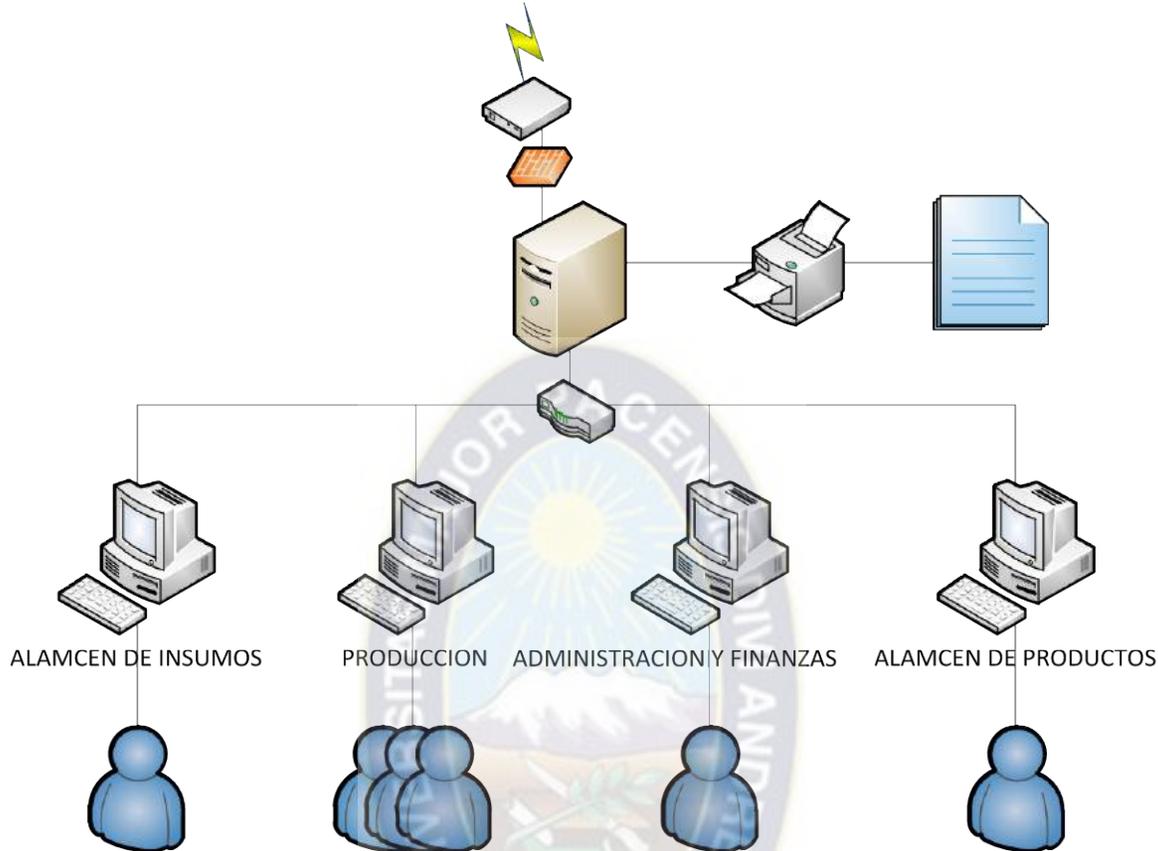
La IS se puede considerar como la ingeniería aplicada al software, esto es, por medios sistematizados y con herramientas preestablecidas, la aplicación de ellos de la manera más eficiente para la obtención de resultados óptimos; objetivos que siempre busca la ingeniería. No es sólo de la resolución de problemas, sino más bien teniendo en cuenta las diferentes soluciones, elegir la más apropiada.

4.2.5 Selección de los componentes del sistema

Hardware

Para tener un sistema informático en la Fase de Administración de datos, requerimos de un equipo que soporte servidor de archivos, servidor de uso. Entonces vamos a elaborar un mapa de red.

FIGURA 4-3
SIS-K. Diagrama de red, 2014



Fuente: Elaborado en la plantilla Diagrama de Red básica de (Microsoft Office Visio 2010) en base a Gerencia General

El mapa está claro, el sistema compone de: seis usuarios; cinco equipos procesadores con teclado, mouse y tarjeta de red; cuatro pantallas; un enrutador; un modem; una impresora; y menos de veinte metros de cable de red.

Los equipos procesadores se dividen en dos categorías servidor y cliente. El equipo servidor es uno solo y tiene las siguientes características: un procesador marca Intel(R) Core (TM) i-5 3°G de 2,6 [GHz]; una Memoria instalada (RAM) de 4 [GB] (2,66 [GB] utilizable); un Disco Duro de 798,64 [GB]. Los equipos cliente son cuatro y tienen las siguientes características: un procesador marca Intel(R) Core (TM) i-3 2°G de 2,6 [GHz]; una Memoria instalada (RAM) de 2 [GB] (1,66 [GB] utilizable); un Disco Duro de 298,64 [GB].

Ahora para validar si este sistema es apto para soportar el sistema vamos a hacer la evaluación de experiencia en Windows la cual deberá ser mayor a 5. La evaluación para este hardware está en el siguiente gráfico.

FIGURA 4-4
La española. Evaluación de experiencia en Windows, 2014

Componente	Detalles	Puntuación	Puntuación total
Procesador	Intel(R) Core(TM) i5-3230M CPU @ 2.60GHz	7,1	4,9 Determinado por la puntuación más baja
Memoria (RAM)	4,00 GB	5,5	
Gráficos	Intel(R) HD Graphics 4000	4,9	
Gráficos de juego	1168 MB (memoria de gráficos total disponible)	6,3	
Disco duro principal	187GB disponibles (293GB en total)	5,9	
Windows 7 Ultimate			

Fuente: Elaborado en base a la Evaluación de experiencia de Windows al Equipo servidor La puntuación es 5,5 determinado por la puntuación más baja considerando que los gráficos no son importantes en esta evaluación. Esta evaluación también sirve para los equipos cliente.

Los teclados, mouse no es exigible una marca en especial, para la tarjeta de red es necesario que soporte una transferencia de 100 [Mbps].

Las pantallas no tienen ninguna exigencia en particular y no es necesario que tenga entrada táctil.

El enrutador tiene que soportar la transferencia de 100 [Mbps] de la tarjeta de red, y espacio para cuatro rutas.

El modem no tiene ninguna exigencia en particular.

La impresora debe soportar el ancho de 9 [Pulgadas].

Software

Para tener un sistema informático en la Fase de Administración de datos, requerimos de una aplicación propia desarrollada exclusivamente para tener una interfaz dentro de la empresa. Entonces requerimos de un entorno de desarrollo integrado (IDE) el cual explicaremos en el CAPÍTULO 12. DISEÑO DE LA INTERFAZ, para crear la aplicación informática. Ahora solo lo vamos a elegir en función a las aptitudes que tienen estos IDE.

Las aptitudes de los IDE se muestran en el siguiente gráfico.

CUADRO 4-1
Aptitudes de los IDE, 2014

APTITUD	MS VISUAL STUDIO 2012	ECLIPSE	MONOIDEVELOP
SOPORTA MÁS DE 10 LEGUAJES DE PROGRAMACIÓN	X		X
TIENE FACILIDAD DE INCORPORAR COMPLEMENTOS	X	X	
MEJOR COMPATIBILIDAD CON MS SQL SERVER 2012	X		
SOPORTA DISPOSITIVOS CPU	X	X	X
SOPORTA DISPOSITIVOS MÓVIL	X	X	
SOPORTA CONSOLAS	X		
TOTAL	6	3	2

Fuente: Elaborado en base a experiencia de uso de los IDE

4.3 Fases del sistema SIS-K

Las fases del sistema se muestran en la siguiente figura:



Fuente: Elaborado en base al DIAGRAMA 1-3

4.3.1 Administración

El subsistema de Recursos humanos el cual desglosaremos en el CAPÍTULO 5. DISEÑO DEL MÓDULO ADMINISTRACIÓN DE RECURSOS HUMANOS. Ahora solo vamos a mostrar la estructura que queremos obtener.

DIAGRAMA 4-2

SIS-K. Subsistema de Administración de recursos humanos, 2014



Fuente: Elaborado en base a (Chiavenato 2006)

4.3.2 Provisión

El subsistema de Provisión el cual desglosaremos en el CAPÍTULO 6. DISEÑO DEL MÓDULO DE PROVISIÓN. Ahora solo vamos a mostrar la estructura que queremos obtener.

DIAGRAMA 4-3

SIS-K. Subsistema de Provisión, 2014



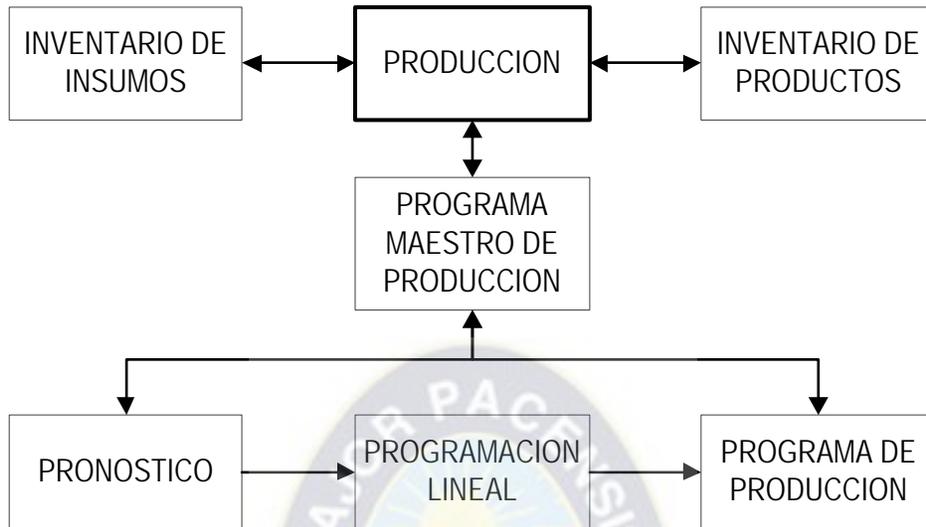
Fuente: Elaborado en base a (Braude 2005)

4.3.3 Producción

El subsistema de producción el cual desglosaremos en el CAPÍTULO 9. DISEÑO DEL MÓDULO PRODUCCIÓN. Ahora solo vamos a mostrar la estructura que queremos obtener.

DIAGRAMA 4-4

SIS-K. Subsistema de producción, 2014



Fuente: Elaborado en base a (Booch, Rumbaugh y Jacobson 1999)

4.3.4 Ventas

El subsistema de Ventas el cual desglosaremos en el CAPÍTULO 7. DISEÑO DEL MÓDULO DE VENTAS. Ahora solo vamos a mostrar la estructura que queremos obtener.

DIAGRAMA 4-5

SIS-K. Subsistema de Ventas, 2014



Fuente: Elaborado en base a (Aranda de Duero 2004-2005)

4.3.5 Financiera

El subsistema de Finanzas el cual desglosaremos en el CAPÍTULO 8. DISEÑO DEL MÓDULO FINANZAS. Ahora solo vamos a mostrar la estructura que queremos obtener.

DIAGRAMA 4-6

SIS-K. Sistema de Finanzas, 2014

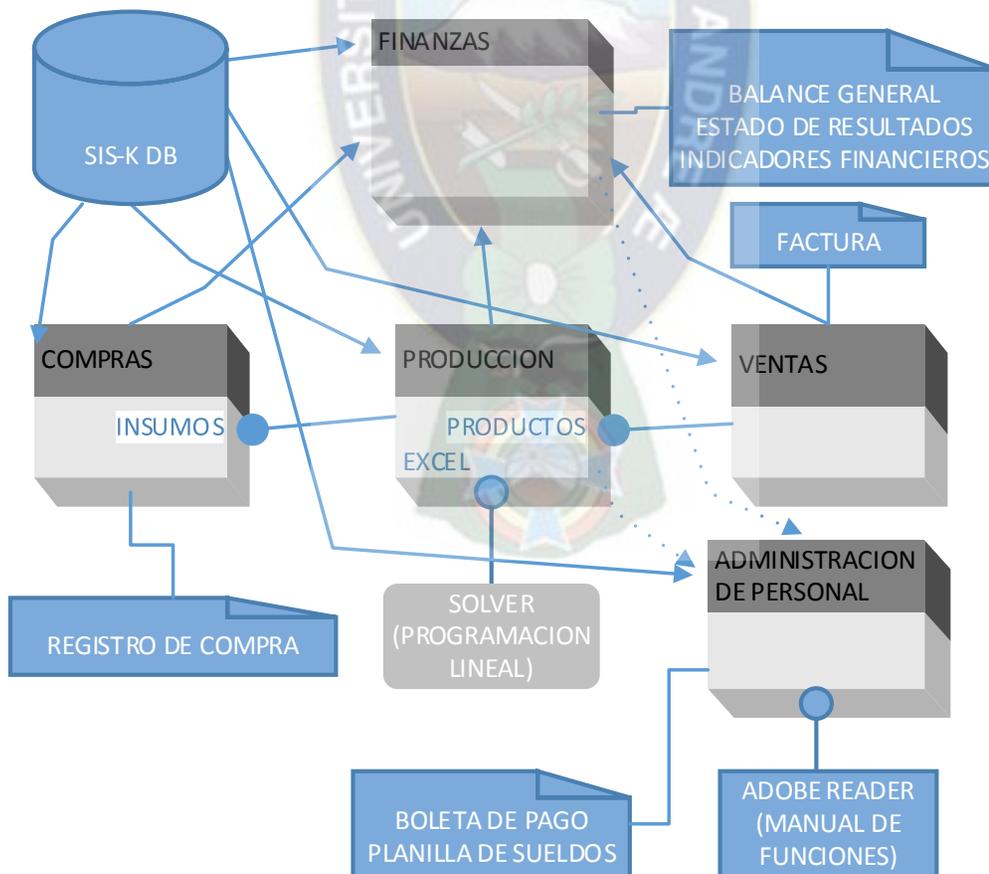


Fuente: Elaborado en base a (Stevenson 2010)

4.4 Diagrama UML de todo el sistema

DIAGRAMA 4-7

SIS-K. Diagrama UML de todo el sistema, 2014



Fuente: Elaborado en base a norma UML

CAPÍTULO 5. DISEÑO DEL MÓDULO ADMINISTRACIÓN DE RECURSOS HUMANOS

5.1 Introducción

Los sistemas de administración de recursos humanos o sistemas de información de recursos humanos conocido también como RR.HH, también conocidos por sus siglas SARH y SIRH respectivamente, forman una interfaz entre la gestión de recursos humanos y la tecnología de información. Esto combina los recursos humanos (RRHH) y en particular sus actividades administrativas con los medios puestos a su disposición por la informática, y se refieren en particular a las actividades de planificación y tratamiento de datos para integrarlos en un único sistema de gestión.

En la administración de empresas, se denomina recursos humanos (RRHH) al trabajo que aporta el conjunto de los empleados o colaboradores de una organización. Pero lo más frecuente es llamar así al sistema o proceso de gestión que se ocupa de seleccionar, contratar, formar, emplear y retener al personal de la organización. Estas tareas las puede desempeñar una persona o departamento en concreto junto a los directivos de la organización.

El objetivo básico es alinear el área o profesionales de RRHH con la estrategia de la organización, lo que permitirá implantar la estrategia organizacional a través de las personas, quienes son consideradas como los únicos recursos vivos e inteligentes capaces de llevar al éxito organizacional y enfrentar los desafíos que hoy en día se percibe en la fuerte competencia mundial. Es imprescindible resaltar que no se administran personas ni recursos humanos, sino que se administra con las personas viéndolas como agentes activos y proactivos dotados de inteligencia, creatividad y habilidades

Generalmente la función de Recursos Humanos está compuesta por áreas tales como reclutamiento y selección, contratación, capacitación, administración o gestión del personal durante la permanencia en la empresa. Dependiendo de la empresa o institución donde la función de Recursos Humanos opere, pueden existir otros grupos que desempeñen distintas responsabilidades que pueden tener que ver con aspectos tales como la administración de la nómina de los empleados o el manejo de las relaciones con sindicatos, entre otros. Para poder ejecutar la estrategia de la organización es fundamental la administración de los

Recursos humanos, para lo cual se deben considerar conceptos tales como la comunicación organizacional, el liderazgo, el trabajo en equipo, la negociación y la cultura organizacional.

5.2 Bases para la administración de recursos humanos

Las bases para la administración de recursos humanos es el manual de funciones.

5.2.1 Manual de funciones¹⁵

Es un compilado de documentos que explican la jerarquía de los puestos, las tareas del puesto y los riesgos del puesto. Este manual debe ser aprobado por el Gerente General, el Dependiente Superior y la Persona Contratada.

El manual incluye además los puestos o unidades administrativas que intervienen precisando su responsabilidad y participación. Suelen contener información y ejemplos de formularios, autorizaciones o documentos necesarios, máquinas o equipo de oficina a utilizar y cualquier otro dato que pueda auxiliar al correcto desarrollo de las actividades dentro de la empresa. En él se encuentra registrada y transmitida sin distorsión la información básica referente al funcionamiento de todas las unidades administrativas, facilita las labores de auditoría, la evaluación y control interno y su vigilancia, la conciencia en los empleados y en sus jefes de que el trabajo se está realizando o no adecuadamente. También el manual de procedimientos contiene una descripción precisa de cómo deben desarrollarse las actividades de cada empresa. Ha de ser un documento interno, del que se debe registrar y controlar las copias que de los mismos se realizan. A la hora de implantar, por ejemplo una ISO, ésta exige 4 procedimientos obligatorios como son:

- Tratamiento de No Conformidades
- Auditoría Interna
- Sistema de Mejora
- Control de la documentación.

Utilidad Permite conocer el funcionamiento interno por lo que respecta a descripción de tareas, ubicación, requerimientos y a los puestos responsables de su ejecución. Auxilian en la inducción del puesto y al adiestramiento y capacitación del personal ya que describen en forma detallada las actividades de cada puesto. Sirve para el análisis o revisión de los procedimientos de un sistema. Interviene en la consulta de todo el personal. Que se desee emprender tareas de simplificación de trabajo como análisis de tiempos, delegación de

¹⁵ Más conocido como manual de procedimientos

autoridad, etc. Para establecer un sistema de información o bien modificar el ya existente. Para uniformar y controlar el cumplimiento de las rutinas de trabajo y evitar su alteración arbitraria. Determina en forma más sencilla las responsabilidades por fallas o errores. Facilita las labores de auditoría, evaluación del control interno y su evaluación. Aumenta la eficiencia de los empleados, indicándoles lo que deben hacer y cómo deben hacerlo. Ayuda a la coordinación de actividades y evitar duplicidades. Construye una base para el análisis posterior del trabajo y el mejoramiento de los sistemas, procedimientos y métodos.

5.2.2 Conformación Del Manual

Identificación: Este documento debe incorporar la siguiente información:

Logotipo de la organización. Nombre oficial de la organización. Denominación y extensión. De corresponder a una unidad en particular debe anotarse el nombre de la misma. Lugar y fecha de elaboración. Número de revisión (en su caso). Unidades responsables de su elaboración, revisión y/o autorización. Clave de la forma. En primer término, las siglas de la organización, en segundo lugar las siglas de la unidad administrativa donde se utiliza la forma y, por último, el número de la forma. Entre las siglas y el número debe colocarse un guion o diagonal.

Índice o contenido: Relación de los capítulos y páginas correspondientes que forman parte del documento.

Prólogo y/o introducción: Exposición sobre el documento, su contenido, objeto, áreas de aplicación e importancia de su revisión y actualización. Puede incluir un mensaje de la máxima autoridad de las áreas comprendidas en el manual.

Objetivos de los procedimientos: Explicación del propósito que se pretende cumplir con los procedimientos.

Los objetivos son uniformar y controlar el cumplimiento de las rutinas de trabajo y evitar su alteración arbitraria; simplificar la responsabilidad por fallas o errores; facilitar las labores de auditoría; facilitar las labores de auditoría, la evaluación del control interno y su vigilancia; que tanto los empleados como sus jefes conozcan si el trabajo se está realizando adecuadamente; reducir los costos al aumentar la eficiencia general, además de otras ventajas adicionales.

Áreas de aplicación y/o alcance de los procedimientos: Esfera de acción que cubren los procedimientos.

Dentro de la administración pública federal los procedimientos han sido clasificados, atendiendo al ámbito de aplicación y a sus alcances, en: procedimientos macroadministrativos y procedimientos mesoadministrativos o sectoriales.

Responsables: Unidades administrativas y/o puestos que intervienen en los procedimientos en cualquiera de sus fases

Políticas o normas de operación: En esta sección se incluyen los criterios o lineamientos generales de acción que se determinan en forma explícita para facilitar la cobertura de responsabilidad de las distintas instancias que participaban en los procedimientos.

Además deberán contemplarse todas las normas de operación que precisan las situaciones alterativas que pudiesen presentarse en la operación de los procedimientos. A continuación se mencionan algunos lineamientos que deben considerarse en su planteamiento: Se definirán perfectamente las políticas y/o normas que circunscriben el marco general de actuación del personal, a efecto de que esté no incurra en fallas. Los lineamientos se elaboran clara y concisamente, a fin de que sean comprendidos incluso por personas no familiarizadas con los aspectos administrativos o con el procedimiento mismo. Deberán ser lo suficientemente explícitas para evitar la continua consulta a los niveles jerárquicos superiores.

Conceptos: Palabras o términos de carácter técnico que se emplean en el procedimiento, las cuales, por su significado o grado de especialización requieren de mayor información o ampliación de su significado, para hacer más accesible al usuario la consulta del manual.

Procedimiento (descripción de las operaciones): Presentación por escrito, en forma narrativa y secuencial, de cada una de las operaciones que se realizan en un procedimiento, explicando en qué consisten, cuándo, cómo, dónde, con qué, y cuánto tiempo se hacen, señalando los responsables de llevarlas a cabo. Cuando la descripción del procedimiento es general, y por lo mismo comprende varias áreas, debe anotarse la unidad administrativa que tiene a su cargo cada operación. Si se trata de una descripción detallada dentro de una unidad administrativa, tiene que indicarse el puesto responsable de cada operación. Es conveniente codificar las operaciones para simplificar su comprensión e identificación, aun en los casos de varias opciones en una misma operación.

Formulario de impresos. Formas impresas que se utilizan en un procedimiento, las cuales se intercalan dentro del mismo o se adjuntan como apéndices. En la descripción de las operaciones que impliquen su uso, debe hacerse referencia específica de éstas, empleando

para ello números indicadores que permitan asociarlas en forma concreta. También se pueden adicionar instructivos para su llenado.

Diagramas de flujo: Representación gráfica de la sucesión en que se realizan las operaciones de un procedimiento y/o el recorrido de formas o materiales, en donde se muestran las unidades administrativas (procedimiento general), o los puestos que intervienen (procedimiento detallado), en cada operación descrita. Además, suelen hacer mención del equipo o recursos utilizados en cada caso. Los diagramas representados en forma sencilla y accesible en el manual, brinda una descripción clara de las operaciones, lo que facilita su comprensión. Para este efecto, es aconsejable el empleo de símbolos y/o gráficos simplificados.

Glosario de términos: Lista de conceptos de carácter técnico relacionados con el contenido y técnicas de elaboración de los manuales de procedimientos, que sirven de apoyo para su uso o consulta. Procedimiento general para la elaboración de manuales administrativos

Complementando al manual de procedimientos, están las instrucciones de trabajo que completan o detallan los procedimientos, ya que se utilizan para documentar procesos específicos.

Otras normas, como son las normas que exigen diferentes procedimientos en función del sector en el que se esté implantando; Agencias de Viajes, Hoteles, Oficinas de Información Turística, Convención Bureau, etc., ya que existe una norma específica para cada uno de los sectores; en contraposición tenemos la norma ISO que es igual para todas las empresas que quieran implantarla, sea cual sea su actividad.

Diseño del proyecto: La tarea de preparar manuales administrativos requiere de mucha precisión, toda vez que los datos tienen que asentarse con la mayor exactitud posible para no generar confusión en la interpretación de su contenido por parte de quien los consulta. Es por ello que se debe poner mucha atención en todas y cada una de sus etapas de integración, delineando un proyecto en el que se consiguen todos los requerimientos, fases y procedimientos que fundamentan la ejecución del trabajo.

Responsables: Para iniciar los trabajos que conducen a la integración de un manual, es indispensable prever que no queda diluida la responsabilidad de la conducción de las acciones en diversas personas, sino que debe designarse a un coordinador, auxiliado por un equipo técnico, al que se le debe encomendar la conducción del proyecto en sus fases de

diseño, implantación y actualización. De esta manera se logra homogeneidad en el contenido y presentación de la información.

Por lo que respecta a las características del equipo técnico, es conveniente que sea personal con un buen manejo de las relaciones humanas y que conozca a la organización en lo que concierne a sus objetivos, estructura, funciones y personal. Para este tipo de trabajo, una organización puede nombrar a la persona que tenga los conocimientos y la experiencia necesarios para llevarlo a cabo. Por la naturaleza de sus funciones puede encargarlo al titular de la unidad de mejoramiento administrativo (en caso de contar con este mecanismo). Asimismo, puede contratar los servicios de consultores externos.

Delimitación del universo de estudio: Los responsables de efectuar los manuales administrativos de una organización tienen que definir y delimitar su universo de trabajo para estar en posibilidad de actuar en él. Para ello es necesario que se realice un estudio preliminar. Este paso es indispensable para conocer en forma global las funciones y actividades que se realizan en el área o áreas donde se va a actuar. Con base en él se puede definir la estrategia global para el levantamiento de información, identificando las fuentes de la misma, actividades por realizar, magnitud y alcances del proyecto, instrumentos requeridos para el trabajo y en general, prever las acciones y estimar los recursos necesarios para efectuar el estudio.

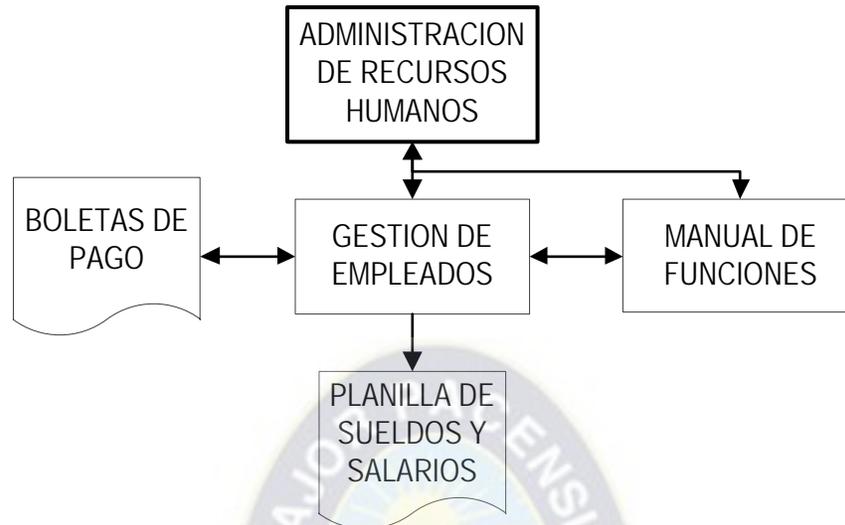
¿Cuándo se hace? Después de obtener respuestas claras y precisas para cada una de las preguntas anteriores, las mismas deben someterse, a su vez, a un nuevo interrogatorio planteando la pregunta ¿por qué? Las nuevas respuestas que se obtengan darán la pauta para formular el manual y las medidas de mejoramiento administrativo.

5.3 Fases del sistema de administración de recursos humanos

En el siguiente diagrama se explica las fases del sistema de recursos humanos.

DIAGRAMA 5-1

Sistema de administración de recursos humanos, 2014



Fuente: Elaborado en base a (Beer 1989)

5.3.1 Gestión de empleados

Entradas

En primera instancia las entradas para esta herramienta son los empleados actuales de la empresa. En segunda instancia son los registros de nuevas contrataciones.

Para las entradas vamos a mostrar un ejemplo del llenado de los campos

CUADRO 5-1

La española. Ejemplo de registro para la gestión de empleados, 2014

Código de empleado	101
Nombre	Julio L. Mendoza Gutiérrez
Dirección del empleado	Calle 4 Z Obrajes Ed El Palmar P 4 N 2
Teléfono del empleado	22112348
Cargo	Gerente General
Fecha de contratación	20/04/1980
Salario	27000
Usuario	Juliomendoza
Contraseña	*****
Foto	[Lenguaje de computador]
CI	2345674

Fuente: Elaboración propia

Proceso

Enlazado con la base de datos, el sistema debe guardar todos los datos de los empleados esto incluye nombre, carnet de identidad, dirección, teléfono, fecha de contratación, cargo, salario básico y foto entre otras.

Salidas

Muestra un reporte en dos partes. La primera parte del reporte muestra una tabla con todos los empleados con una barra para edición y otra para búsqueda avanzada. La segunda parte del reporte muestra de manera detallada todos los datos del empleado seleccionado.

FIGURA 5-1

La española. Ejemplo de reporte para la gestión de empleados, 2014



The screenshot displays a web application interface for employee management. The window title is "Gestión de Empleados". At the top, there are navigation buttons: "Nuevo", "Modificar", and "Eliminar". Below this is a section titled "Empleados Registrados" containing a table with the following data:

Cod Empl	Nombre	CI	Cargo	Salario	Fecha Contratación
101	Julio L. Mendoza Gutierrez	2345674	Gerente General	27000.0000	20/04/1990
102	Pedro Villegas Ramirez	2342354	Cortador	2500.0000	30/04/1990
103	Marco Guispe Alvarado	4395475	Encargado de N.	2500.0000	29/04/1981
105	Johnny Ticona Alaga	4345354	Cortador de Carne	2500.0000	29/04/1981
106	Osobor Apaza Ticona	5312935	Operador Cuber	2500.0000	16/08/1985

Below the table is a "Buscar Empleado" section with a "Modo de Búsqueda" dropdown menu and a "Buscar" button. The bottom section, "Datos del Empleado", shows the details for the selected employee (ID 101):

Código empleado:	101	Cargo:	Gerente General
Nombre:	Julio L. Mendoza Gutierrez	Salario:	27000.0000
CI:	2345674	Contraseña:	JMGGG101
Dirección:	Calle 4 Z Obreros Ed El Palmar P 4 N 2	Foto:	
Teléfono:	22112348		
Fecha contratación:	domingo, 20 de abril de 1990		

Fuente: Elaboración propia

Para el proceso de búsqueda de personal. Por ejemplo el usuario está buscando a Yazmín y él no sabe exactamente como se escribe su nombre así que es muy probable que no lo encuentre fácilmente. Entonces el usuario debe dirigirse al botón desplegable y seleccionar la opción nombre, luego anota en el cuadro de texto la letra "Y", y es posible que le salgan cuatro registros pero ninguno de ellos lo es, entonces anota la letra "J", y es posible que le salgan 15 registros, entonces completa la entrada con "Jaz" y es probable que no le salga

ningún registro, entonces cambia la entrada a “Jas” y le sale un registro con el nombre “Jasmín”. El problema se trataba de dos palabras homónimas. Si el usuario no hubiese tenido un sistema tan práctico probablemente hubiese tardado por lo menos veinte veces el tiempo requerido considerando la situación planteada.

5.3.2 Manual de funciones

Entradas

Todos los manuales creados con las funciones correspondientes a cada cargo y almacenados en el programa. Primero debemos transcribir todos los manuales de funciones a un formato digital, luego convertirlo a un formato de lectura rápida como “.pdf”, luego compilarlo en un programa para su fácil acceso.

Proceso

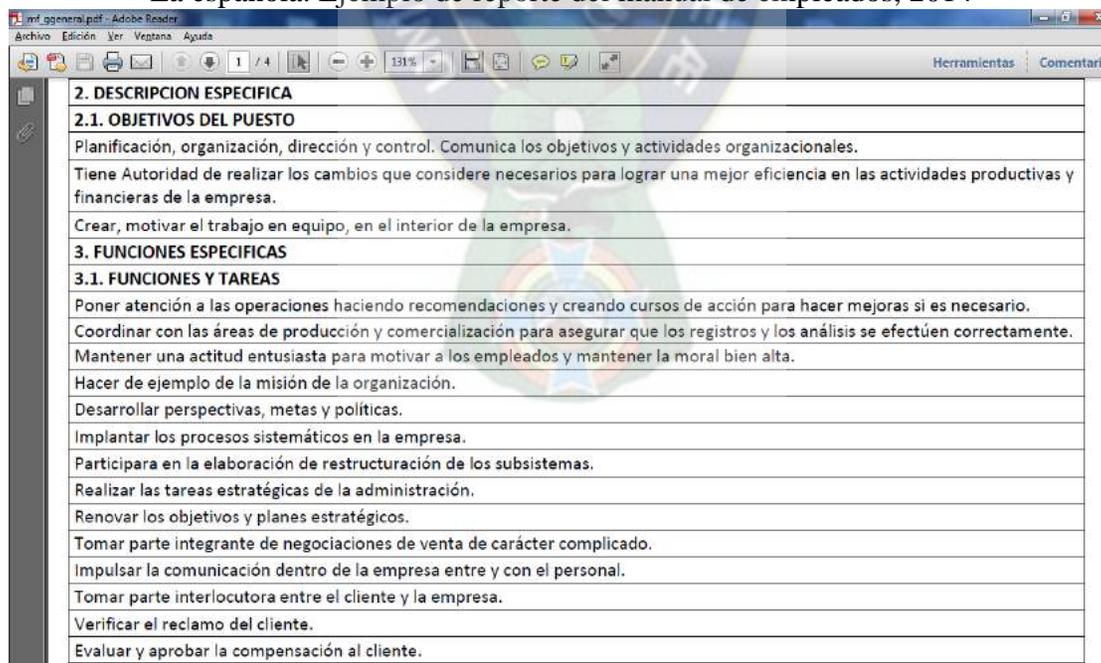
Busca el manual del cargo requerido. Hemos derivado este proceso a un programa más adecuado para su ejecución, como es Adobe Reader 7 o superior.

Salidas

Muestra el manual en formato pdf. Para su posterior impresión.

FIGURA 5-2

La española. Ejemplo de reporte del manual de empleados, 2014



The image shows a screenshot of the Adobe Reader application window. The title bar reads 'rnf_general.pdf - Adobe Reader'. The menu bar includes 'Archivo', 'Edición', 'Ver', 'Ventana', and 'Ayuda'. The toolbar shows various icons for navigation and editing, with the zoom level set to 131%. The main content area displays a table with the following structure:

2. DESCRIPCIÓN ESPECÍFICA
2.1. OBJETIVOS DEL PUESTO
Planificación, organización, dirección y control. Comunica los objetivos y actividades organizacionales.
Tiene Autoridad de realizar los cambios que considere necesarios para lograr una mejor eficiencia en las actividades productivas y financieras de la empresa.
Crear, motivar el trabajo en equipo, en el interior de la empresa.
3. FUNCIONES ESPECÍFICAS
3.1. FUNCIONES Y TAREAS
Poner atención a las operaciones haciendo recomendaciones y creando cursos de acción para hacer mejoras si es necesario.
Coordinar con las áreas de producción y comercialización para asegurar que los registros y los análisis se efectúen correctamente.
Mantener una actitud entusiasta para motivar a los empleados y mantener la moral bien alta.
Hacer de ejemplo de la misión de la organización.
Desarrollar perspectivas, metas y políticas.
Implantar los procesos sistemáticos en la empresa.
Participar en la elaboración de reestructuración de los subsistemas.
Realizar las tareas estratégicas de la administración.
Renovar los objetivos y planes estratégicos.
Tomar parte integrante de negociaciones de venta de carácter complicado.
Impulsar la comunicación dentro de la empresa entre y con el personal.
Tomar parte interlocutora entre el cliente y la empresa.
Verificar el reclamo del cliente.
Evaluar y aprobar la compensación al cliente.

Fuente: Elaboración propia

5.3.3 Boletas de pago

Entradas

Son todos los datos de cada empleado, el número de días hábiles al mes, el número de horas extra que se trabajó en el mes y si corresponde o no los dominicales.

Todos los datos de registro de empleados ya están en la base de datos, entonces la única entrada para este objeto es el número de días trabajados.

Por ejemplo vamos a poner que son 22 días sin horas extra.

Proceso

Hace todos los cálculos pertinentes en cuanto al total ganado por horas extra, dominicales y además debe calcular los descuentos correspondientes a cada empleado como ser aportes AFP y descuentos RC-IVA.

Salidas

Genera una boleta de pago que muestra de forma detallada todos los ingresos y egresos que tuvo el trabajador al mes.

En el manual de funciones vamos a mostrar específicamente el procedimiento para el correcto llenado de la boleta de pago, ahora solo vamos a mostrar un ejemplo de la boleta de pago.

FIGURA 5-3
La española. Ejemplo de reporte del boleta de pago, 2014

INGRESOS		DESCUENTOS	
Sueldo Base:	27000 por 22 dias/mes	AFP:	5544,45
Bono de Produccion:	0	RC-IVA:	4870,40
Monto Horas extra:	0 # HE: 0	Otros Descuentos:	0
Bono de Antigüedad:	13500		
Dominicales:	4909,09		
Otros Bonos:	0		
Total Ganado:	45409,09	Total Descuentos:	10414,85
		Liquido Pagable:	34994,24

Fuente: Elaboración propia

5.3.4 Planilla de sueldos y salarios

Entradas

Las entradas para la planilla de sueldos y salario son todos los registros que se hicieron en las boletas de pago. Este solo reúne todos los sueldos y salarios en una sola planilla.

Proceso

Carga todos los registros correspondientes a cada mes de los empleados y los pone dentro de la planilla de sueldos y salarios patronal que se presenta al ministerio de trabajo.

Salidas

Genera una planilla de sueldos y salarios patronal este debe ser firmado por todos los empleados de la planta.

FIGURA 5-4

La española. Ejemplo de reporte de planilla de pago, 2014

Razon Social: Industria de Alimentos de fiabres y Embutidos "La Española"
 No. Empleador:
 Correspondiente al mes de Enero de 2014

PLANILLA DE SUELDOS Y SALARIOS
 PERSONAL PERMANENTE
 (en Bolivianos)

Nombre	CI	Cargo	FechaCont	Dias Trab	Sueldo	Bono Antig	#HE	Costo HE	Bono Prod	Domi-nicales	Otros Bonos	Total Ganado	AFP
Raul Gons...	2345...	Gerente ...	20/04/1980	30	7000.0000	0.0000	0	0.0000	0.0000	12.0000	8.0000	7020.0000	857.1420
Raul Gons...	2345...	Gerente ...	20/04/1980	30	7000.0000	50.0000	4	80.0000	300.0000	56.0000	0.0000	7486.0000	914.0406
Pedro Ville...	2342...	Contador	30/04/1980	30	2500.0000	300.0000	5	100.0000	30.0000	0.0000	0.0000	2930.0000	357.7530
Jose Jose	9998...	Encarga...	29/04/1980	30	2500.0000	300.0000	5	100.0000	5.0000	0.0000	0.0000	2905.0000	354.7005
Pedro Ville...	2342...	Contador	30/04/1980	30	2500.0000	1250.0000	4	3.6085	8.0000	333.3333	0.0000	4094.9419	499.9924
Pedro Ville...	2342...	Contador	30/04/1980	25	2500.0000	1250.0000	0	3.6085	0.0000	400.0000	0.0000	4153.6085	507.1556
Julio L. Me...	2345...	Gerente ...	20/04/1980	22	7000.0000	3500.0000	0	0.0000	0.0000	0.0000	0.0000	10500.0...	1282.0500
Marco Qui...	4998...	Encarga...	29/04/1981	25	2500.0000	1250.0000	0	0.0000	0.0000	400.0000	0.0000	4150.0000	506.7150
Johnny Tic...	4345...	Cortador ...	29/04/1981	25	2500.0000	13500.0...	0	0.0000	0.0000	400.0000	0.0000	16400.0...	2002.4400
Cristobal A	5312	Onerado	16/08/1985	25	2500.0000	13500.0	0	0.0000	0.0000	400.0000	0.0000	16400.0	2002.4400

Fuente: Elaboración propia

CAPÍTULO 6. DISEÑO DEL MÓDULO DE PROVISIÓN

6.1 Introducción

El módulo de provisión es el encargado de administrar a todos los proveedores de la empresa para ello este módulo debe contar con herramientas para crear, guardar, modificar y eliminar registros de nuestros proveedores en la base de datos, así mismo también debe contar con una herramienta que genere comprobantes de compra donde se especifique el insumo a comprar, la cantidad, el proveedor y el costo.

6.2 Bases de la gestión de la provisión

La gestión de compras es un elemento imprescindible para mejorar la competitividad de las empresas. En los últimos años, con la aparición de las nuevas tecnologías, ha experimentado un gran impulso.

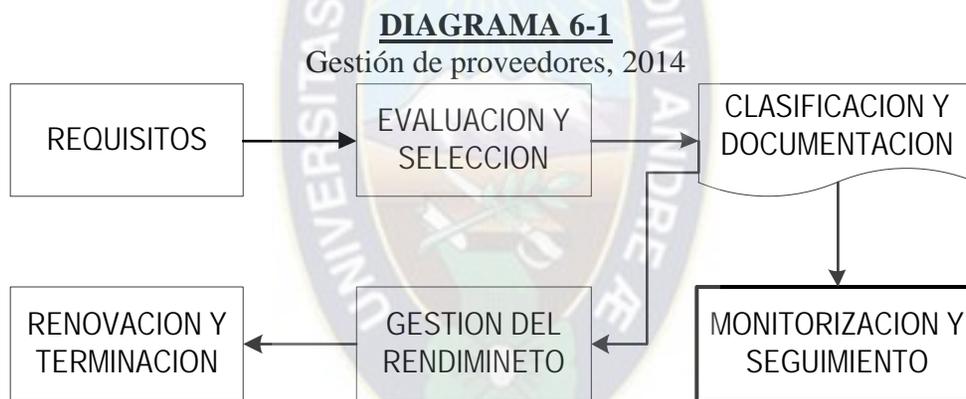
Los principales objetivos de la Gestión de Proveedores consisten en:

- Aportar el máximo valor añadido al menor coste en aquellos servicios que prestan los proveedores.
- Asegurar que los contratos y acuerdos con proveedores están alineados con la estrategia y necesidades de negocio de la organización.
- Gestionar la relación con los proveedores.
- Gestionar el rendimiento de los proveedores.
- Negociar los contratos con los proveedores y gestionarlos a lo largo de su ciclo de vida.
- Mantener una política de proveedores y una Base de Datos de Proveedores y Contratos (SCD).

Los principales riesgos a los que se enfrenta la Gestión de Proveedores son:

- La Gestión de la Demanda no proporciona las directrices básicas para racionalizar el gasto, por lo que la Gestión de Proveedores se ve forzada a improvisar los niveles de capacidad a contratar de los suministradores.
- Los contratos en vigor son demasiado vagos y no contemplan objetivos fácilmente cuantificables como horas de trabajo, número de entregables, etc.

- Los contratos son demasiado exigentes en calidad-precio, por lo que las negociaciones con los proveedores se tornan auténticas discusiones bizantinas que acaban alargándose demasiado.
- La Gestión de Proveedores no tiene a su alcance indicadores de rendimiento del servicio o los recibe demasiado tarde, por lo que si existen retrasos o disminuciones de calidad en el suministro, no podrá actuar con eficacia para corregirlo
- La Gestión de Proveedores se ocupa de definir y gestionar:
 - Los requisitos de contratación que se van a exigir a los proveedores.
 - Los procesos de evaluación y selección de proveedores.
 - La clasificación y documentación de la relación con los proveedores.
 - Gestión del Rendimiento de los proveedores
 - Renovación o terminación.



Fuente: Elaborado en base a (CHECKING 2005)

6.3 Fases de la gestión de la provisión

En el siguiente diagrama se muestra las fases de la gestión de provisión.

DIAGRAMA 6-2

SIS-K: Sistema de provisión, 2014



Fuente: Elaborado en base a (Beer 1989)

6.3.1 Gestión de Proveedores

Esta herramienta permite administrar a todos nuestros proveedores otorgándoles un código para su fácil identificación además se especifica el nombre de la empresa, su dirección, teléfono, NIT y un saldo por pagar en caso de que los insumos comprados sean otorgados a crédito.

Entradas

En primera instancia las entradas para este sub modulo son todos los actuales proveedores de la empresa. En segunda instancia se registrarán a los nuevos proveedores.

CUADRO 6-1

La española. Ejemplo de registro para la gestión de proveedores, 2014

Código de proveedor	501
Empresa	Wiña y Pacha
Dirección	Comunidad Norte Barco Belen
Teléfono	71286550
CI o NIT	201013
Código de empleado	101
Saldo cuentas por pagar	0
Saldo total de compra	0

Fuente: Elaboración propia

Proceso

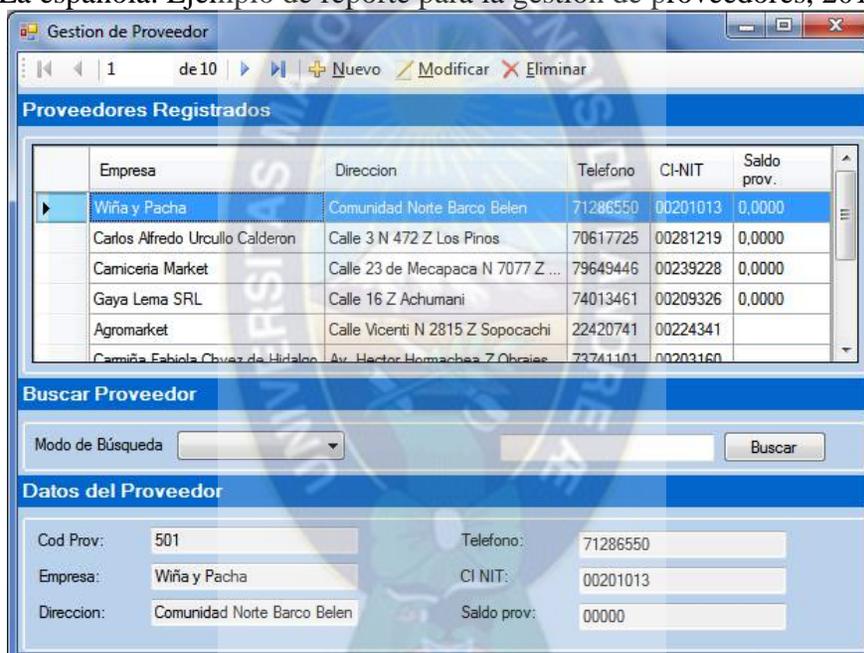
Enlazado con la base de datos los datos de los proveedores son evaluados de forma automática en función a los precios y tiempos de entrega escogiendo el sistema a los más eficientes.

Salidas

Genera un reporte con los mejores proveedores mostrando el nombre de la empresa, una descripción de que insumos nos proveerá, el teléfono para contactarnos con ellos y el saldo por pagar en caso de que tengamos alguna deuda con ellos.

FIGURA 6-1

La española. Ejemplo de reporte para la gestión de proveedores, 2014



The screenshot shows a web application window titled "Gestion de Proveedor". At the top, there are navigation buttons: "Nuevo", "Modificar", and "Eliminar". Below this is a table titled "Proveedores Registrados" with the following data:

Empresa	Direccion	Telefono	CI-NIT	Saldo prov.
Wiña y Pacha	Comunidad Norte Barco Belen	71286550	00201013	0,0000
Carlos Alfredo Urcullo Calderon	Calle 3 N 472 Z Los Pinos	70617725	00281219	0,0000
Camiceria Market	Calle 23 de Mecapaca N 7077 Z ...	79649446	00239228	0,0000
Gaya Lema SRL	Calle 16 Z Achumani	74013461	00209326	0,0000
Agromarket	Calle Vicenti N 2815 Z Sopocachi	22420741	00224341	
Camilla Fabiola Chavez de Hidalgo	Av. Hector Homachez Z Obrajes	73741101	00203160	

Below the table is a search section titled "Buscar Proveedor" with a dropdown menu for "Modo de Búsqueda" and a "Buscar" button. At the bottom, there is a "Datos del Proveedor" section with the following details:

Cod Prov:	501	Telefono:	71286550
Empresa:	Wiña y Pacha	CI NIT:	00201013
Direccion:	Comunidad Norte Barco Belen	Saldo prov:	00000

Fuente: Elaboración propia

6.3.2 Comprobante de compra

Los comprobantes de compra son una parte fundamental para la selección de proveedores debido a que gracias a ellos se pueden obtener los datos de tiempo de entrega desde que se realiza el pedido. También es un registro para auditoria de la empresa.

Entradas

Las entradas para este sub modulo son el nombre de la empresa, la cantidad del pedido, el insumo y el importe del mismo, también se registra la fecha y hora que se realizó el pedido pero eso en un segundo plano.

Todos los datos de proveedores ya están en la base de datos, entonces la única entrada aquí es la cantidad pedida y el importe. Por ejemplo vamos a poner cantidad de 1000 e importe de 230000.

Proceso

Enlazado con la base de datos se almacena esta información para su posterior proceso utilizando un algoritmo para seleccionar a los mejores proveedores.

Salidas

Genera un reporte con los datos mencionados anteriormente para tener un comprobante físico que la empresa utilizara como respaldo.

FIGURA 6-2
La española. Ejemplo de reporte para el registro de compra, 2014



The image shows a screenshot of a web-based purchase registration form. The form is titled "REGISTRO DE COMPRA" and is overlaid on a large, semi-transparent watermark of the University of the Pacific (Universidad Mayor de San Andrés) logo. The form contains the following fields and values:

Codigo Compra:	18
Proveedor:	Wiña y Pacha
Descripcion:	Conservantes
Cantidad:	1000
Importe:	23000

To the right of the form, there is a printer icon and a small "Print" button. The background of the form features a stylized sun and mountains.

Fuente: Elaboración propia

CAPÍTULO 7. DISEÑO DEL MÓDULO DE VENTAS

7.1 Introducción

El módulo de ventas es el que se encarga de la facturación de las ventas efectivas utilizando para ello dos herramientas. El primero gestiona los clientes dando la posibilidad de crear, guardar, modificar y eliminar registros de los clientes en la base de datos. La segunda debe generar comprobantes de ventas o facturas donde se especifiquen todos los campos requeridos por (Impuestos Nacionales de Bolivia 2007, 6-9)¹⁶ el cual se muestra en el ANEXO E-1.

Para nuestro efecto aplica el segundo aspecto del artículo 11. Facturación computarizada.

7.2 Bases para la gestión de ventas

7.2.1 Gestión de ventas

Es un proceso dinámico donde interactúan diferentes elementos que contribuyen a que la venta se realice, teniendo como premisa la satisfacción de las necesidades del cliente.

La gestión de ventas es uno de los componentes clave que debe practicar cada negocio basado en las ventas. Es la formación y la gestión de un equipo de ventas y el seguimiento y notificación de las ventas de la compañía. Es importante para un negocio, porque si los principios de la gestión de ventas se practican correctamente, pueden aumentar las ventas de la empresa.

7.2.2 Elementos fundamentales de la gestión de ventas.

- El Enfoque hacia el cliente
- El Equipo de Ventas.
- El Proceso de Venta.
- La Supervisión y Motivación del Equipo de ventas.
- La Comunicación en las Ventas.
- El Merchandising.

¹⁶ Resolución Normativa de Directorio N° 10.0016.07. Nuevo Sistema de Facturación (NSF-07)

7.2.3 El cliente

No es quien compra nuestros productos, sino alguien que tiene necesidades y que estamos interesados en satisfacer, por tanto es la “Vía de Satisfacción de Necesidades y Expectativas Humanas”.

Los clientes son además:

- Personas que utilizan productos o servicios de la empresa.
- Personas que se ven afectadas o beneficiadas por su producto o servicio.
- Personas que venden o suministran sus productos a otros.
- Los miembros de su equipo de trabajo.
- Personas que están directamente subordinadas a usted o que usted se le subordina (por los objetivos de trabajo).
- Personas que están a su alrededor en su centro de trabajo.
- Personas que dependen del éxito de la Empresa.
- Usted mismo

Estos se pueden clasificar de la siguiente manera:

Según su posible frecuencia de compras

Este tipo de clientes se lo identifica mediante una investigación de mercados que permite determinar su posible frecuencia de compras en el caso de que se conviertan en clientes actuales; por ello, se los divide de manera similar en:

- Clientes potenciales de compra frecuente.
- Clientes potenciales de compra habitual.
- Clientes potenciales de compra ocasional.

Según su posible volumen de compras

Esta es otra clasificación que se realiza mediante una previa investigación de mercados que permite identificar sus posibles volúmenes de compras en el caso de que se conviertan en clientes actuales; por ello, se los divide de manera similar en:

- Clientes potenciales de alto volumen de compras.
- Clientes potenciales de promedio volumen de compras.
- Clientes potenciales de bajo volumen de compras.

Según su grado de influencia

Este tipo de clientes se lo identifica mediante una investigación en el mercado meta que permite identificar a las personas que ejercen influencia en el público objetivo y a sus líderes de opinión, a los cuales, convendría convertirlos en clientes actuales para que se constituyan en Clientes Influyentes en un futuro cercano. Por ello, se dividen de forma similar en:

- Clientes potenciales altamente influyentes.
- Clientes potenciales de influencia regular.
- Clientes potenciales de influencia familiar.

Necesidades y expectativas de los clientes

Los clientes tienen necesidades y expectativas que deben ser tenidas en cuenta por la organización. Una necesidad es algo que el consumidor realmente necesita, como una cuota de seguro de coche baja. Una expectativa es algo que el consumidor no necesariamente va a conseguir pero que espera conseguir del producto o servicio, como que su coche le lleve desde su lugar de salida hasta su destino.

Las necesidades y las expectativas de las personas varían con el tiempo, ya sea por el crecimiento y desarrollo normal del individuo o por la influencia del entorno. Es normal que nuestras necesidades cambien con el pasar de los años, nuestras prioridades se modifican con la madurez y experiencia, así como con la satisfacción de las mismas. Algunas necesidades se denominan "básicas" porque perduran con el tiempo y se relacionan con aspectos físicos y ambientales como la necesidad de comer o la necesidad de tener un lugar en donde vivir.

Las necesidades no se presentan necesariamente de manera escalonada o secuencial, muchas veces podemos tener varias necesidades de manera simultánea, como la necesidad de aprender y la de descansar, las cuales se pueden dar al mismo tiempo sin que se afecte la una u la otra.

7.2.4 Proceso de ventas

Toda empresa tiene sus propios recursos para la venta. Estos recursos pueden variar en relación con la naturaleza de los productos vendidos y el volumen de actividad de la empresa; la empresa puede apoyar su gestión de ventas en:

- Vendedores independientes
- Representantes exclusivos
- Vendedores asalariados.

El que se dé una decisión positiva de compra a favor de una empresa en particular está fuertemente vinculada con la persona a quien le corresponde la misión de vender, la cual deberá poseer competencias, habilidades, entrenamiento y la motivación adecuadas para persuadir al potencial cliente a que adquiera el bien o servicio ofrecidos por las empresa.

7.2.5 Fuerza de ventas.

Es el personal externo a la empresa o propio de ésta que se dedica a la comercialización de sus productos o servicios, aunque hay también otros colaboradores que participan en el desarrollo de las ventas y en sentido amplio son componentes de esta fuerza.

7.2.6 Metodología para la gestión del equipo de ventas

- Revise el grado de competencias, habilidades y herramientas con las que están dotados los vendedores de su empresa, poniendo el énfasis en los aspectos como la orientación al logro, la preocupación por hacer el trabajo de la mejor forma posible, con la disciplina necesaria para llevarlo a cabo, su orientación al cliente percibir las necesidades de un cliente concreto y ser capaz de satisfacerlas desde el dossier de ofertas que tiene la organización para la cual trabaja y habilidades persuasivas.
- Evalúe cuál ha sido el tipo de formación que ha proporcionado a su fuerza de ventas y propóngase un plan inmediato de formación - motivación como incentivo para la realización de un esfuerzo más sistemático de colocación de productos. Este programa deberá refrescar las técnicas de ventas, el conocimiento de los productos y, agilizar el proceso de intermediación entre el cliente y la empresa.
- Revise la organización de las ventas y préstele una mayor atención al desarrollo de criterios que le permita diferenciar a los clientes por categorías. Sobre la base de esta categorización, establezca relaciones entre clientes concretos y tipos de productos que demandan, así como entre estas dos variables y los sectores geográficos en los que se encuentran.
- Revise la política de remuneración aplicada a la fuerza de ventas e intente relacionarla con el volumen de negocios y los objetivos logrados: sueldo fijo + comisión sobre volumen de negocio + prima por objetivos.
- Establezca un seguimiento más cercano de las actividades cotidianas de cada uno de los integrantes de la fuerza de ventas. Establezca reuniones semanales y préstele atención a la relación entre las actividades realizadas y los resultados obtenidos.

- Mantenga un contacto estrecho con sus principales clientes; revise conjuntamente con ellos el entorno contextual de la relación y examine sobre esa base las condiciones sobre las cuales se realizan las ventas.

7.2.7 Estrategias para la gestión de ventas

Fijación de meta

Para lograr los objetivos de ventas, la empresa necesita establecer las cifras o metas de ventas para el personal. Una forma para que una empresa logre y mantenga el crecimiento es aumentar sus cifras de ventas. Los gerentes de ventas pueden establecer metas de ventas que promuevan el crecimiento y sean alcanzables para el personal. Muchos gerentes de ventas usan bonos en efectivo u otros incentivos para motivar al personal a que logre los objetivos. Las metas de ventas para el personal también se pueden configurar para que coincidan con los puntos fuertes de cada miembro del personal.

Seguimiento

La gestión de ventas permite a la dirección realizar un seguimiento de las ventas globales de la empresa, así como las ventas individuales de cada empleado. Usando el seguimiento de ventas, la administración es capaz de saber si la empresa está en camino de cumplir sus objetivos o si los miembros individuales del equipo de ventas no están produciendo suficientes ventas. Al mantener la fuerza de ventas constantemente al día sobre el estado de tus ventas, puedes ayudarlos a ajustar sus técnicas de venta y productividad para alcanzar los objetivos de ventas de la compañía.

Reportar

Usando la gestión de ventas, una empresa puede producir informes de ventas que se puedan usar para realizar un seguimiento del rendimiento de tu fuerza de ventas a través de diferentes períodos. Por ejemplo, puedes usar los informes de ventas para comparar las ventas de la empresa en diferentes años durante el mismo período. Los informes de ventas pueden determinar la dirección que la empresa debe tomar en base a los resultados. Por ejemplo, si los informes de ventas determinan que tu compañía está experimentando un importante crecimiento año tras año, esto puede indicar que la expansión es una posible dirección para la compañía.

7.2.8 Sistema de ventas

A medida que la empresa crece, puede ser más difícil de controlar y administrar el proceso de ventas sin un sistema en su lugar. La gestión de ventas ofrece a las empresas un sistema para entrenar y manejar a los empleados al tiempo que agiliza el proceso de venta desde el empleado de ventas individual hasta el cliente. Esto es beneficioso porque si hay un problema en cualquier punto del proceso de venta que puede afectar a la rentabilidad de la empresa, se puede identificar y corregir rápidamente.

7.2.9 El merchandising

Es la parte del marketing que engloba las técnicas comerciales que permiten presentar al posible comprador final el producto o servicio en las mejores condiciones materiales y psicológicas.

Tiende a sustituir la presentación pasiva del producto o servicio por una presentación activa, apelando a todo lo que pueda hacerlo más atractivo.



Fuente: Elaborado en base a (Kotler 1995)

Principios básicos del merchandising.

- Flujo de tráfico
- Situación del producto
- Orden de marca.
- Posición de los envases.
- Productos agrupados:
- Marcado de precio:
- Punto de venta: Se utiliza para comunicar al cliente:
 - Presencia y gama de productos.
 - Alternativas de compras
 - Ubicación de los productos
 - Acciones promocionales.
- Mantenimiento y limpieza.
- Encarado. La logo marca debe estar presentada siempre de frente a la vista del comprador. Como se dice comúnmente a 90° con su vista.
- Rotación del producto.

Técnicas del merchandising

- Ubicación preferente del producto.
- Pilas y exposiciones masivas.
- Cubetas a nivel del suelo (floor stands).
- Extensiones del lineal.
- Mástiles.
- Señalizadores.
- Presentadores.
- Demostraciones y degustaciones,
- La luz, el agua, el aire y el fuego. Atraen mucho a las personas, al igual que el movimiento. La música adecuada al lugar y a la ocasión relaja, tonifica y entusiasma.

7.3 Fases de la gestión de ventas

Las fases para la gestión de ventas se muestran en el siguiente diagrama.



Fuente: Elaborado en base a (Gronroos 1990)

7.3.1 Gestión de clientes

Esta herramienta permite administrar a todos nuestros clientes otorgándoles un código para su fácil identificación además se especifica el nombre de la empresa, su dirección, teléfono, NIT y un saldo por cobrar en caso de que los productos que sean otorgados a crédito.

Entradas

En primera instancia las entradas para este su módulo son todos los actuales clientes de la empresa. En segunda instancia se registran a los nuevos clientes.

CUADRO 7-1

La española. Ejemplo de registro para la gestión de clientes, 2014

Código de cliente	101
Empresa	Hipermaxi
Dirección	Av. Alpacoma Esq Av de Policía N 500 Z C Satélite
Teléfono	22831283
CI o NIT	15356340014
Código de empleado	101
Saldo cuentas por cobrar	5
Saldo total de venta	100
¿Consumo final?	1

Fuente: Elaboración propia

Proceso

Enlazado con la base de datos, los datos de los clientes son evaluados de forma automática, el cual genera un reporte al Gerente General para mejorar la atención al cliente en función de los precios y los tiempos de entrega.

Salidas

Genera un reporte con los mejores clientes mostrando su cuenta, saldo y teléfono para contactarnos con ellos para promocionar los productos o cobrar la cuenta por cobrar.

7.3.2 Registro de venta

Los registros de venta son una parte fundamental para mejorar la atención al cliente, debido a que gracias a ellos se pueden obtener los datos de tiempo de entrega desde que se realiza el pedido. También es un registro para auditoria de la empresa.

Entrada

Las entradas para este sub modulo son el nombre de la empresa, la cantidad del pedido, el producto y el importe del mismo, también se registra la fecha y hora que se realizó el pedido pero eso en un segundo plano.

Proceso

Enlazado con la base de datos se almacena esta información para su posterior proceso utilizando un algoritmo para mejorar la atención al cliente.

Salida

Genera dos reportes con los datos mencionados anteriormente para tener un comprobante físico que la empresa utilizara como respaldo y otro para el cliente.

7.3.3 Factura

Es el reporte de registro de venta que esta modelado en función al nuevo sistema de facturación NSF-07 (Impuestos Nacionales de Bolivia 2007).

Por ejemplo la factura se muestra así.

FIGURA 7-1

La española. Ejemplo de registro para la gestión de clientes, 2014

LA ESPAÑOLA
C. URZULA GOYZUETA NO. 558
EL ALTO - BOLIVIA

F A C T U R A
ORIGINAL

NIT: 2703795012
FACTURA NO.: 13
AUTORIZACION NRO.: 390400921080

FECHA: 27/05/2014 10:59:00
SEÑOR (ES): Hipemaxi
NIT/CI: 15356340014

CANT	CONCEPTO	PRECIO	TOTAL
100	Salchicha viena	35,7000	3570,0000
200	Chorizo	40,0000	8000,0000
100	Mortadela	41,0000	4100,0000

TOTAL A PAGAR BS.: 15670,0000
EFECTIVO: 15700
CAMBIO BS.: 30,0000

SON: 00/100 BS.

 "La reproducción total o parcial y/o el uso no autorizado de esta Nota Fiscal constituye un delito a ser sancionado conforme a Ley"

Fuente: Elaboración propia

CAPÍTULO 8. DISEÑO DEL MÓDULO FINANZAS

8.1 Introducción

El modulo financiero es el que se encargara de la contabilidad de la empresa utilizando para ello cinco herramientas, el primero se encarga de realizar los registros contables, el segundo que es un gestor de cuentas, el tercero que muestra varios indicadores financieros que la empresa requiere, el cuarto un balance general y por último el quinto que es un estado de resultados, todos estos sub módulos ayudan al contador a que su trabajo sea más eficiente.

8.2 Bases para la gestión financiera

La planeación de recursos se da a través de la planeación financiera, la obtención y administración de los mismos se determina con la integración de las estructuras financiera y de inversiones que permitan el logro de objetivos organizacionales y más concretamente, del objetivo de la administración financiera.

El análisis financiero es una herramienta de la administración financiera que permite detectar las situaciones débiles y ventajosas para planear la gestión de una empresa. El proceso de análisis e interpretación de estados financieros consiste en aplicar procedimientos analíticos a las cifras de los estados financieros para descubrir relaciones significativas que coadyuven a la toma de decisiones.

Consiste en la aplicación de técnicas y métodos que tienen como objetivo proporcionar elementos de juicio suficientes para interpretar la situación financiera de la empresa.

Incluye el uso de diversos estados financieros. Estos estados tratan de lograr varias cosas. Primero, presentan los activos y pasivos de una empresa mercantil en un momento en el tiempo, por lo general, al final de un año o de un trimestre.

Esta presentación se conoce como balance general. Por otra parte el estado de resultados presenta los ingresos y los gastos de la empresa para un periodo particular del tiempo, por lo general también de un año o de un trimestre. Mientras que el balance general representa una fotografía de la posición financiera de la empresa en un momento en el tiempo, el estado de resultados presenta un resumen de rentabilidad a través del tiempo.

- Balance general.- Es un resumen claro y sencillo de la situación financiera de la empresa en una fecha determinada. Muestra todos los bienes propiedad de la

empresa (activo), así como todas sus deudas (pasivo) y, por último el patrimonio de la empresa (capital).

- Estado de resultados.- Es un informe que permite determinar se la empresa registro utilidades o pérdidas en un periodo determinado.

8.2.1 Funciones

- Elaboración de planes (presupuestos) que darán forma a la posición futura de la empresa, utilizando las técnicas de análisis financiero como punto de partida.
- Obtención de fondos, cuidando la estructura financiera, tanto por su costo como por el control de la empresa.
- Aplicación de los recursos, cuidando la liquidez al definir la estructura de inversiones.

8.2.2 Herramientas

Para el autor Abraham Perdomo Moreno, en su texto Administración del Capital de Trabajo, el Administrador Financiero cuenta con los siguientes instrumentos para realizar su labor:

- Recopilación de datos significativos: Primera fase del proceso de administración financiera de una entidad que implica recabar los datos necesarios para su posterior estudio.
- Análisis Financiero: El análisis financiero se refiere al estudio y evaluación del pasado económico-financiero de una entidad, favoreciendo con ello la toma de decisiones acertada que conduzca al logro de los objetivos preestablecidos. Lo anterior permite conocer la situación financiera de la entidad en aspectos específicos. La principal técnica de análisis financiero es la de razones financieras, misma que estudiaremos más adelante.
- Planeación Financiera: Es el estudio, evaluación y proyección del futuro económico-financiero de una entidad, para la toma de decisiones acertada y logro de objetivos organizacionales. En este sentido nos referiremos a los presupuestos, como planes financieros de una entidad y a los estados financieros proyectados o proforma.
- Control Financiero: Parte de la administración financiera que se encarga de comparar lo realizado contra la planeación financiera, detecta desviaciones y, en su caso, establece medidas correctivas necesarias.

- Toma de decisiones acertadas: como consecuencia de todo lo anterior.

Tal como se observa en párrafos anteriores, el análisis financiero evalúa y estudia el pasado financiero de una empresa, mientras que la planeación financiera se orienta al futuro y se basa en proyecciones, mismas que se obtienen a través de los presupuestos y los estados financieros proforma.

8.2.3 Técnicas del análisis financiero

Existen diversos métodos para analizar el contenido de los estados financieros como los son:

- Método de Análisis Vertical o Estático: Cuando se realiza a los estados financieros correspondientes a un periodo. Generalmente se comparan los resultados contra un parámetro, permite evaluar la eficiencia de la empresa dentro del mismo ejercicio, es decir se utiliza para analizar los estados financieros correspondientes a una fecha fija o a un periodo determinado. Para la realización de este procedimiento se aplica el procedimiento de porcentajes integrales y el de razones financieras.
- Método de Análisis Horizontal o Dinámico: Consiste en comparar cifras de diferentes periodos, es decir, de Estados Financieros Comparativos y obtener diferencias que a veces se calculan como un porcentaje de aumento o disminución respecto a un periodo más antiguo considerado como base.

8.2.4 Razones financieras

Las razones financieras constituyen un método para conocer hechos relevantes acerca de las operaciones y la situación financiera de la empresa. Para que el método sea eficaz, estas deben ser evaluadas conjuntamente y no en forma individual.

Se definirá una razón financiera como una relación entre dos cantidades de los estados financieros de una empresa, la que se obtiene dividiendo una cantidad entre otra.

El propósito de usar las razones en los análisis de los estados financieros es reducir la cantidad de datos a una forma práctica y darle un mayor significado a la información.

Para hacer más comprensible su estudio las hemos clasificado en cuatro grupos que a continuación se mencionan:

CUADRO 8-1
Razones financieras, 2014

	RAZÓN	FORMULA	CONCEPTO
LIQUIDEZ	Razón circulante	$\frac{\text{act. circulante}}{\text{pas. circulante}}$	Cobertura de las obligaciones (deudor) de corto plazo por parte del activo circulante.
	Prueba de ácido	$\frac{\text{act.circ.-inventarios}}{\text{pas. circ.}}$	Mide la forma en que los pas. de mayor liquidez cubren y garantizan al pasivo circulante.
ACTIVIDAD	Rotación del inventario	$\frac{\text{costo venta}}{\text{inventario}}$	Número de veces que los inventarios fueron convertidos a ventas en el período
	Promedio medio de inventario	$\frac{360}{\text{rotación de inventario}}$	Período de duración de los inventarios hasta que son vendidos
	Rotación de las cuentas por cobrar	$\frac{\text{venta neta a crédito}}{c \times c.}$	Velocidad en que se transforman a efectivo las cuentas por cobrar
	Período medio de cobro	$\frac{360 \times c \times c}{\text{venta neta a crédito}}$	Eficiencia con que la empresa cobra sus ventas a crédito
	Rotación del activo circulante	$\frac{\text{ventas netas totales}}{\text{act. circ.}}$	Número de veces que el activo circulante es transformado en ventas
	Rotación del activo fijo	$\frac{\text{ventas netas totales}}{\text{act.fijo neto}}$	Grado de efectividad del activo fijo para generar ventas
	Rotación del activo total	$\frac{\text{ventas netas totales}}{\text{act. total}}$	Eficiencia en el uso de recursos totales para generar ventas
	Razón de la deuda	$\frac{\text{pas. total}}{\text{act. total}}$	Proporción del act. financiado mediante deudas
	Razón de endeudamiento	$\frac{\text{pasivo total}}{\text{capital total}}$	Relación entre fondos apartados por los acreedores y los socios
	Cobertura de intereses	$\frac{\text{utild.antes int.e imp.}}{g. \times \text{intereses}}$	Refleja la capacidad de la empresa para pagar su carga financiera anual con sus utilidades
ENDEUDAMIENTO	Margen de utilidad bruta	$\frac{\text{utilidad bruta}}{\text{ventas netas}}$	Rentas sobre las operaciones de ventas y producción
	Margen de venta de explotación	$\frac{\text{utilidad de operación}}{\text{ventas netas}}$	Porcentaje de utilidad obtenido de las operaciones normales de la empresa
	Margen neto de utilidad	$\frac{\text{utilidad neta}}{\text{ventas netas}}$	Rendimiento final derivado de las ventas con respecto a la utilidad
RENTABILIDAD	Rendimiento sobre la inversión de explotación	$\frac{\text{utilidad de operación}}{\text{act. total}}$	Mide la capacidad básica de generación de utilidades de los activos
	Rendimiento sobre la inversión total	$\frac{\text{utilidad neta}}{\text{act. total}}$	Rentas final obtenidas sobre los activos totales muestra la eficiencia y la eficacia en el uso de los recursos
	Renta sobre capital	$\frac{\text{utilidad neta}}{\text{capital total}}$	Rendimiento final de los socios sobre su inversión en la empresa

Fuente: Elaborado en base a (Gitman 2003)

8.3 Fases de la gestión financiera



Fuente: Elaborado en base a (Ortega Castro 2010)

8.3.1 Registros contables

Entradas

En primera instancia se registran a todas las cuentas y sus respectivos saldos, en segunda instancia son todos los movimientos de los saldos de las cuentas y una pequeña descripción cabe recalcar que también cuenta con un registro de transacción para identificar la misma.

Proceso

Se identifican los tipos de cuenta a registrarse, se coloca el importe de las cuentas que intervienen en la transacción y se realiza el debido descuento o incremento en los saldos de las cuentas de forma automática.

Salida

La salida para este módulo es el libro diario.

8.3.2 Gestor de cuentas

Entradas

Las entradas para este sub sistema como en el anterior caso son las cuentas y saldos pero de manera más específica porque contara también con registros de tipo de cuenta, grupo y rubro.

CUADRO 8-2

La española. Ejemplo de registro de cuentas, 2014

Código de Cuenta	10001
Cuenta	Caja
Saldo	1300
Rubro	1
Grupo	1
Título	1100

Fuente: Elaboración propia

Proceso

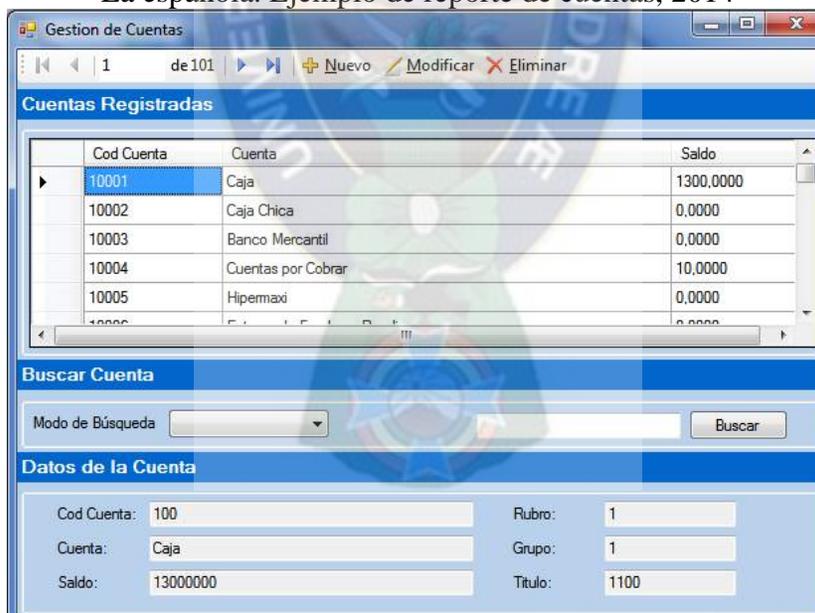
Al estar enlazado con la base de datos cualquier modificación en este afectara a todos los sub módulos que utilicen las cuentas en general a todo el modulo financiero, tendrá las opciones de edición, creación y eliminación de cuentas.

Salidas

Crea un reporte que muestra todas las cuentas y sus saldos con una barra de edición y otra de búsqueda.

FIGURA 8-1

La española. Ejemplo de reporte de cuentas, 2014



Fuente: Elaboración propia

8.3.3 Índices financieros

Entradas

La entrada para los indicadores financieros son los saldos de las cuentas que ya están en la base de datos.

Proceso

Utiliza las cuentas del sistema, las filtra y a continuación las procesa, a través de los algoritmos cargados en la base de datos se generan los índices financieros en la base de datos

Salidas

Crea un reporte con todos los índices financieros de la empresa.

FIGURA 8-2
La española. Ejemplo de reporte de indicadores financieros, 2014



Fuente: Elaboración propia

8.3.4 Balance general

Entradas

Son todas las cuentas y saldos que tiene la empresa, los cuales ya están en la base de datos.

Proceso

Filtra las cuentas por tipo, grupo y rubro, ordenando las cuentas de forma detallada a forma de un resumen del estado actual de la empresa

Salidas

Genera un reporte que muestra todos los bienes propiedad de la empresa (activo), así como todas sus deudas (pasivo) y, por último el patrimonio de la empresa (capital).

FIGURA 8-3

La española. Ejemplo de reporte de balance general, 2014

"Industria de Alimentos de Fiambres y embutidos La Española SRL"

BALANCE GENERAL
al 27 de Mayo de 2014
(Expresado en Bs.)



ACTIVO		PASIVO	
ACTIVO CORRIENTE		PASIVO CORRIENTE	
Caja	1300,00	Cuentas por Pagar	10,00
Caja chica	0,00	Proveedores por Pagar	0,00
Banco Mercantil	0,00	Sueldos y salarios por pagar	0,00
Cuentas por Cobrar	10,00	Impuesto IVA debito fiscal	290,00
Impuesto IVA credito fiscal	0,00	AFP Futuro de Bolivia	10,00
IUE por Compensar	0,00	AFP BBV Prevision	20,00
	1310,00		330,00
INVENTARIOS		PASIVO NO CORRIENTE	
Inventario de prod. tem.	20,00	Prestamo Hipotecario	0,00
Inventario de Insumos	0,00	Prestamo documentario	10,00
	20,00	Prevision beneficios sociales	0,00
		Provision aguinaldos	0,00
ACTIVO NO CORRIENTE			10,00
ACTIVO FIJO		TOTAL PASIVO	
Terrenos	10,00		340,00
Edificios y construcciones	0,00	CAPITAL	
Dep. Acum de edif y cons.	0,00	Capital	1000,00
Muebles y enseres	0,00	Aportes por capitalizar	0,00
Dep. Acum de muebl y ens.	0,00	Reserva legal	0,00
Maquinaria y equipo	0,00		
Dep. Acum de Maq. y eq.	0,00	TOTAL CAPITAL	
Vehiculos	0,00		1000,00
Dep. Acum. de Vehiculos	0,00		
Herramientas	0,00		
Dep. Acum. de Herramientas	0,00		
Equipos de computacion	0,00		
Dep. Acum. de Eq. de Comp.	0,00		
	10,00		
ACTIVO DIFERIDO			
Gastos pagados x adelant.	0,00		
	0,00		
TOTAL ACTIVO	1340,00	TOTAL PASIVO Y CAPITAL	1340,00

Fuente: Elaboración propia

8.3.5 Estado de resultados

Entradas

Son todas las cuentas y saldos que tiene la empresa, las cuales ya están en la base de datos.

Proceso

Filtra las cuentas por tipo, grupo y rubro, ordenando las cuentas de forma detallada a forma de un informe que permite determinar se la empresa registro utilidades o pérdidas en un periodo determinado

Salidas

Genera un reporte que muestra todos Ingresos percibidos por la empresa, los costos de producción, los gastos administrativos, operativos, ingresos y perdidas extraordinarias y la cantidad de impuestos a pagar, dando como resultado la utilidad o pérdida en el presente periodo.

FIGURA 8-4

La española. Ejemplo de reporte de balance general, 2014

"Industria de Alimentos de Fiambres y embutidos La Española SRL"

ESTADO DE RESULTADOS

Del 1 de Enero de 2014 al 27 de Mayo de 2014

(Expresado en Bs.)



Ventas	700,00	
- Costo de Produccion	380,00	
Utilidad Bruta		320,00
- Gastos Administrativos	238,00	
- Gastos de Ventas	60,00	
Utilidad en Operaciones		22,00
+ Ingresos extraordinarios	0,00	
- Gastos extraordinarios	6,00	
Utilidad antes de Impuestos		16,00
- Impuestos	4,00	
Utilidad Neta		16,00

Fuente: Elaboración Propia

CAPÍTULO 9. DISEÑO DEL MÓDULO PRODUCCIÓN

9.1 Introducción

El módulo de producción es aquel que ayudara a la empresa en la gestión de los inventarios tanto de insumos como de productos terminados, así mismo podrá generar un pronóstico de la demanda general y un programa de producción para ello se crearon las herramientas de: gestión de insumos, gestión de productos terminados y un programa maestro de producción.

9.2 Bases de la gestión de la producción

Sistema de planeación y control de la producción, es un programa de manufactura según una estrategia institucional.

Las estrategias institucionales están diferenciadas en cuatro grandes grupos:

9.2.1 Sistema de agotamiento de depósito

O también gestión de existencias, gestión de inventarios, regula el flujo entre las entradas de existencias y las salidas. La forma de regular el flujo de entrada es variando la frecuencia y el tamaño de los pedidos que se realicen a los proveedores. El control sobre el flujo de salida es mucho menor pues las condiciones son impuestas por los consumidores.

La gestión de existencias ha de garantizar que siempre que un cliente solicite un producto, éste sea proporcionado. Lo ideal sería que el flujo de entrada fuese igual al de salida, pero esto no es materialmente posible, pues es necesario un tiempo para responder adecuadamente. Por lo tanto se ha de intentar que el nivel de existencias sea mínimo, sin que se produzcan rupturas en la salida.

Uno de los objetivos fundamentales de la gestión de existencias es conseguir satisfacer las necesidades de los clientes, garantizando la llegada de los productos en tiempo, forma y cantidad esperados. Sin embargo, este no es el único objetivo, pues es fundamental mantener un equilibrio entre lo anterior y los costes que derivan de la posesión de las existencias.

Funciones de las existencias

Su función es la de servir de instrumento de regulación de toda la cadena logística, con el fin de conseguir un flujo de materiales continuo. Las siguientes funciones de las existencias nos las indican los autores ya nombrados anteriormente.

La función de las existencias es conseguir:

Compensar los tiempos de transporte necesarios para acercar el producto al cliente. Es necesario tener existencias cerca del consumidor ya que el momento y lugar en el que es demandado un producto suelen ser diferentes del sitio en que este producto se genera.

Evitar rupturas del movimiento de materiales por cualquier suceso, como por ejemplo los desajustes en los sistemas de transporte.

Tipos de existencias

Las existencias pueden ser clasificadas desde varios puntos de vista, de acuerdo a su función, a las actividades de la empresa y a la naturaleza de los productos.

La autora Francisca Parra Guerrero en su obra Gestión de Stocks considera útiles para la gestión de existencias las siguientes clasificaciones:

Clasificación según la función.

Se clasifica como se detalla posteriormente:

Existencias de seguridad o de protección

Artículo principal: Stock de seguridad

Volumen de existencias que se mantiene en almacén superior al necesario para el funcionamiento normal de la empresa que se constituye como protección frente a la incertidumbre de la demanda y del plazo de entrega del pedido.

Si existe un cierto tiempo entre el momento en que se hace el pedido al proveedor, y el momento en que éste llega al almacén, es decir, los aprovisionamientos no son instantáneos, entonces es indispensable prevenir un cierto número de existencias para hacer frente a la demanda o salidas de almacén, durante el plazo de reposición. Si durante este espacio de tiempo la demanda es exactamente conocida, no hay ningún inconveniente en calcular las existencias necesarias para este fin. Sin embargo, si la demanda es aleatoria, entonces tendremos un factor de incertidumbre en la cantidad necesaria de existencias futuras para hacer frente a esta demanda. Es por esto por lo que a la hora de hacer un pedido, se requiere tener en almacén un número de existencias igual al consumo normal que tiene lugar durante el plazo de reposición, más una cierta cantidad, que aunque no esperamos salga del almacén antes de que llegue el nuevo aprovisionamiento, es de hecho necesaria para prevenir las posibles fluctuaciones imprevistas de la demanda.

También se requiere el aprovisionamiento en almacén de existencias de seguridad en el caso en el que la demanda es absolutamente conocida en todo momento pero existe ahora un factor de incertidumbre en el plazo de reposición. Si el plazo de reposición es aleatorio, es necesario contar con un cierto número de existencias, suficientes para atender la demanda durante los días en que el nuevo pedido puede tardar en llegar al almacén por encima del tiempo que se considera como plazo normal; si no fuera así existe el riesgo de que falten existencias para atender a las salidas ciertos días, si el nuevo pedido tarda en llegar un tiempo imprevisto.

Por último, las dos situaciones anteriores podrían darse de forma conjunta, es decir, que las salidas de almacén estén sujetas a una cierta oscilación y que el plazo de reposición sea un tanto incierto.

La cantidad de existencias de seguridad depende de la variabilidad de la demanda y de la del plazo de entrega del pedido, y del riesgo de encontrarse sin existencias que la empresa está dispuesta a asumir. Evidentemente el riesgo de ruptura será menor cuanto mayor sea el volumen de existencias de seguridad.

La efectividad de los almacenes donde sean necesarias existencias de seguridad depende del cálculo de esté. Los parámetros que permiten estimar las fluctuaciones en el cálculo de las existencias de seguridad pueden medirse mediante distribuciones estadísticas.

Existencias medias

Es el volumen medio de existencias que tenemos en el almacén durante un periodo de tiempo. Representa la inversión media que tenemos en existencias.

El nivel de existencias en almacén evoluciona entre un máximo, que coincide con la entrada de un pedido al almacén, y un mínimo, que refleja el volumen de existencias en el momento antes de tener lugar la llegada al almacén de un nuevo pedido. Existencias medias es la medida aritmética de ambos extremos, para un solo plazo de aprovisionamiento (tiempo que media entre dos entradas consecutivas de almacén).

Si queremos calcular las existencias medias para un periodo de tiempo superior a un plazo de aprovisionamiento, tenemos que tener en cuenta que no solo se habrán producido salidas, sino que se habrá producido más de una entrada, con lo que habrá que aplicar la media ponderada según el sistema de aprovisionamiento de que se trate.

Pedidos en cantidades fijas y fechas fijas. Los tiempos que median entre dos llegadas consecutivas al almacén son uniformes y las existencias máximas y mínimas en cada ciclo

son iguales, por lo que el valor de existencias será constante e igual a la media aritmética simple de existencias máximas y mínimas de un solo plazo de aprovisionamiento. Destacamos que, no es necesario mantener existencias de seguridad, pues todos los elementos que median en la gestión son totalmente cuantificables. Esto supone que las existencias mínimas serán cero, por lo que las existencias medias serán iguales al volumen del lote a pedir Q , dividido por dos:

$$SM = \frac{Q}{2}$$

Pedidos en cantidades fijas y fechas variables. Si el plazo de entrega es nulo, es decir los aprovisionamientos son instantáneos, los niveles de existencias máximos son todos iguales y los mínimos de existencias serán cero, por tanto las existencias medias serán también $Q/2$.

Pedidos en cantidades variables y fecha fija. Si los pedidos se hacen siempre con la misma frecuencia pero por cantidades variables, o sea, que los plazos de aprovisionamiento son iguales, nos encontraremos con distintos niveles máximos de existencias en almacén, y, si la demanda es aleatoria, con distintos niveles mínimos. Para calcular en éste caso las existencias medias calcularemos la media aritmética simple de todas las existencias máximas y mínimas habidas durante dicho período de tiempo T :

$$SM = \frac{\sum_{i=1}^n (a_i + b_i)}{2n}$$

Donde a_i son los niveles máximos de existencias, b_i los niveles mínimos de existencias, y n es el número de plazos de aprovisionamiento comprendidos en el tiempo al cual referimos las existencias medias (T).

Pedidos en cantidades variables y fechas variables. En este caso es evidente que, para cada plazo de aprovisionamiento, tendremos unos niveles máximos y mínimos de existencias distintos con las consiguientes existencias medias también diferentes. Además, como los plazos de aprovisionamientos también tienen una duración variable, habrá que cuantificar las existencias medias globales mediante una media aritmética ponderada:

$$SM = \frac{\sum_{i=1}^n (a_i + b_i)}{2T} t_i$$

Donde a_i y b_i corresponden a los niveles de existencias máximo y mínimo respectivamente, t_i son la unidades de tiempo que transcurren entre la llegada al almacén de dos pedidos

consecutivos, y T es el período de tiempo total al cual referimos las existencias medias, en la unidades de tiempo correspondientes.

Existencias de anticipación

Se trata de existencias de producción estacional o periódica como es el caso de materias primas de las que hay que aprovisionarse en el momento en el que están disponibles. Pueden constituirse también para situaciones en las que es conveniente aprovisionarse en el momento en el que el precio del artículo es más bajo con la intención de especular con ellos, o para utilizarlos en periodos en los que el aprovisionamiento es más costoso, como por ejemplo en situaciones de huelgas.

Existencias sobrantes

Constituyen existencias sobrantes todos aquellos artículos que estando en buen estado no son necesarios. Estas existencias deben salir del almacén, bien sea utilizándolo de otro modo al que en principio estaba destinado, o bien, si es posible, devolviéndolo al proveedor, o si no hay otra solución, tirándolo. Estos artículos no deben ser abandonados indefinidamente en el almacén.

Existencias activas

Este tipo de existencias son conocidas también como existencias normales, existencias cíclicas o existencias de trabajo.

Normalmente no se compran o producen artículos a medida que se van demandando, sino que se lanza una orden de pedido de un tamaño superior a las necesidades del momento, dando lugar a existencias que son consumidas a lo largo del tiempo. Estas existencias siguen un comportamiento cíclico.

Estas son las existencias que abastecen la demanda de la empresa en una situación normal de funcionamiento.

Clasificación según la naturaleza física de los productos

En la gestión de almacenes puede ser determinante la naturaleza física de las existencias. La autora Parra Guerrero hace esta clasificación desde dos puntos de vista:

Según la duración de la vida útil de los productos

- Perecederos
- No perecederos
- Artículos con fecha de caducidad marcada

Según el tipo de actividad de la empresa

En empresas comerciales

- Artículos básicos. Productos que caracterizan a una empresa y que la incluyen en un sector determinado.
- Artículos obsoletos o pasados de moda. Artículos a los que no se dio salida en periodos de venta anteriores. Estos artículos han de dejar de ocupar un lugar en el almacén ya que esto conlleva un coste, por lo que habrá que encontrar la forma de darles salida.
- Artículos deteriorados o defectuosos. El volumen de este tipo de artículos en una empresa comercial no suele ser elevado ya que normalmente el proveedor suele hacerse responsable, sobre todo si el defecto es de fábrica. En caso contrario también es importante dar salida a estos artículos.
- Artículos complementarios. Formado por artículos que no son esenciales para la empresa comercial pero que sin embargo son necesarios para dar un mejor servicio al cliente.

En empresas industriales

- Materias primas. Este tipo de existencias es necesario para abastecer el proceso productivo, si se agota la materia prima el proceso de producción se verá interrumpido. El ritmo de fabricación será el que marque las salidas del almacén de materia prima.
- Productos semi-acabados procedentes de los proveedores exteriores. La gestión de este tipo de existencias tiene las mismas características que las materias primas.
- Productos semi-elaborados, pero no procedentes de proveedores exteriores, sino de una fase del proceso productivo. Son productos que ya han sufrido parte de las operaciones de producción y están a la espera de entrar en otra fase para continuar el proceso de producción. Su venta no tendrá lugar hasta que se complete dicho proceso productivo. Este tipo de existencias ha de suponer unos costes mínimos pero a la vez su volumen ha de ser el adecuado para cumplir con su objetivo, la continuidad del proceso productivo.
- Productos acabados. Son los artículos finales destinados bien al consumo final o a la venta.

- Repuestos. Todos aquellos artículos y piezas de maquinaria necesarias para solucionar averías.
- Recambio. Lo normal es que una empresa sea capaz de producir varios artículos, en la mayoría de los casos una misma máquina es utilizada para la producción de más de un tipo de artículo, pero para esto es común que haya que cambiar alguna pieza o componente de la máquina. Estas piezas y componentes constituyen las existencias de recambio.
- Suministros industriales. Todo material necesario para el mantenimiento y buena conservación del equipo industrial.

Costes de la gestión de existencias

Los costes más relevantes que se tienen en cuenta en la gestión de existencias son:

- Coste de aprovisionamiento:
 - Coste del pedido.
 - Coste de emisión de pedido.
- Coste de almacenaje.
- Coste asociado a la existencia de demanda insatisfecha.

Coste de aprovisionamiento

Es el coste total que tiene lugar cada vez que se hace un pedido de un artículo. Este coste se divide en dos:

Coste del pedido: Es el resultado de multiplicar el valor unitario del artículo por el número de artículos de que consta el pedido. Este coste es variable, pues depende de la cantidad del artículo pedido.

Coste de emisión del pedido: Este coste es fijo y será diferente dependiendo de si se trata de artículos comprados a un proveedor externo o si son pedidos de artículos manufacturados en la misma empresa.

En el primero de los casos el coste de emisión del pedido incluye varios costes, tales como: tiempo de personal administrativo, tarifas telefónicas y postales, inspecciones y operaciones de carga y descarga.

En el caso de artículos manufacturados por la propia empresa existen además de todos los costes anteriores, los costes asociados a la preparación de los equipos para el lanzamiento de la producción del pedido. Estos costes pueden incluir o no los costes asociados a la

interrupción de la producción, dependerá de si es necesario o no interrumpir la producción para la preparación de los equipos.

Coste de almacenaje

Son todos aquellos costes que tiene la empresa como consecuencia de conservar una determinada cantidad de existencias. Los factores que influyen en el coste de almacenaje pueden ser:

Obsolescencia: Puede ocurrir que haya artículos en el almacén a los que no se llegue a dar salida debido a que se hayan producido cambios en los gustos de los consumidores, o bien, se haya producido una evolución tecnológica durante el tiempo en que el artículo ha estado almacenado.

Robos y desperfectos: Durante el tiempo de mantenimiento de existencias pueden darse condiciones tales que se pueda producir la pérdida de algunos artículos, como por ejemplo roturas accidentales, condiciones ambientales no adecuadas y robos.

Seguros: Normalmente las existencias están cubiertos por algún tipo de seguro.

Almacén: Para mantener existencias es necesario disponer de almacenes, así como del personal y equipos adecuados para el manejo de materiales.

Capital: El coste de oportunidad del capital es el coste en el que se incurre al tener inmovilizado en existencias el capital correspondiente en vez de invertirlo.

Es normal que el coste de almacenaje se exprese como un porcentaje anual de la inversión en existencias incluyendo los conceptos anteriores.

Coste asociado a la existencia de demanda insatisfecha

Este coste se produce cuando no es posible atender a la demanda debido a que cuando esta se produce no hay existencias en el almacén. Esta situación es conocida como rotura de existencias.

El objetivo de la gestión de existencias

El objetivo de la gestión de existencias para un artículo individual debe dar respuestas a las siguientes preguntas básicas:

¿Cuándo debe lanzarse una orden de pedido de dicho artículo?

¿Qué cantidad del artículo debe pedirse en cada una de estas órdenes de pedido?

Clasificación de los sistemas de gestión de existencias

Los sistemas de gestión de existencias se clasifican en modelos de situaciones de existencias. El criterio de clasificación de estos modelos se hará en base al grado de conocimiento de la demanda.

Pueden darse dos situaciones para el grado de conocimiento de la demanda:

La demanda futura es conocida con exactitud, no existiendo incertidumbre en cuanto a su evolución. En este caso se dice que la demanda es conocida con certidumbre.

En el periodo considerado la previsión de la demanda no garantiza un valor concreto, en la mejor de las situaciones, se espera un valor medio, alrededor del cual es previsible cierta variabilidad. En este caso se dice que la demanda es conocida en términos probabilísticos.

Modelos deterministas: demanda independiente

Los modelos deterministas son aquellos que suponen una demanda aproximadamente constante y conocida con certidumbre. Este modelo responde a la pregunta sobre el momento en el que hay que lanzar la orden de pedido. Como la demanda es conocida y no varía en el tiempo, será suficiente conocer el tiempo de espera para saber el momento exacto en que debe ser lanzada la orden de pedido.

Por lo tanto la pregunta que nos queda responder en los modelos deterministas es la referente a la cantidad de pedido. Para contestarla determinamos el llamado lote económico de pedido.

Lote económico de pedido

Artículo principal: Cantidad Económica de Pedido

Para desarrollar un modelo simple vamos a considerar una serie de hipótesis:

- La demanda es continua a una tasa constante.
- Esta situación se prolonga en el tiempo indefinidamente.
- No hay restricciones en cuanto a cantidad de pedido, la capacidad de almacenamiento y el capital disponible.
- El plazo de entrega es nulo.
- No se permiten roturas de existencias.
- No se consideran descuentos en el precio de compra del artículo.
- Los costes se consideran invariables en el tiempo.

Aunque estas hipótesis son muy restrictivas, y es poco probable que existan situaciones de existencias que se ajusten a ellas simultáneamente, su consideración simplifica el análisis.

La mayor utilidad de este modelo sencillo es que a partir de los resultados obtenidos permite desarrollar reglas de decisión para sistemas más complejos y adaptados a situaciones reales.

En este modelo sencillo, por las hipótesis aceptadas, los costes que van a ser tenidos en cuenta son los de emisión de pedidos y el coste de almacenaje.

Variable y símbolos utilizados:

D: Demanda, expresada en unidades de artículo mantenido en existencias por periodo de un año

c: Valor unitario del artículo considerado

CE: Coste de emisión de ordenes de pedido

r: Coste de almacenaje, expresado en tanto por uno. El valor unitario del artículo c, luego el coste de mantener una unidad de artículo en almacén durante un año es $r \times c$ (unidades monetarias)

Q: Cantidad de pedido, en unidades de artículo por lote

CT: Coste total de las existencias por año

Cuando es lanzada una orden de pedido de un tamaño de Q unidades, como por hipótesis el aprovisionamiento es instantáneo, el nivel de existencias alcanza inmediatamente dicho valor Q. Desde ese momento, al ser la demanda continua y a una tasa constante, se produce una disminución del nivel de existencias hasta que éste llega a cero, momento en el que se lanza una nueva orden de pedido, repitiéndose el proceso de forma recurrente. La evolución en el tiempo del nivel de existencias tendrá la forma característica de diente de sierra.

El coste anual de emisión de pedidos será:

$$\text{Coste anual de emisión} = \frac{D}{Q} \times CE$$

Por lo tanto mientras mayor sea el lote, menor será el número de pedidos que habrá que hacer y en consecuencia los costes anuales de emisión de pedidos serán menores.

Mantener una unidad del artículo en existencias durante un año supone un coste de $r \times c$ (unidades monetarias). El pedido se realiza en cantidades y fechas fijas, por lo que las existencias medias son $Q/2$ unidades, y el coste anual de almacenaje es:

$$\text{Coste anual de almacenaje} = \frac{Q}{2} \times r \times c$$

A medida que aumente el tamaño del lote de pedido, aumenta el coste de almacenaje.

El coste total anual de las existencias será la suma de estos dos conceptos anteriores:

CT= Coste anual de emisión + Coste anual de almacenaje

$$CT = \frac{D}{Q} \times CE + \frac{Q}{2} \times r \times c$$

Formula de Wilson:

$$Q^* = \sqrt{\frac{2 \times D \times CE}{r \times c}}$$

Llamada así en honor a R. H. Wilson. Esta fórmula nos da el valor del lote Q que hace mínimo el coste total. Se representa por Q* y se obtiene al derivar con respecto a Q la expresión de CT e igualar a cero.

El coste total mínimo se obtendrá de sustituir el valor obtenido de Q* en la expresión correspondiente:

$$CT^* = \sqrt{2 \times D \times CE \times r \times c}$$

Lote económico de producción

Artículo principal: Lote Económico de Producción

Normalmente una orden de pedido es seguida de una orden de producción del artículo pedido, por lo que es necesario un cierto periodo de tiempo para completar dicha orden de producción. Durante este tiempo el artículo está siendo producido y demandado. Para que este caso tenga sentido la tasa de producción, tiene que ser mayor que la tasa de demanda, ya que si no fuese así no habría existencias en ningún momento.

Se define la tasa de producción, P, como el número de unidades producidas en un periodo de tiempo generalmente un año.

Cuando las existencias se agotan, punto A, se inicia la producción de la orden de pedido del lote Q. Se requiere un tiempo de producción Q/P. Durante este tiempo las existencias se van acumulando a una tasa P-D, por lo que cuando se acabe la producción del lote de tamaño Q se alcanzará el nivel máximo de inventario I (punto B), que es:

$$I = \frac{Q}{P}(P - D) = Q\left(1 - \frac{D}{P}\right)$$

Desde este punto, el nivel de existencias decrece, como consecuencia de una demanda uniforme y constante, cuando las existencias se agotan el ciclo se inicia de nuevo.

Coste anual de emisión:

$$\text{Coste anual de emisión} = \frac{D}{Q} \times CE$$

Las existencias medias:

$$\text{Existencias medias} = \frac{Q}{2} \left(1 - \frac{D}{P}\right)$$

Por lo que el coste anual de almacenaje es:

$$\text{Coste anual de almacenaje} = \frac{Q}{2} \left(1 - \frac{D}{P}\right) \times r \times c$$

El coste total anual:

CT = Coste anual de emisión + Coste anual de almacenaje

$$CT = \frac{D}{Q} \times CE + \frac{Q}{2} \left(1 - \frac{D}{P}\right) \times r \times c$$

Podemos obtener de la misma forma que para el caso del modelo simple, el valor del lote óptimo que minimiza los costes:

$$Q^* = \sqrt{\frac{2 \times D \times CE}{r \times c \times \left(1 - \frac{D}{P}\right)}}$$

Como era de esperar, para un aprovisionamiento instantáneo, $P = \infty$, obtenemos la fórmula de Wilson.

Descuentos por cantidad

1. Fórmula basada en EOQ Cantidad Económica de Pedido
2. Tiene en cuenta descuentos por volumen
3. La nueva fórmula de coste total incluye el precio unitario:

$$\text{Coste Total Anual} = \frac{d_a}{Q} C_s + \frac{Q}{2} C_h + d_a C_u = \frac{d_a}{Q} C_s + \frac{Q}{2} C_{h\%} C_u + d_a C_u$$

Dónde:

C_u : precio por unidad

$C_{h\%}$: Coste de mantener el inventario en porcentaje del precio por unidad

C_s : Costes de envío

d_a : demanda anual

C_h : Costes de almacenamiento

Pasos

1º Calcular Q^* para cada precio usando la fórmula de EOQ Cantidad Económica de Pedido ajustada:

$$Q^* = \sqrt{\frac{2d_a C_s}{C_{h\%} C_u}}$$

2º Si Q^* es menos que la cantidad mínima para el descuento, ajustarla hasta ésta.

3º Calcular CT para cada Q^*

4ª Escoger la Q^* con la que obtenemos los Costes Totales más bajos

Modelos no deterministas: demanda dependiente

Los modelos no deterministas son aquellos que suponen que las ventas reales para un determinado período de tiempo varían aleatoriamente alrededor de un valor medio aproximadamente constante. El factor a tener en cuenta en este modelo es la variabilidad de la demanda durante el tiempo de espera o plazo de reaprovisionamiento, de forma que permita determinar las existencias de seguridad acorde con el nivel previamente fijado de servicio a los clientes.

En los modelos no deterministas es necesario responder a las dos preguntas de cuánto y cuándo pedir para cada artículo que se mantiene en almacén. Según sea el sistema de información, previamente adoptado, un sistema de revisión continua o de revisión periódica, podemos clasificar los sistemas de gestión en dos grandes grupos:

Sistemas de revisión continua: Sistemas de información en los que el estado de las existencias se actualiza inmediatamente después de cada transacción (entrada o salida del artículo en las existencias). El sistema de gestión de existencias más sencillo y fácil de implantar para este tipo de sistema de información es el sistema de lote óptimo-punto de pedido.

Sistemas de revisión periódica: El estado de las existencias del artículo es conocido cada cierto tiempo, llamado periodo de revisión. El sistema de gestión de existencias más representativo para este tipo de sistema de información es el sistema de revisión periódica-nivel de pedido.

En estos modelos es necesario estimar la demanda durante el periodo de reaprovisionamiento, estimando la demanda media durante dicho plazo y una medida de las desviaciones previsibles en torno a este valor medio. La demanda más significativa es la demanda durante el plazo de reaprovisionamiento, ya que es en este periodo de tiempo es el que existe riesgo de rotura de existencias. Es para prevenir esta situación para lo que se tiene existencias de seguridad.

La clasificación ABC

Artículo principal: Análisis ABC

En la mayoría de los casos, se caracterizan las situaciones de existencias por tener una gran cantidad de artículos mantenidos en existencias. La clasificación ABC de los artículos que están en existencias es un instrumento eficaz para abordar la resolución de los problemas de existencias.³

En los sistemas de existencias con un gran número de artículos se observa que una pequeña porción de ellos representa un elevado porcentaje del valor monetario anual acumulado, y una gran fracción de artículos representa un pequeño porcentaje de dicho valor monetario anual total. Se establece una clasificación de los artículos en tres categorías:

A (los más importantes)

B (importancia intermedia) y

C (los menos importantes).

Según se ha definido esta clasificación, un artículo caro con poco movimiento puede ser clasificado en la misma clase que un artículo de poco valor unitario y mucho movimiento. Como el objetivo es desarrollar reglas homogéneas de gestión para cada grupo de artículos, es lógico distinguir entre estas dos situaciones, diferenciando entre artículos de alto o bajo movimiento en términos de la demanda durante el tiempo de espera.

La mayor atención será prestada a los artículos de clase A. La mayoría de las veces estos artículos representan entre un 3 y un 10 por ciento del total de los artículos en existencias y suponen más del 50 por ciento de la demanda total anual expresada en términos monetarios. Los artículos de clase B componen una clase numerosa, ya que normalmente más del 40 por ciento de los artículos representan la mayor parte del valor monetario de la demanda anual restante.

Los artículos de clase C representan el resto del valor monetario de la demanda. Para estos artículos, las reglas de decisión deben ser lo más simples posibles. Normalmente se recomienda disponer de existencias de seguridad altas para estos artículos poco importantes, con la intención de minimizar los inconvenientes causados por una rotura de existencias.

9.2.2 Sistema empujar o MRP

También llamado planificación de los materiales o MRP es un Sistema de Planificación y Administración, normalmente asociada con un software que plantea la producción y un sistema de control de inventarios.

Tiene el propósito de que se tengan los materiales requeridos, en el momento oportuno para cumplir con las demandas de los clientes. El MRP sugiere una lista de órdenes de compra. Programa las adquisiciones a proveedores en función de la producción programada.

Es un sistema que intenta dar a conocer simultáneamente tres objetivos primordiales:

Asegurar materiales y productos que estén disponibles para la producción y entrega a los clientes.

Mantener los niveles de inventario adecuados para la operación.

Planear las actividades de manufactura, horarios de entrega y actividades de compra.

Objetivos del sistema empujar

El MRP determina cuántos componentes se necesitan, así como cuándo hay que llevar a cabo el Plan Maestro de Producción, que se traduce en una serie de órdenes de compra y fabricación de los materiales necesarios para satisfacer la demanda de productos finales.

- Disminuir inventarios.
- Disminuir los tiempos de espera en la producción y en la entrega.
- Determinar obligaciones realistas.
- Incrementar la eficiencia.
- Proveer alerta temprana.
- Proveer un escenario de planeamiento de largo plazo.

Un sistema MRP debe satisfacer las siguientes condiciones:

- Asegurarse de que los materiales y productos solicitados para la producción son repartidos a los clientes.
- Mantener el mínimo nivel de inventario.
- Planear actividades de:
 - Fabricación.
 - Entregas.
 - Compras.

Procedimiento

El MRP, es un *sistema de planificación de la producción y de gestión de stocks* (o inventarios) que responde a las preguntas: ¿Qué? ¿Cuánto? y ¿Cuándo?, se debe fabricar y/o aprovisionar material.

El procedimiento del MRP está basado en dos ideas esenciales:

- La demanda de la mayoría de los artículos no es independiente, únicamente lo es la de los productos terminados.
- Las necesidades de cada artículo y el momento en que deben ser satisfechas estas necesidades, se pueden calcular a partir de unos datos bastantes sencillos:
 1. Las demandas independientes.
 2. La estructura del producto.

Así pues, el MRP consiste esencialmente en el cálculo de necesidades netas de los artículos necesarios, introduciendo un factor nuevo, no considerado en los métodos tradicionales, como es el plazo de fabricación o entrega de cada uno de los artículos, indicando la oportunidad de fabricar (o aprovisionar) los componentes respecto a su utilización en la siguiente fase del proceso.

En la base del nacimiento de los sistemas MRP está la distinción entre demanda independiente y demanda dependiente.

Esta distinción es importante, debido a que la gestión de stocks de un producto varía según su tipo de demanda. Las demandas independientes aplican métodos estadísticos de previsión por demanda continua y en las dependientes se utilizan los sistemas MRP.

El concepto de MRP es sencillo: se trata de saber qué y cuanto se debe aprovisionar/fabricar y en qué momento para cumplir con los compromisos adquiridos.

El sistema de planificación viene configurado por 3 parámetros:

- Horizonte
- Periodo
- Frecuencia

El sistema MRP

Su objetivo es disminuir el volumen de existencia a partir de lanzar la orden de compra o fabricación en el momento adecuado según los resultados del Programa Maestro de Producción. Su aplicación es útil donde existan algunas de las condiciones siguientes:

- El producto final es complejo y requiere de varios niveles de subensamble y ensamble.
- El producto final es costoso.
- El tiempo de procesamiento de la materia prima y componentes, sea grande.
- El ciclo de producción (lead time) del producto final sea largo.
- Se desee consolidar los requerimientos para diversos productos.
- El proceso se caracteriza por ítems con demandas dependientes fundamentalmente y la fabricación sea intermitente.

El sistema MRP comprende la información obtenida de al menos tres fuentes:

- El plan maestro de producción, el cual contiene las cantidades y fechas en que han de estar disponibles los productos que están sometidos a demanda externa (productos finales y piezas de repuesto).
- El estado del inventario, que recoge las cantidades de cada una de las referencias de la planta que están disponibles o en curso de fabricación, debiendo conocerse la fecha de recepción de estas últimas.
- La lista de materiales, que representa la estructura de fabricación en la empresa conociendo el árbol de fabricación de cada una de las referencias que aparecen en el Plan Maestro de Producción.

A partir de estos datos proporciona como resultado la siguiente información:

- El plan de producción de cada uno de los ítems que han de ser fabricados, especificando cantidades y fechas en que han de ser lanzadas las órdenes de fabricación. Para calcular las cargas de trabajo de cada una de las secciones de la planta y posteriormente para establecer el programa detallado de fabricación.
- El plan de aprovisionamiento, detallando las fechas y tamaños de los pedidos a proveedores para aquellas referencias que son adquiridas en el exterior.

- El informe de excepciones, que permite conocer qué órdenes de fabricación van retrasadas y cuáles son sus posibles repercusiones sobre el plan de producción y en última instancia, sobre las fechas de entrega de los pedidos a los clientes.

Plan maestro de producción PMP o MPS (Master production schedule)

Con base en los pedidos de los clientes y los pronósticos de demanda, nos dice qué productos finales hay que fabricar y en qué plazos deben tenerse terminados. También contiene las cantidades y fechas de la disponibilidad de los productos de la planta que están sujetos a demanda externa (productos finales y piezas de repuesto).

La función del plan maestro es adecuar la producción en la fábrica a los dictados de la demanda externa. Una vez fijado este, el cometido del resto del sistema es su cumplimiento y ejecución con el máximo de eficiencia.

Para esto el Plan maestro de producción se basa un tiempo que se establece para el cálculo de las fechas de producción y abastecimiento. Se ha estandarizado que este tiempo sea de una semana laboral.

Lista de materiales o BOM (*Bill of Materials*)

Desde el punto de vista del control de la producción interesa conocer los componentes que intervienen en el conjunto final, mostrando las sucesivas etapas de la fabricación. La estructura de fabricación es la lista precisa y completa de todos los materiales y componentes que se requieren para la fabricación o montaje del producto final.

Para definir esta estructura existen dos requisitos:

- Cada componente o material que interviene debe tener asignado un código que lo identifique de forma precisa.
- A cada elemento le corresponde un nivel en la estructura, asignado en sentido descendente. Así, al producto final le corresponde el nivel cero. Los componentes y materiales que intervienen en la última operación de montaje son de nivel uno.

En resumen, las listas de materiales han de organizarse para satisfacer todas las necesidades del mismo, incluyendo la de facilitar el conocimiento permanente.

Factor Humano en la Planeación de la Producción

La diferencia que hay entre la teoría (Los planes de producción realizados por algoritmos computarizados) y la práctica (Secuencia real en los talleres), es lo denominado el factor

humano. El ser humano se convierte en un complemento necesario para planeación de la producción y actúan de acuerdo a un sistema genérico de modelación de errores.

Un sistema genérico de modelación de errores corrige los fallos ocurridos en los planes de producción. Para ello el sistema integra tres niveles de error, estando relacionado el nivel de forma ascendente, con la intensidad de supervisión por parte del planificador o jefe de taller:

- El primer nivel, Nivel Basado en Habilidades, contiene soluciones a errores rutinarios que no necesitan supervisión.
- En el segundo nivel, Nivel Basado en Reglas, existen una serie de reglas condicionales, establecidas con la experiencia y el conocimiento de los trabajadores.
- Y en el tercer nivel, Nivel Basado en el Conocimiento, se buscan herramientas de análisis y solución de problemas y cuando se llega a una solución esta se convierte en una regla para el nivel inferior.

En general el conocimiento tácito que tienen los seres humanos son un complemento indispensable para obtener el plan de producción óptimo.

9.2.3 Sistema Jalar o Justo a tiempo JIT

El método justo a tiempo (traducción del inglés Just in Time) es un sistema de organización de la producción para las fábricas, de origen japonés. También conocido como método Toyota o JIT, permite aumentar la productividad. Permite reducir el costo de la gestión y por pérdidas en almacenes debido a acciones innecesarias. De esta forma, no se produce bajo suposiciones, sino sobre pedidos reales. Una definición del objetivo del Justo a Tiempo sería «producir los elementos que se necesitan, en las cantidades que se necesitan, en el momento en que se necesitan».

La producción JIT es simultáneamente una filosofía y un sistema integrado de gestión de la producción, que evolucionó lentamente a través de un proceso de prueba y error a lo largo de un período de más de quince años. En las fábricas japonesas se estableció un ambiente adecuado para esta evolución desde el momento en que dio a sus empleados la orden de que “eliminaran el desperdicio”. El desperdicio puede definirse como "cualquier cosa distinta de la cantidad mínima de equipamiento, materiales, partes, espacio y tiempo, que sea absolutamente esencial para añadir valor al producto" (Suzaki, 1985).

Para el desarrollo del JIT no hubo ningún plan maestro ni ningún borrador. Taiichi Ohno, su creador, describe el desarrollo del JIT del siguiente modo: "Al intentar aplicarlo, se

pusieron de manifiesto una serie de problemas. A medida que estos se aclaraban, me indicaban la dirección del siguiente movimiento. Creo que sólo mirando hacia atrás, somos capaces de entender cómo finalmente las piezas terminaron encajando".

Los sistemas JIT han tenido un auge sin precedentes durante las últimas décadas. Así, después del éxito de las compañías japonesas durante los años que siguieron a la crisis de los setenta, investigadores y empresas de todo el mundo centraron su atención en una forma de producción que, hasta ese momento, se había considerado vinculada con las tradiciones tanto culturales como sociales de Japón y, por tanto, muy difícil de implantar en industrias no japonesas. Sin embargo más tarde quedó demostrado que, si bien la puesta en práctica de los principios y técnicas que sostenían los sistemas de producción JIT requerían un profundo cambio en la filosofía de producción, no tenían como requisito imprescindible una forma de sociedad específica. Tras ser adoptado formalmente por numerosas plantas japonesas en los años 70, el sistema JIT comenzó a ser implantado en Estados Unidos en los años 80. En el caso de España, algunas de las experiencias iniciales de implantación de técnicas de producción JIT mostraron la viabilidad de estos enfoques en ese país.

Fundamentos del proceso

Recursos flexibles

La flexibilidad en los recursos, materializada en el empleo de trabajadores versátiles y de máquinas multiuso, fue uno de los primeros elementos en ser ajustado. Mediante el estudio de movimientos y tiempos, Ohno observó que los ciclos de trabajo de las máquinas y de los operarios que las manejaban eran muy diferentes. Con frecuencia, el empleado debía esperar un cierto tiempo mientras la máquina realizaba su función.

Surgió así la idea de que un solo operario podía manejar varias máquinas. Para facilitar la puesta en práctica, las máquinas se colocaban en paralelo o en forma de L. Al aumentar paulatinamente el número de máquinas a cargo de cada empleado, estas se acabaron colocando en forma de U, donde el principio y el final de la línea están juntos. Al principio todas las máquinas eran del mismo tipo. Posteriormente, la variedad de la maquinaria que manejaba cada operario se fue ampliando. Resultó preciso, por lo tanto, formar y preparar a los trabajadores para realizar distintos tipos de tareas, y crear al efecto programas de rotación de puestos específicos. También fue necesario efectuar modificaciones en las máquinas. Hubo que instalar mecanismos que detuvieran las máquinas automáticamente

una vez que su trabajo hubiera concluido. También se idearon fijaciones que facilitaban la labor, de forma que no fuera necesario desplazarse cuando hiciera falta. La versatilidad de los empleados estimuló la adquisición de maquinaria de uso múltiple, que redujo los desplazamientos de los empleados, así como la dificultad de adaptación de las máquinas a diversos usos y los periodos de espera para que las otras máquinas estuvieran disponibles.

Distribución en planta celular

Las células agrupan máquinas de diversos tipos para elaborar piezas de forma similar o con requerimientos de procesamiento parecidos. La organización de la maquinaria en cada célula recuerda a una cadena de montaje, normalmente en forma de U. El trabajo se desplaza por la célula de un proceso a otro, mientras los empleados siguen un camino establecido. La forma en que las células se distribuyen facilita la producción simultánea de diferentes productos, y permite que problemas derivados de las variaciones en el volumen de producción puedan ser resueltos incorporando más personal a la célula. Como en cada célula se elaboran artículos similares, el tiempo de adaptación de las máquinas es pequeño y el tamaño de los lotes de producción puede disminuir.-

Sistema jalar

Uno de los grandes problemas a los que se enfrentan las empresas, particularmente las automovilísticas, es la coordinación entre la producción, entrega de materiales, partes con la elaboración de ensamblados parciales y las necesidades de la cadena de montaje. Tradicionalmente, los inventarios se han empleado como elemento amortiguador de los fallos de coordinación. La respuesta de la producción JIT al problema fue el sistema jalar.

Este sistema requiere invertir el habitual flujo proceso-información, que caracteriza al tradicional sistema empujar. En este último, se elabora un programa que establece la labor a realizar para cada una de las estaciones de trabajo, cada una de las cuales “empuja” posteriormente el trabajo ya realizado hasta la siguiente etapa. Sin embargo, en el sistema jalar los trabajadores retroceden hasta la estación anterior para retirar de ella los materiales y partes que necesitan para procesarlos inmediatamente. Cuando se retira el material, los operarios de la estación previa saben que ha llegado el momento de comenzar a producir para reemplazar la producción retirada por la siguiente estación. Si la producción no se retira, los empleados de la estación previa detienen su labor. De este modo se evita tanto el exceso como el defecto en la producción. Se produce sólo lo necesario, entendiendo como

tal no lo que viene establecido en un plan, sino lo que los consumidores demandan. Para controlar mejor el funcionamiento del sistema, se consideró necesario establecer un mecanismo de formalización, denominado sistema de Kanban (en japonés, tarjetas).

Producción en pequeños lotes

Producir en lotes pequeños resulta atractivo desde dos perspectivas. Por un lado, se necesita menos espacio y se inmovilizan menos recursos, la distancia entre los procesos puede ser reducida, y con ella el coste de transporte interno entre estaciones. Por otro, la reducción de los niveles de inventario hace que los procesos se vuelvan más interdependientes, lo que permite detectar y resolver rápidamente los problemas.

Reducción de los tiempos de fabricación y minimizado de los tiempos de entrega

Los problemas comerciales de toma de pedidos desaparecen cuando se conoce la respuesta de fabricación. No se escatima en maquinaria de producción. Se trabaja acorde a los tiempos de trabajo, nada más. Se reduce el tiempo de terminación (lead time) de un producto, el cual está integrado por cuatro componentes:

- El tiempo de movimiento, que se reduce acercando las máquinas, simplificando los desplazamientos, estableciendo rutas más racionales o eliminando la necesidad de desplazar materiales.
- El tiempo de espera, que puede mejorarse programando mejor la producción e instalando más capacidad.
- El tiempo de adaptación de las máquinas: es con frecuencia el gran cuello de botella al que se enfrentan las empresas, y su reducción constituye uno de los elementos vitales del sistema JIT.
- El tiempo de procesamiento, que puede reducirse disminuyendo el tamaño de los lotes o incrementando la eficiencia de la maquinaria o los operarios.

Minimizar el stock

Reducir el tamaño del stock también obliga a una muy buena relación con los proveedores y subcontratistas, y además así ayuda a disminuir en gran medida los costes de almacenamiento (inventario).

Tolerancia cero a errores

Nada debe fabricarse sin la seguridad de poder hacerlo sin defectos, pues los defectos tienen un coste importante y además con los defectos se tiene entregas tardías, y por tanto se pierde el sentido de la filosofía JIT

Metodología 5 (S)

La metodología 5s tiene la creación de lugares de trabajo más organizados, ordenados, limpios y seguros. Mediante su conocimiento y aplicación se pretende crear una cultura empresarial que facilite, por un lado, el manejo de los recursos de la empresa, y por otro, la organización de los diferentes ambientes laborales, con el propósito de generar un cambio de conductas que repercutan en un aumento de la productividad. Incide directamente en la forma en que los obreros realizan su trabajo. Representan principios básicos japoneses, cuyos nombres empiezan con la letra S:

1- Seiri (organización) 2- Seiton (orden) 3- Seiso (limpieza) 4- Seiketsu (esmero) 5- Shitsuke (rigor)

Cero paradas técnicas

Se busca que las máquinas no tengan averías, ni tiempos muertos en recorridos, ni tiempos muertos en cambio de herramientas.

Adaptación rápida de la maquinaria. Sistema SMED.

El sistema SMED (Single Minute Exchange of Dies, Sistema de Tiempos Cortos de Preparación) con capacidad permite reducir el tiempo de cambio de herramientas en las máquinas aportando ventajas competitivas para la empresa. Los principios sobre los que se basa el sistema son los siguientes:

- Separar la adaptación interna de la externa. La interna es aquella que se ha de realizar cuando la máquina está detenida. La externa, aquella que puede realizarse anticipadamente, mientras la máquina está aún funcionando. Para cuando la máquina haya terminado de procesar un lote, es necesario que los operarios hayan realizado la adaptación externa, y estén preparados para llevar a cabo la interna. Sólo esta idea puede ahorrar el 30-50% del tiempo.
- Convertir la adaptación interna en externa. Ello implica asegurarse de que todas las condiciones operativas (reunir herramientas, calentar los moldes, etc.) se cumplen antes de detener la maquinaria.

- Simplificar todos los aspectos de la adaptación. Las actividades de adaptación externa pueden mejorarse organizando adecuadamente el lugar de trabajo, situando las herramientas cerca de los lugares donde se emplean y llevando a cabo labores de mantenimiento preventivo sobre la maquinaria. Las actividades de adaptación interna pueden reducirse simplificando o eliminando los ajustes.
- Realizar las actividades de adaptación en paralelo, o eliminarlas totalmente. Añadir una persona extra al equipo de adaptación puede reducir sensiblemente el tiempo de configuración. En muchos casos, el tiempo que tardan dos personas en hacer un trabajo es muy inferior a la mitad de lo que tardaría una sola.

Para analizar objetivamente el proceso de adaptación es útil confiar la labor de mejora a un equipo en el que colaboren operarios e ingenieros. Suele ser útil grabar en vídeo los procesos para intentar mejorarlos. Se puede recurrir a aplicar los estudios de tiempos y movimientos. Una vez ideadas las mejoras en los procedimientos, será necesario practicar hasta que se logre aplicarlos perfectamente. En Corea Steve Hulgirs inicia una actividad parecida.

Metodología TPM

El Mantenimiento Productivo Total (TPM, Total Productive Maintenance) es una adaptación del Mantenimiento Productivo occidental, al que los japoneses han añadido la palabra “Total” para especificar que el conjunto del personal de producción debe estar implicado en las acciones de mantenimiento y, asimismo, que deben ser integrados los aspectos relacionados con el mantenimiento de equipos, preparación de equipos, calidad, etc., que tradicionalmente se trataban de forma separada. Esta situación genera en los operarios un ambiente de responsabilidad en relación con la seguridad y el funcionamiento de su puesto de trabajo, involucrando a los trabajadores en tareas de mantenimiento, induciéndolos a prevenir averías y, en definitiva involucrándoles en el objetivo más general de la mejora continua.

Este enfoque de mantenimiento puede ponerse en práctica con rapidez y supone enseguida una reducción considerable de la falta de disponibilidad de las máquinas, al mismo tiempo que disminuye los niveles de errores, incrementa la productividad y reduce los costes.

Llevar un sistema estadístico y un Control Estadístico de Procesos para verificar la evolución y regularidad en la evolución de las máquinas forma parte también del TPM.

Producción uniforme

Para eliminar el desperdicio, los sistemas productivos JIT tratan de mantener un flujo de producción uniforme. Los cambios en la demanda final provocan fuertes variaciones en el ritmo de producción de la cadena de montaje final, que se trasladan multiplicadas a las células de producción de componentes. Las pequeñas variaciones en la demanda pueden ser absorbidas sin problemas por el sistema Kanban, como se explicará en el apartado 3. Sin embargo, cambios más bruscos terminan provocando la acumulación de existencias o la necesidad de establecer horas extras para poder cumplir con los objetivos de producción. Una vía para reducir la incertidumbre pasa por mejorar los pronósticos de la demanda. Otra alternativa consiste en intentar equilibrar, en la medida de lo posible, la producción a lo largo del horizonte de planificación. No se trata de producir la misma cantidad de cada producto todos los días, sino de mezclar pequeñas cantidades de distintos productos en la producción diaria. Así se consigue producir algo de cada artículo todos los días, con lo que se responde mejor a las variaciones en la demanda. Se logra también estabilizar la producción de componentes, reducir los niveles de inventario y apoyar al sistema jalar de producción.

Calidad en la fuente. Cero defectos

Para que el sistema JIT funcione adecuadamente, es preciso alcanzar niveles muy elevados de calidad. Las propias características del sistema promueven la elevación de los niveles de calidad. Así, la producción en pequeños lotes permite que los operarios detecten mejor los defectos e identifiquen sus causas. La meta es alcanzar el “cero defectos”, para lo que es preciso identificar los problemas de calidad en la fuente (también llamados rocas), resolverlos, y nunca dejar pasar un producto defectuoso. Con este fin, se traslada la responsabilidad sobre la calidad de los inspectores a los operarios, dándoles la potestad de ejercer jidoka, lo cual significa que tienen la autoridad para detener toda la cadena de montaje si se descubren problemas de calidad. Para promover el uso de esta facultad, todos los trabajadores tienen acceso a un interruptor que activa unas luces de emergencia o detiene el proceso productivo. Los problemas que surgen cada día se van anotando, y se reserva una parte de la jornada laboral al mantenimiento preventivo. Dedicar tiempo a planificar, formar, resolver problemas y mejorar el entorno del trabajo es clave para el éxito de la producción JIT.

Redes de proveedores

Disponer de una red de proveedores dignos de confianza es vital para el sistema JIT. Es necesario que los proveedores cumplan con exigentes requerimientos de calidad, y que se ubiquen en las proximidades de la empresa, para facilitar entregas frecuentes de pequeños lotes de partes o componentes. Una de las creencias más extendidas respecto a los sistemas JIT es que no eliminan la necesidad de mantener stocks, sino que solamente la desplazan hacia los proveedores. Esto sólo es cierto si los proveedores no aplican también el sistema. Si lo hacen correctamente, pueden aprovechar las ventajas derivadas de una demanda estable y segura, de los avisos previos respecto a variaciones en el volumen de producción, de la asistencia en cuestiones de ingeniería y administración, y en general, de los beneficios que se derivan de las estrechas relaciones cliente-proveedor que caracterizan a la producción justo a tiempo. Algunas de las tendencias recientes de las políticas de los proveedores son:

- Ubicarse cerca del cliente.
- Emplear camiones pequeños, de carga lateral, y realizar embarques conjuntos.
- Establecer pequeños almacenes cerca del cliente, o compartir los almacenes con otros proveedores.
- Emplear contenedores estandarizados y hacer las entregas de acuerdo con un programa de entregas preciso.
- Convertirse en un proveedor certificado, y aceptar cobrar por intervalos de tiempo en lugar de por entregas.

Mejora continua

La producción JIT es un sistema práctico, surgido del intento de eliminar el desperdicio y simplificar la producción mediante la aplicación del método de prueba y error. El último de los elementos que lo caracteriza, la mejora continua, es el más definitorio de todos, porque el JIT es un sistema que persigue optimizar permanentemente los niveles de inventario, los tiempos de adaptación, los niveles de calidad, etc. Por lo tanto, se puede decir que la producción ajustada es un sistema que se encuentra en una situación de permanente evolución, esto es, de mejora continua. Algunos de los elementos de esta mejora continua son:

- Control visual: tiene que ver con organizar los recursos que intervienen en el sistema productivo de manera que se pueda lograr que los problemas se adviertan

con mayor facilidad y que los trabajadores sean más conscientes de su ambiente de trabajo. La visib inútiles se eliminan y se asignan ubicaciones fijas para los objetos útiles (materiales, partes o herramientas), y los trabajadores se ocupan de cuidar escrupulosamente los equipos y herramientas, así como el propio espacio en que se desarrolla el trabajo y su función.

- Poka-Yoke: este concepto se refiere a los mecanismos o dispositivos simples que previenen la ocurrencia de problemas. Así, las máquinas que se detienen automáticamente después de producir un número establecido de unidades, o los sensores que impiden introducir demasiados artículos en un embalaje, son ejemplos de poka-yoke.
- Implicación total de los empleados: la mejora continua no es una cuestión que pueda dejarse en manos de un departamento o de un comité de expertos. Para lograrla, es precisa la implicación total de los empleados. La esencia misma del éxito de un sistema JIT reside en la predisposición de los trabajadores a señalar los problemas de calidad, a detener la producción cuando sea preciso, y para aportar ideas de mejora, analizar los procesos, realizar diversas funciones y modificar sus rutinas de trabajo. Para lograr niveles elevados de participación, las empresas deben adaptar su cultura corporativa y crear expectativas respecto a la implicación. Esto requiere formar a los empleados en técnicas para resolver los problemas, y darles la oportunidad de ponerlas en práctica. Es preciso, si no se quiere que el flujo de ideas se interrumpa, aplicar una proporción importante de las ideas que se aportan.

Principios de la mejora continua: los siguientes principios pueden ser de utilidad para iniciar o avanzar en el esfuerzo de la mejora continua:

- Crear una mentalidad para la mejora. Negar el status quo. Pensar en positivo, no en negativo. Las excusas no valen.
- Intentarlo una y otra vez: no hay que buscar la perfección a la primera. Las pequeñas mejoras son la base de las grandes. Actuar y después valorar los resultados. Corregir los errores tan pronto como se advierten.
- Pensar, no “adquirir” mejoras, cuestionarse el porqué de los problemas cuantas veces sea necesario.

- Trabajar en equipos. Con frecuencia, la creatividad de 10 personas puede superar al conocimiento de un solo individuo.
- Asumir que la mejora no tiene límites. No darse nunca por satisfecho. Habitarse a buscar formas mejores de hacer las cosas

9.2.4 Enfoque a cuellos de botella o teoría de las restricciones

Eliyahu Goldratt, físico israelí (igual que su personaje del libro, que poca imaginación tiene el pobre), propuso un sistema de secuenciación de carga infinita, con el nombre de OPT. Como idea principal equilibraba la producción centrándose en los cuellos de botella. Como cuerpo teórico, posteriormente expuso su TOC (Theory of Constraints).

Las reglas que propone son:

- Hay que equilibrar el flujo, no la capacidad.
- El nivel de utilización de un recurso no-cuello de botella no queda determinado por su potencial, sino por alguna otra restricción del sistema.
- Utilización y activación de un recurso no son sinónimos.
- Una hora perdida en un cuello de botella es una hora perdida en todo el sistema.
- Una hora ahorrada en un no-cuello de botella es un espejismo.
- Los cuellos de botella determinan volumen de ventas y los inventarios.
- Los lotes de transferencia y de proceso no tienen por qué ser iguales.
- Los lotes de proceso no deben de ser fijos.
- Todas las restricciones deben de ser consideradas simultáneamente: los tiempos de suministro no pueden ser predeterminados. La suma de los óptimos locales no coincide con el óptimo global.

Proceso

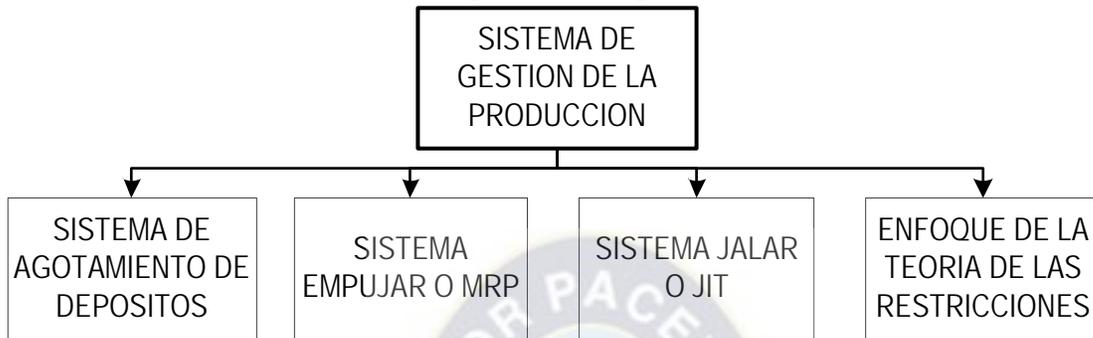
La esencia de la teoría de las restricciones se basa en cinco puntos correlativos de aplicación:

- Identificar los cuellos de botella del sistema.
- Decidir cómo explotarlos.
- Subordinar todo a la decisión anterior.
- Superar la restricción del sistema (elevar su capacidad).
- Si en los pasos anteriores se ha roto una restricción, regresar al paso pero no permitir la inercia.

El siguiente diagrama muestra los sistemas de gestión de la producción

DIAGRAMA 9-1

Sistema de gestión de la producción, 2014



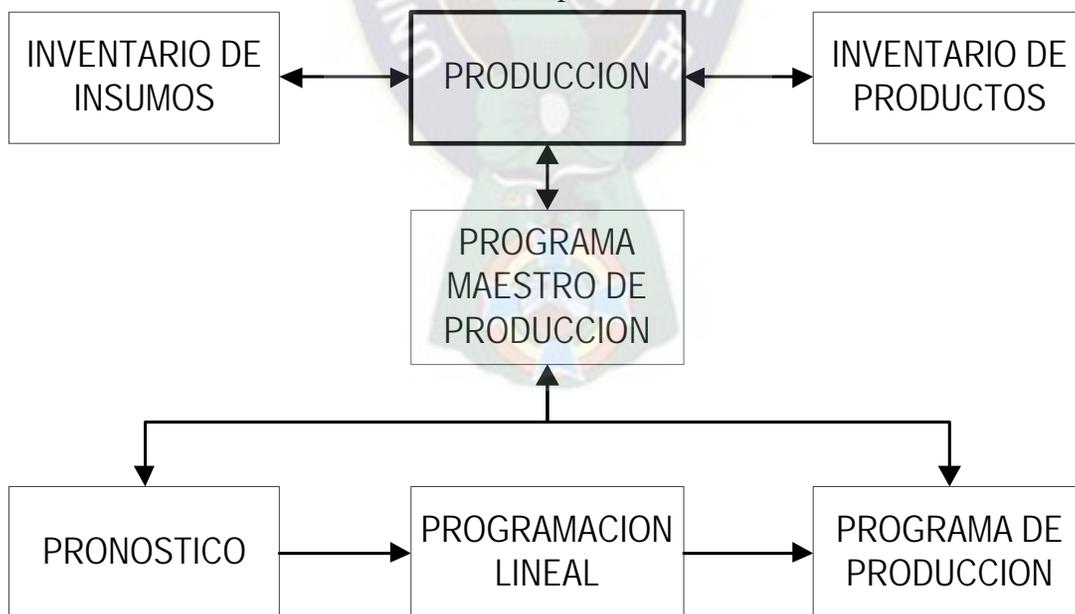
Fuente: Elaborado en base a (Gaither y Frazier 2008)

Para la empresa el sistema utilizado y el cual es la base del sistema SIS-K, es el sistema empujar o MRP.

9.3 Fases de la gestión de la producción

DIAGRAMA 9-2

SIS-K: Sistema de producción, 2014



Fuente: Elaborado en base a (Vargas Arce y Blanco Aguilar 2014)

9.3.1 Inventario de insumos

Esta herramienta permite controlar todos los insumos de la empresa como ser: carne vacuna, de cerdo, de pollo, tripas, sustancias curantes, hielo, etc. Contando con una cuadro que nos mostrara una pequeña descripción del insumo, el costo del insumo y la cantidad que hay en inventario del mismo. Este objeto sirve para controlar la entrada y salida de insumo.

Entradas

En primera instancia las entradas para este sub modulo son todos los ítems que tenemos en insumos, en segunda instancia este sub modulo está vinculado con el módulo de compras que incrementa de manera automática las cantidades de insumos con los que cuenta la empresa.

Proceso

Enlazado con el módulo de producción se modifican de forma automática la cantidad de insumos de acuerdo a los requerimientos del plan de producción

Salidas

Deberá generar un reporte que muestre las cantidades de insumos que se utilizaron en cada mes.

Este reporte es el mismo que el de gestión de proveedores, solo que este esta agrupado por insumo.

FIGURA 9-1

La española. Ejemplo de reporte de insumos, 2014

The screenshot shows a software window titled "Gestion de Insumos". At the top, there are navigation buttons: "Nuevo", "Modificar", and "Eliminar". Below this is a table titled "Insumos Registrados" with the following data:

Cod Insumo	Descripcion	Precio	Existencias
601	Conservantes	20,0000	1224
602	Came de res	30,0000	263
603	Came de pollo	14,0000	426
604	Came de cerdo	27,0000	634
605	Tripa Natural	13,0000	1045

Below the table is a search section titled "Buscar Insumo" with a dropdown menu for "Modo de Búsqueda" and a "Buscar" button. At the bottom, there is a "Datos del Insumo" section with the following fields:

Cod Insumo:	601	Existencia fisica:	1224
Descripcion:	Conservantes	Saldo Inventario:	4480000
Costo:	200000		

Fuente: Elaboración propia

9.3.2 Inventario de productos

Esta herramienta permite controlar los productos de la empresa como ser: salchichas, chorizos, mortadela, jamón, salame, etc. Este contara con un cuadro que facilite la visualización de la descripción del producto, precio de venta, cantidad con que se cuenta en inventario. Este objeto sirve para controlar la entrada y salida de producto terminado.

Entradas

En primera instancia las entradas para este sub modulo son toda la gama de productos con que la empresa cuenta, en segunda instancia este sub modulo estará vinculado con el módulo de producción incrementando sus existencias de acuerdo a lo que la empresa produzca.

Proceso

Enlazado con el módulo de ventas se modifican de forma automática la cantidad de productos de acuerdo a los pedidos de nuestros clientes.

Salidas

Deberá generar un reporte que muestre las cantidades de productos que se vendieron en cada mes, especificando que persona realizó la venta.

Este reporte es el mismo que el de gestión de ventas, solo que este esta agrupado por producto.

FIGURA 9-2

La española. Ejemplo de reporte de producto terminado, 2014

Cod Producto	Producto	Precio	Existencias
201	Salchicha viena	35,7000	49890
202	Chorizo	40,0000	2788
203	Mortadela	41,0000	900
204	Jamon	46,0000	2000
205	Salame	49,0000	500

Buscar Producto

Modo de Búsqueda: Buscar

Datos del Producto

Cod Producto: 201 Existencias: 49890
Descripcion: Salchicha viena Saldo Inventario: 1785000,0000
Precio: 35,7000

Fuente: Elaboración propia

9.3.3 Programa maestro de producción

Esta es la herramienta central de la aplicación informática, ya que cumple con el objetivo general del proyecto. Esta herramienta tiene tres componentes: el primero es Pronóstico, este componente estima la demanda en base a un método que estableceremos en el CAPÍTULO 10. INGENIERÍA DEL PROYECTO; el segundo es la programación lineal, ese componente analiza la demanda (D_t) por producto, los coeficientes de rendimientos de los productos, el precio de los productos, el costo de los productos, la disponibilidad de mano de obra y la disponibilidad de insumos para obtener una programación que optimice las ganancias brutas de la empresa (Z); el último es programa de producción, este genera una tabla que sirve de guía al área de producción, para cumplir con el programa.

Entradas Pronóstico

Esta herramienta requiere como motor de datos a una tabla en MS Office Excel que tiene los datos históricos de la demanda por producto de por lo menos veinticuatro meses atrás.

Proceso Pronóstico

Esta herramienta requiere como análisis de datos al servicio de análisis SOLVER de MS Office Excel. Esta herramienta debe ser actualizada cada doce meses.

Salidas Pronóstico

Deberá generar un reporte con las estimaciones de demanda de los diferentes productos para los siguientes doce meses. También deberá mostrar los datos históricos de la demanda por producto.

Entradas Programación lineal

Esta herramienta requiere como motor de datos a una tabla en MS Office Excel que tiene un modelo de programación lineal para el programa de producción que se modelara en el CAPÍTULO 11. DISEÑO DE MÉTODOS. Para este modelo de programación lineal requerimos de: el precio de productos (P), el costo de producción diurna por producto (cd), el costo de producción nocturna por producto (cn), con esto anterior obtenemos la ganancia por productos y horario (a), la combinación lineal de esta ganancia con la cantidad a producir de cada producto no da las ganancias brutas de la empresa (Z). Las restricciones están sujetas a la combinación lineal de los coeficientes de rendimientos que se calcularán en el CAPÍTULO 10. INGENIERÍA DEL PROYECTO y la cantidad a producir por

producto limitado por las horas disponibles por sección. Las restricciones faltantes son la cantidad a producir por producto condicionado por la demanda de estos.

Este modelo debe ser flexible para dinamizar el sistema ya que los coeficientes están en constante cambio. Por lo que debe dar la posibilidad de cambiar los coeficientes de la función objetivo y las limitantes de las restricciones.

Proceso Programación lineal

Esta herramienta requiere como análisis de datos al servicio de análisis SOLVER de MS Office Excel. Esta herramienta soluciona el modelo de programación lineal propuesto en segundo plano.

Salidas Programación lineal

Deberá generar un reporte con las cantidades a producir de los productos.

Entrada Programa de producción

Esta herramienta requiere como motor de datos al anterior reporte de las cantidades a producir por producto.

Proceso Programación lineal

Esta herramienta requiere como análisis de datos al servicio de MS Office Excel que filtra la cantidad a producir por producto por fecha.

Salidas Programación lineal

Deberá generar un reporte con una tabla que muestre el programa de producción por fecha.

CAPÍTULO 10. INGENIERÍA DEL PROYECTO

10.1 Introducción

Como mencionamos anteriormente necesitamos calcular los parámetros de la empresa en las dimensiones que mencionamos. En este capítulo calcularemos esas variables y las validaremos, y en el siguiente capítulo volveremos a hacer correr el sistema para verificar los valores óptimos e implementar al sistema informático.

10.1.1 Definiciones

Medición del trabajo, es la medición de los tiempos de producción.

Medición de la capacidad, es la medición de las capacidades de producción.

10.2 Bases para la ingeniería del proyecto

Estudio del producto, es el análisis de las características técnicas del producto en intersección con los requerimientos públicos, para sintetizar internamente en un diseño final.

Estudio de procesos, es el análisis de los métodos de transformación en intersección con los requerimientos de los propietarios, para sintetizar en un diseño de planta final.

10.2.1 Objetivo

Calcular los parámetros de rendimiento para la programación de la producción.

10.2.2 Modelo de ingeniería del proyecto

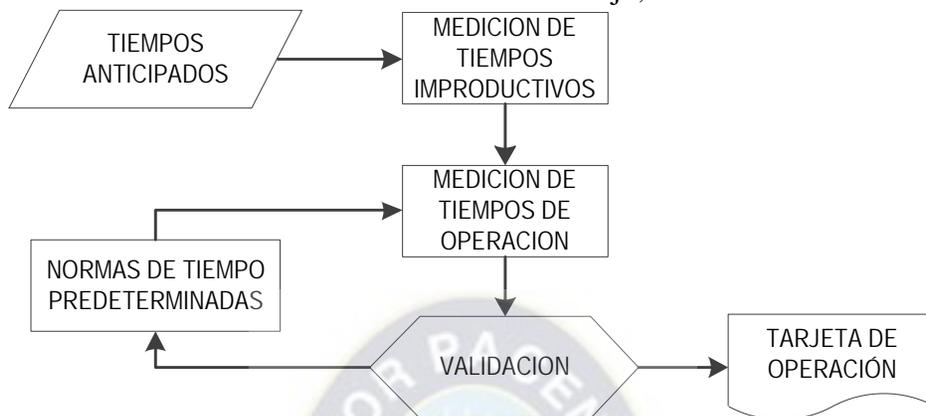
Para la medición del trabajo vamos a calcular los tiempos y movimientos de los elementos de las operaciones. Esto nos va a servir para la medición de la capacidad.

Para la medición de la capacidad de trabajo vamos a medir los rendimientos de las máquinas y las limitaciones de capacidad. Esto sirve para el capítulo de desarrollo de métodos.

10.3 Fases de la medición del trabajo

DIAGRAMA 10-1

Fases de la medición del trabajo, 2014



Fuente: Elaborado en base a (Niebel y Freidvalds 2004) y (Organizacion Internacional del Trabajo 1996)

10.3.1 Definiciones

Medición de tiempos improductivos.- Es la cuantificación de la productividad de las secciones en un porcentaje.

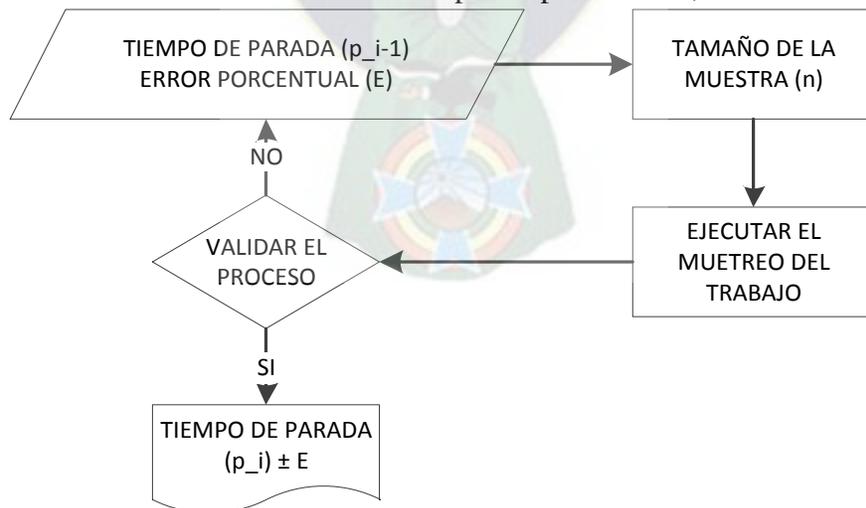
Medición de tiempos de operación.- Es la cuantificación de los tiempos de las tareas.

Normas de tiempo predeterminadas.- Son estándares de movimientos.

10.4 Fases de la medición de tiempos improductivos

DIAGRAMA 10-2

Fases de medición de tiempos improductivos, 2014

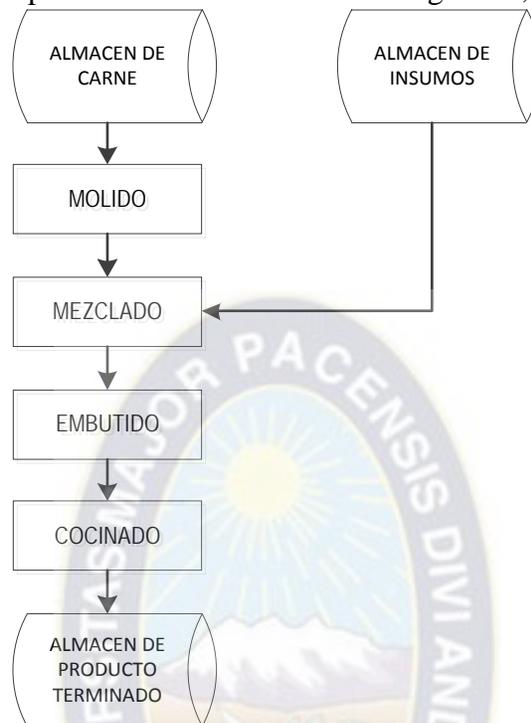


Fuente: Elaborado en base a (Niebel y Freidvalds 2004) y (Organizacion Internacional del Trabajo 1996)

10.4.1 Medición de tiempos improductivos

DIAGRAMA 10-3

La Española: Proceso de fabricación general, 2014



Fuente: Elaborado en base a Gerencia de Producción

Tamaño de la muestra

El tamaño de la muestra (n) es igual al coeficiente ($Z_{\alpha/2}$) elevado a la segunda potencia, por la proporción de tiempo productivo anticipado (\hat{p}_{i-1}), por la proporción de tiempo improductivo anticipado ($1 - \hat{p}_{i-1}$), sobre el error porcentual (E) elevado a la segunda potencia.¹⁷

$$n = \frac{Z_{\alpha/2}^2 * \hat{p}_{i-1} * (1 - \hat{p}_{i-1})}{E^2}$$

En consenso con los directivos de la empresa determinamos el nivel de significancia (α) de 2% y el error porcentual (E) de 5%.

Para la sección de molido, que tiene una proporción de tiempo productivo anticipado igual a 0,9 el tamaño de muestra es:

¹⁷ En esta ocasión vamos a utilizar la fórmula propuesta por (Organización Internacional del Trabajo 1996), la cual es validada por el Ing. Juan Carlos Martínez López

$$n = \frac{5,411894431 * 0,9 * (1 - 0,9)}{0,05^2}$$

$$n = 194,8281995$$

$$n \cong 195$$

En la siguiente tabla se muestran los tamaños de muestra para las diferentes secciones de la planta:

CUADRO 10-1

La Española: Tamaño de muestra anticipado, 2014

SECCIÓN	PROPORCIÓN DE TIEMPO PRODUCTIVO ANTICIPADO	TAMAÑO DE MUESTRA
MOLIDO	0,9	195
MEZCLADO	0,8	347
EMBUTIDO	0,6	520
COCINADO	0,85	277

Fuente: Elaborado en base a la Ecuación tamaño de la muestra

Cabe recalcar que las operaciones de trozado y molido son secuenciales y por consiguiente llevan la misma proporción de tiempo productivo conjunta. Las operaciones de ahumado, cocinado y enfriado son secuenciales y por consiguiente llevan la misma proporción de tiempo productivo.

Sección de molido

Para esta operación se deben generar 195 números aleatorios entre 0 y 1, distribuidos en 30 días calendario entre el 2 de Enero al 31 de Enero del presente año, a través de la aplicación (Microsoft Office Excel 2013) y vamos a validarlos con estadística no paramétrica. Los números aleatorios se muestran en la tabla de ANEXO A.

1. Prueba de frecuencias para distribución uniforme con Chi-Cuadrada¹⁸

i. Formulación de la hipótesis

H₀: Los valores se distribuyen uniformemente

H_a: Los valores no se distribuyen uniformemente

ii. Determinación del número de clases (k)

$$k \geq \log_2 n$$

$$k \geq \log_2 195$$

¹⁸ Esta prueba también es conocida como prueba de bondad de ajuste.

$$k \geq 7,60733$$

$$k = 8$$

iii. Estadístico de prueba ($\chi^2_{calculada}$)

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i}$$

CUADRO 10-2

La Española: Estadístico de Puebla, 2014

Intervalo	Ei	Oi	(Oi-Ei)^2/Ei
1	24,4	13	5,30833
2	24,4	29	0,87756
3	24,4	23	0,07756
4	24,4	22	0,23141
5	24,4	28	0,53910
6	24,4	23	0,07756
7	24,4	22	0,23141
8	24,4	35	4,63141
Suma			11,97436

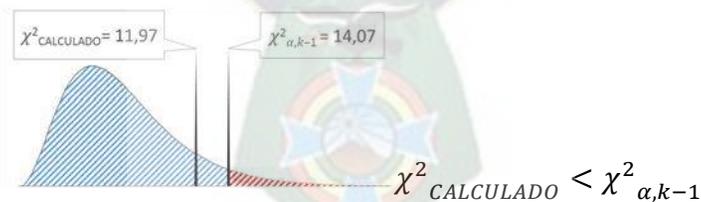
Fuente: Elaboración de propia

$$\chi^2 = 11,97436$$

iv. Estadístico teórico ($\chi^2_{\alpha,k-1}$)

$$\chi^2_{\alpha,k-1} = 14,06714$$

v. Contraste



vi. Conclusión

Para un nivel de significancia del 5%, podemos aceptar que la secuencia de números aleatorios se distribuye uniformemente.

2. Prueba general de autocorrelación

i. Especificación del modelo econométrico de la teoría

$$u_t = \rho_0 u_{t-1} + \rho_1 u_{t-2} + \rho_2 u_{t-3} + \rho_3 u_{t-4} + \rho_4 u_{t-5} + \rho_5 u_{t-6} + \rho_6 u_{t-7} + \varepsilon_t^{19}$$

¹⁹ Para este modelo suponemos que u_t es igual a cero, pero esto nos presenta un problema de estimación, por lo tanto vamos a darle un valor tan pequeño igual a 0,001

ii. Planteamiento de la hipótesis

$$H_0: \rho_0 = \rho_1 = \rho_2 = \rho_3 = \rho_4 = \rho_5 = \rho_6 = 0$$

$$H_a: \rho_0 \neq \rho_1 \neq \rho_2 \neq \rho_3 \neq \rho_4 \neq \rho_5 \neq \rho_6 \neq 0$$

iii. Estimación de los parámetros

CUADRO 10-3

La Española. Estimación de Parametros, 2014

ρ_i	-91,70E-9	343,64E-9	167,62E-9	-48,51E-9	-88,95E-9	112,49E-9	-491,84E-9
σ_{ρ_i}	(2,54E-6)	(2,59E-6)	(2,55E-6)	(2,55E-6)	(2,56E-6)	(2,56E-6)	(2,54E-6)
$R^2 = 0,00031776$		$\sigma_\varepsilon = 1,02E - 05$					
$F = 0,00821907$		$n - k = 181$					
$SCR = 5,97E - 12$		$SCE = 1,88E - 08$					

Fuente: Elaboración de propia

iv. Estadístico de prueba

$$t = \frac{\rho_t - 0}{\sigma_{\rho_t}}$$

CUADRO 10-4

La Española. Estadístico de prueba, 2014

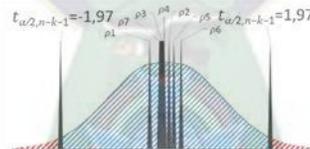
t	-0,19	0,04	-0,03	-0,02	0,07	0,13	-0,04
-----	-------	------	-------	-------	------	------	-------

Fuente: Elaboración de propia

v. Estadístico de prueba

$$t_{\alpha/2, n-k-1} = t_{0,975, 180} = 1,96$$

vi. Contraste



vii. Conclusión

Para un nivel de significancia del 5%, podemos aceptar que todos los parámetros de estimación son iguales a cero por lo tanto no existe autocorrelación.

3. Prueba de signos

Cuando un valor sea mayor al anterior le asignamos (+), caso contrario le asignamos (-).

i. Formulación de la hipótesis

H_0 : La secuencia de rachas R no tiene autocorrelación negativa

Sección de mezclado

Para esta operación se deben generar 347 números aleatorios entre 0 y 1, distribuidos en 30 días calendario entre el 1° de Febrero al 2 de Marzo del presente año, a través de la aplicación de (Microsoft Office Excel 2013) y vamos a validarlos con estadística no paramétrica. Los números aleatorios se muestran en la tabla de ANEXO A.

1. Prueba de frecuencias para distribución uniforme con Kolmogorov-Smirnov

i. Formulación de la hipótesis

H_0 : Los valores R_i se distribuyen uniformemente

H_a : Los valores R_i no se distribuyen uniformemente

ii. Estadístico de prueba $\text{Max}\{D^+, D^-\}$

$$D^+ = \text{Max}_{1 \leq i \leq N} \left[\frac{i}{N} - R_i \right], D^- = \text{Max}_{1 \leq i \leq N} \left[R_i - \frac{i-1}{N} \right]$$

$$\text{Max}\{D^+ = 0,03687; D^- = 0,03399\}$$

$$D = 0,03687$$

iii. Estadístico teórico (Kolmogorov – Smirnov)

$$D_{0,05;347} = 1,36$$

iv. Contraste

$$D_{\text{CALCULADO}} < D_{0,05;347}$$

v. Conclusión

Para un nivel de significancia del 5%, podemos aceptar que la secuencia de números aleatorios se distribuye uniformemente.

2. Prueba general de autocorrelación

i. Especificación del modelo econométrico de la teoría

$$u_t = \rho_0 u_{t-1} + \rho_1 u_{t-2} + \rho_2 u_{t-3} + \rho_3 u_{t-4} + \rho_4 u_{t-5} + \rho_5 u_{t-6} + \rho_6 u_{t-7} + \varepsilon_t$$

ii. Planteamiento de la hipótesis

$$H_0: \rho_0 = \rho_1 = \rho_2 = \rho_3 = \rho_4 = \rho_5 = \rho_6 = 0$$

$$H_a: \rho_0 \neq \rho_1 \neq \rho_2 \neq \rho_3 \neq \rho_4 \neq \rho_5 \neq \rho_6 \neq 0$$

iii. Estimación de los parámetros

CUADRO 10-5

La Española: Estimación de Parametros, 2014

ρ_i	-100,07E-9	-109,97E-9	-78,16E-9	-97,38E-9	-62,86E-9	160,90E-9	126,98E-9
σ_{ρ_i}	1,84E-6	1,85E-6	1,82E-6	1,82E-6	1,86E-6	1,84E-6	1,83E-6
$R^2 = 74,39E - 6$				$\sigma_\varepsilon = 10,10E - 6$			
$F = 3,54E - 03$				$n - k = 333$			
$SCR = 2,53E - 12$				$SCE = 3,40E - 08$			

Fuente: Elaboración de propia

iv. Estadístico de prueba

$$t = \frac{\rho_t - 0}{\sigma_{\rho_t}}$$

CUADRO 10-6

La Española: Estadístico de prueba, 2014

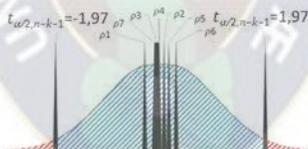
t	0,07	0,09	-0,03	-0,05	-0,04	-0,06	-0,05
-----	------	------	-------	-------	-------	-------	-------

Fuente: Elaboración de propia

v. Estadístico de prueba

$$t_{\alpha/2, n-k-1} = t_{0,975, 180} = 1,96$$

vi. Contraste



vii. Conclusión

Para un nivel de significancia del 5%, podemos aceptar que todos los parámetros de estimación son iguales a cero por lo tanto no existe autocorrelación.

3. Prueba de signos

i. Formulación de la hipótesis

H_0 : La secuencia de rachas R no tiene autocorrelación negativa

H_a : La secuencia de rachas R tiene autocorrelación negativa

ii. Estadístico de prueba

Racha positiva es una sucesión de eventos (N+), racha negativa es una sucesión (N-)

$$N = 183 \quad N^+ = 175 \quad N^- = 172$$

$$\mu_N = \frac{2N^+N^-}{N^+ + N^-} + 1 = 174,49$$

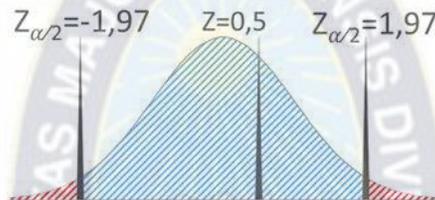
$$\sigma_N = \sqrt{\frac{2N^+N^-(N^+N^- - N)}{N^2(N - 1)}} = 17,19$$

$$Z = \frac{N - \mu_N}{\sigma_N} = 0,50$$

iii. Estadístico teórico

$$|Z_{\alpha/2}| = 1,96$$

iv. Contraste



v. Conclusión

Para un nivel de significancia del 5%, podemos aceptar que la secuencia de números aleatorios no tiene autocorrelación negativa.

Conclusión final

Para un nivel de confianza de 5% podemos aceptar que los números aleatorios cumplen las propiedades de uniformidad e independencia.

Validados los números aleatorios vamos a generar la tabla de 11 a 12 observaciones como se muestra en la tabla de ANEXO A.

Sección de embutido

Para esta operación se deben generar 520 números aleatorios entre 0 y 1, distribuidos en 30 días calendario entre el 1° de Febrero al 2 de Marzo del presente año, a través de la aplicación de (Microsoft Office Excel 2013) y vamos a validarlos con estadística no paramétrica. Los números aleatorios de muestran en la tabla de ANEXO A.

1. Prueba de frecuencias para distribución uniforme con Chi-Cuadrada

i. Formulación de la hipótesis

H_0 : Los valores se distribuyen uniformemente

H_a : Los valores no se distribuyen uniformemente

ii. Determinación del número de clases (k)

$$k \geq \log_2 n$$

$$k \geq \log_2 520$$

$$k \geq 9,02 \cong 10$$

iii. Estadístico de prueba ($\chi^2_{calculada}$)

$$\chi^2 = \sum \frac{(O_i - E_i)^2}{E_i}$$

CUADRO 10-7

La Española: Estadístico de prueba, 2014

Intervalo	Ei	Oi	(Oi-Ei)^2/Ei
1	52,0	57	0,48077
2	52,0	56	0,30769
3	52,0	47	0,48077
4	52,0	56	0,30769
5	52,0	50	0,07692
6	52,0	57	0,48077
7	52,0	50	0,07692
8	52,0	42	1,92308
9	52,0	49	0,17308
10	52,0	56	0,30769
Suma			4,615384615

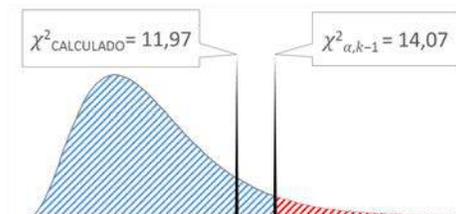
Fuente: Elaboración de propia

$$\chi^2 = 11,97436$$

iv. Estadístico teórico ($\chi^2_{\alpha,k-1}$)

$$\chi^2_{\alpha,k-1} = 14,06714$$

v. Contraste



$$\chi^2_{CALCULADO} < \chi^2_{\alpha,k-1}$$

vi. Conclusión

Para un nivel de significancia del 5%, podemos aceptar que la secuencia de números aleatorios se distribuye uniformemente.

2. Prueba general de autocorrelación

i. Especificación del modelo econométrico de la teoría

$$u_t = \rho_0 u_{t-1} + \rho_1 u_{t-2} + \rho_2 u_{t-3} + \rho_3 u_{t-4} + \rho_4 u_{t-5} + \rho_5 u_{t-6} + \rho_6 u_{t-7} + \varepsilon_t$$

ii. Planteamiento de la hipótesis

$$H_0: \rho_0 = \rho_1 = \rho_2 = \rho_3 = \rho_4 = \rho_5 = \rho_6 = 0$$

$$H_a: \rho_0 \neq \rho_1 \neq \rho_2 \neq \rho_3 \neq \rho_4 \neq \rho_5 \neq \rho_6 \neq 0$$

iii. Estimación de los parámetros

CUADRO 10-8

La Española: Estimación de los Parámetros, 2014

ρ_i	3,39E-6	-1,36E-6	-14,86E-6	-1,69E-6	-6,68E-6	6,63E-6	11,27E-6
σ_{ρ_i}	141,25E-6	140,08E-6	140,37E-6	140,01E-6	140,13E-6	141,37E-6	141,57E-6
$R^2 = 3,90E - 05$		$\sigma_\varepsilon = 1,01E - 03$					
$F = 0,00281669$		$n - k = 506$					
$SCR = 2,00E - 08$		$SCE = 5,13E - 04$					

Fuente: Elaboración de propia

iv. Estadístico de prueba

$$t = \frac{\rho_t - 0}{\sigma_{\rho_t}}$$

CUADRO 10-9

La Española: Estadístico de prueba, 2014

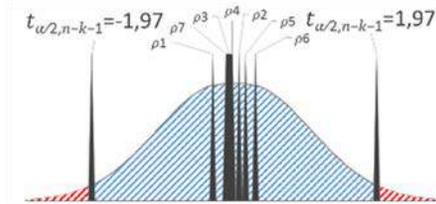
t	0,08	0,05	-0,05	-0,01	-0,11	-0,01	0,02
-----	------	------	-------	-------	-------	-------	------

Fuente: Elaboración de propia

v. Estadístico de prueba

$$t_{\alpha/2, n-k-1} = t_{0,975, 180} = 1,96$$

vi. Contraste



vii. Conclusión

Para un nivel de significancia del 5%, podemos aceptar que todos los parámetros de estimación son iguales a cero por lo tanto no existe autocorrelación.

3. Prueba de signos

i. Formulación de la hipótesis

H_0 : La secuencia de rachas R no tiene autocorrelación negativa

H_a : La secuencia de rachas R tiene autocorrelación negativa

ii. Estadístico de prueba

Racha positiva es una sucesión de eventos (N^+), racha negativa es una sucesión de eventos (N^-)

$$N = 283 \quad N^+ = 254 \quad N^- = 266$$

$$\mu_N = \frac{2N^+N^-}{N^+ + N^-} + 1 = 260,86$$

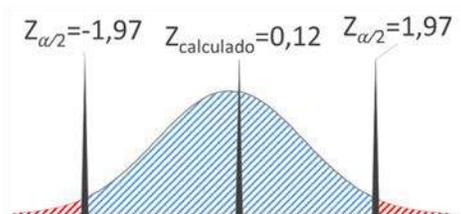
$$\sigma_N = \sqrt{\frac{2N^+N^-(N^+N^- - N)}{N^2(N-1)}} = 20,06$$

$$Z = \frac{N - \mu_N}{\sigma_N} = 1,10$$

iii. Estadístico teórico

$$|Z_{\alpha/2}| = 1,96$$

iv. Contraste



v. Conclusión

Para un nivel de significancia del 5%, podemos aceptar que la secuencia de números aleatorios no tiene autocorrelación negativa.

Conclusión final

Para un nivel de confianza de 5% podemos aceptar que los números aleatorios cumplen las propiedades de uniformidad e independencia.

Validados los números aleatorios vamos a generar la tabla de 17 a 18 observaciones como se muestran en la tabla de ANEXOS A.

Sección de cocinado

Para esta operación se deben generar 277 números aleatorios entre 0 y 1, distribuidos en 30 días calendario entre el 2 de Enero al 31 de Enero del presente año, a través de la aplicación de (Microsoft Office Excel 2013) y vamos a validarlos con estadística no paramétrica. Los números aleatorios de muestran en la tabla de ANEXO A.

1. Prueba de frecuencias para distribución uniforme con Kolmogorov-Smirnov

i. Formulación de la hipótesis

H_0 : Los valores R_i se distribuyen uniformemente

H_a : Los valores R_i no se distribuyen uniformemente

ii. Estadístico de prueba $Max\{D^+, D^-\}$

$$D^+ = Max_{1 \leq i \leq N} \left[\frac{i}{N} - R_i \right], D^- = Max_{1 \leq i \leq N} \left[R_i - \frac{i-1}{N} \right]$$

$$Max\{D^+ = 0,71816; D^- = 0,71528\}$$

$$D = 0,71816$$

iii. Estadístico teórico (Kolmogorov – Smirnov)

$$D_{0,05;276} = 1,36$$

iv. Contraste

$$D_{CALCULADO} < D_{0,05;276}$$

v. Conclusión

Para un nivel de significancia del 5%, podemos aceptar que la secuencia de números aleatorios se distribuye uniformemente.

2. Prueba general de autocorrelación

i. Especificación del modelo econométrico de la teoría

$$u_t = \rho_0 u_{t-1} + \rho_1 u_{t-2} + \rho_2 u_{t-3} + \rho_3 u_{t-4} + \rho_4 u_{t-5} + \rho_5 u_{t-6} + \rho_6 u_{t-7} + \varepsilon_t$$

ii. Planteamiento de la hipótesis

$$H_0: \rho_0 = \rho_1 = \rho_2 = \rho_3 = \rho_4 = \rho_5 = \rho_6 = 0$$

$$H_a: \rho_0 \neq \rho_1 \neq \rho_2 \neq \rho_3 \neq \rho_4 \neq \rho_5 \neq \rho_6 \neq 0$$

iii. Estimación de los parámetros

CUADRO 10-10

La Española: Estimación de los Parametros, 2014

ρ_i	23,79E-9	-585,30E-9	-335,72E-9	-449,76E-9	-64,03E-9	-297,39E-9	-339,01E-9
σ_{ρ_i}	2,00E-6	2,07E-6	1,97E-6	2,01E-6	1,96E-6	2,05E-6	2,01E-6
$R^2 = 4,09E - 3$				$\sigma_\varepsilon = 10,11E - 6$			
$F = 0,154$				$n - k = 263$			
$SCR = 1,10E - 10$				$SCE = 2,69E - 08$			

Fuente: Elaboración de propia

iv. Estadístico de prueba

$$t = \frac{\rho_t - 0}{\sigma_{\rho_t}}$$

CUADRO 10-11

La Española: Estadístico de prueba, 2014

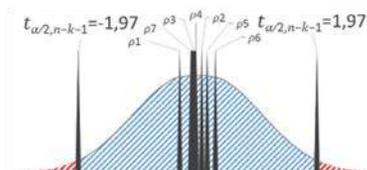
t	0,01	-0,28	-0,17	-0,22	-0,03	-0,14	-0,17
-----	------	-------	-------	-------	-------	-------	-------

Fuente: Elaboración de propia

v. Estadístico de prueba

$$t_{\alpha/2, n-k-1} = t_{0,975, 180} = 1,96$$

vi. Contraste



vii. Conclusión

Para un nivel de significancia del 5%, podemos aceptar que todos los parámetros de estimación son iguales a cero por lo tanto no existe autocorrelación.

3. Prueba de signos

i. Formulación de la hipótesis

H_0 : La secuencia de rachas R no tiene autocorrelación negativa

H_a : La secuencia de rachas R tiene autocorrelación negativa

ii. Estadístico de prueba

Racha positiva es una sucesión de eventos (N^+), racha negativa es una sucesión de eventos (N^-)

$$N = 147 \quad N^+ = 141 \quad N^- = 136$$

$$\mu_N = \frac{2N^+N^-}{N^+ + N^-} + 1 = 139,45$$

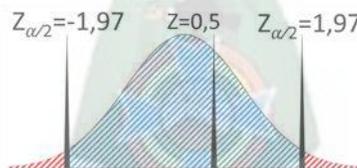
$$\sigma_N = \sqrt{\frac{2N^+N^-(N^+N^- - N)}{N^2(N - 1)}} = 15,21$$

$$Z = \frac{N - \mu_N}{\sigma_N} = 0,50$$

iii. Estadístico teórico

$$|Z_{\alpha/2}| = 1,96$$

iv. Contraste



v. Conclusión

Para un nivel de significancia del 5%, podemos aceptar que la secuencia de números aleatorios no tiene autocorrelación negativa.

Conclusión final

Para un nivel de confianza de 5% podemos aceptar que los números aleatorios cumplen las propiedades de uniformidad e independencia.

Validados los números aleatorios vamos a generar la tabla de 11 a 12 observaciones como se muestra en la tabla de ANEXOS A.

La siguiente tabla nos muestra el resumen del estudio de tiempos de parada.

CUADRO 10-12

La española: Resumen de estudio de paradas, 2014

MEZCLADO	MOLIDO	MEZCLA DO	EMBUTI DO	COCINA DO
TIEMPO PRODUCTIVO	87,2%	81,0%	64,2%	87,4%
TIEMPO IMPRODUCTIVO	12,8%	19,0%	35,8%	12,6%
NECESIDADES PERSONALES DEL TRABAJADOR	6,2%	8,6%	6,2%	6,1%
ESPERA SUMINISTROS EN MANTENIMIENTO	2,6%	5,2%	2,3%	2,5%
INACTIVA	4,1%	2,3%	24,2%	1,1%

Fuente: Elaborado en base al estudio de tiempos de parada ANEXO C

10.5 Fases de la medición de tiempos de operación

DIAGRAMA 10-4

Fases de la medición de tiempos de operación, 2014



Fuente: Elaborado con base en (Niegel y Freidvalds 2004)

10.5.1 Medición del tiempo de los elementos de las tareas

Tamaño de la muestra para las tareas

El tamaño de muestra (n) es igual al estadístico t de student ($t_{\alpha/2}$) por la desviación estándar de la muestra (S_x) por 100 todo lo anterior sobre el promedio de la muestra (\bar{x}) por el error relativo porcentual ($\% \varepsilon$) todo lo anterior elevado al cuadrado.

$$n = \left(\frac{t_{\alpha/2, v} * S_x * 100}{\bar{x} * \% \varepsilon} \right)^2$$

En consenso con los directivos de la empresa determinamos el nivel de significancia (α) de 5% y el error relativo porcentual ($\% \varepsilon$) de 8% para todas las operaciones.

Como no se tienen datos de la media poblacional y la variación poblacional, vamos a tomar una muestra de tamaño 5 en todas las operaciones solo de los tiempos totales de operación para tener nuestros valores de la media anticipada y la desviación estándar anticipada.²⁰

Los elementos principales de las operaciones son: trozado, molido, preparado de aditivos, mezclado, embutido, ahumado, cocinado y enfriado.²¹

CUADRO 10-13

La española: Tiempos anticipados al Estudio de tiempos y movimientos, 2014

Tarea	Muestra					Promedio [min]	Desviación estándar [min]
	1	2	3	4	5		
Trozado	11:36	10:28	10:22	8:45	13:3	10:51	1:36
Molido	42:52	31:39	32:58	35:57	33:3	35:18	4:31
Preparación de aditivos	1:33	1:17	1:21	1:11	1:36	1:23	0:10
Mezclado	21:24	29:30	19:36	30:6	23:12	84:46	4:47
Ebutido	8:14	17:59	26:42	8:20	21:32	76:34	8:10
Ahumado	55:6	6:28	14:12	1:36	7:27	124:58	7:7
Cocinado	17:30	16:6	18:54	16:6	17:54	17:18	1:12
Enfriado	2:6	3:22	2:37	3:11	2:47	2:49	0:30

Fuente: Elaborado en base al procedimiento sugerido por Ing. Juan Carlos Martínez López

²⁰ Este procedimiento no es estándar, (Niebel y Freidvalds 2004) sugiere utilizar otra fórmula que no convence a los autores por lo tanto vamos a consultar a Ing. Juan Carlos Martínez López, especialista en Laboratorio de Física, que recomienda este procedimiento.

²¹ No estamos considerando el elemento de curado por ser este un elemento extraño, así como también no estamos considerando elementos de las operaciones de almacenaje por no ser de importancia crucial en el estudio.

Entonces para calcular el tamaño de muestra tenemos que iterar, para la operación de trozado el tamaño de muestra es:

$$n = \left(\frac{t_{1\%,v} * 1,6 * 100}{10,8 * 10} \right)^2$$

$$n = t_{1\%,v}^2 * 2,1746$$

CUADRO 10-14

La española: Tamaño de muestra real para la operación de trozado, 2014

Tamaño de muestra	Grados de libertad	Estadístico t	Tamaño aparente de la muestra
2	1	12,7062047	548
3	2	4,30265273	62
4	3	3,18244631	34
5	4	2,77644511	26
6	5	2,57058184	22
7	6	2,44691185	20
8	7	2,36462425	18
9	8	2,30600414	18
10	9	2,26215716	17
11	10	2,22813885	16
12	11	2,20098516	16
13	12	2,17881283	16
14	13	2,16036866	15
15	14	2,14478669	15

Fuente: Elaborado en base a (Alvarez C y Huayta C 2007) y a la Ecuación de medición de tiempos

La primera columna es un número generado de manera correlativa, la segunda columna son los grados de libertad que equivalen a el tamaño de muestra menos 1, le tercera columna es la inversa acumulada del estadístico t de student con los grados de libertad de la columna 2 y el nivel de significancia predeterminado de 5% y la última columna es la t de student calculada en la columna 3, para el cálculo final según la ecuación. Entonces tenemos que hacer esta operación hasta que los tamaños de muestra coincidan en el mismo número.

Para la tarea de trozado el tamaño de muestra es 15²², para considerar el error relativo porcentual mínimo vamos a considerar el máximo el tamaño de muestra inicial. En caso

²² El tamaño de muestra equivale a la cantidad de ciclos a medir.

que el tamaño de muestra excediere los 30 datos se debe volver a calcular con la función normal estándar.

CUADRO 10-15

La española: Tamaño de muestra para las operaciones, 2014

Tarea	Cantidad de ciclos
Trozado	15
Molido	12
Preparación de aditivos	11
Mezclado	5
Embutido	5
Ahumado	5
Cocinado	5
Enfriado	21

Fuente: Elaborado con base en los datos del CUADRO 3-3 y el procedimiento de (Alvarez C y Huayta C 2007)

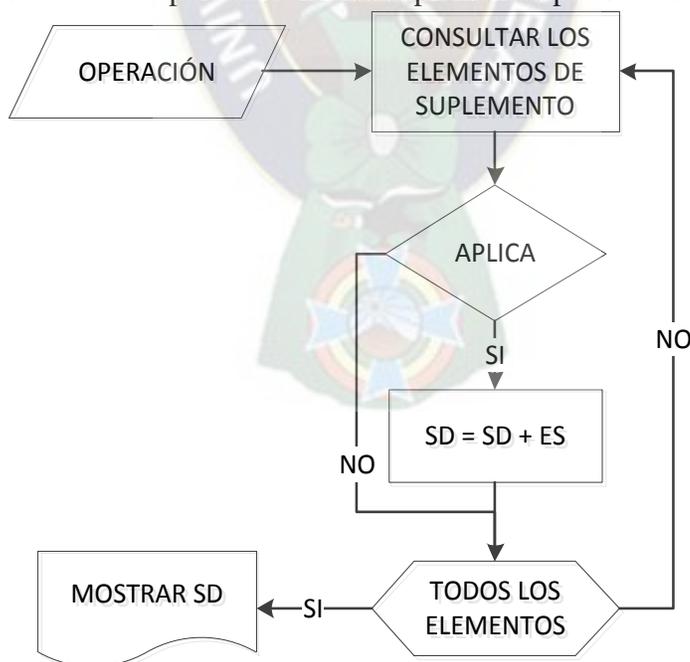
Suplementos

Los suplementos (SD) como porcentaje del tiempo normal son función de la fatiga (F), las demoras (D), las necesidades personales (NP)²³, y los suplementos adicionales (SA).

$$SD = f(F, D, SA)$$

DIAGRAMA 10-5

Procedimiento para calcular los suplementos por sección, 2014



Fuente: Elaborado con base en (Organizacion Internacional del Trabajo 1996)

²³ Las necesidades personales (NP) ya están incluidas en el estudio de tiempos improductivos, es por eso que ya nos las tomaremos en cuenta.

Los suplementos son un tema relativamente complejo ya que no se tiene precisión en el cálculo de los elementos individualmente. Entonces vamos a utilizar el procedimiento recomendado por la Oficina Internacional del Trabajo (OIT), los cuales recomiendan una tabla de elementos. Ahora lo que tenemos que hacer es verificar si ese elemento aplica para la operación. La tabla de recomendación se muestra en el cuadro siguiente.²⁴



²⁴ No vamos a realizar ninguna validación al formulario ya que este procedimiento es estándar y sugerido por la (Organización Internacional del Trabajo 1996)

CUADRO 10-16

Suplementos recomendados por la OIT

A.	Suplementos constantes:	
1.	Suplemento personal	5
2.	Suplemento por fatiga básica	4
B.	Suplementos variables:	
1.	Suplemento por estar de pie	2
2.	Suplemento por posición anormal:	
a.	un poco incomoda	0
b.	incomoda (agachado)	2
c.	muy incómoda (tendido, estirado)	7
3.	Uso de la fuerza o energía muscular (levantar, jalar o empujar)	
	Peso levantado en libras:	
5		0
10		1
15		2
20		3
25		4
30		5
35		7
40		9
45		11
50		13
60		17
70		22
4.	Mala iluminación:	
a.	un poco debajo de la recomendada	1
b.	bastante menor que la recomendada	2
c.	muy inadecuada	5
5.	Condiciones atmosféricas (calor y humedad) - variable	0-100
6.	Atención requerida:	
a.	trabajo bastante fino	0
b.	trabajo fino y preciso	2
c.	trabajo muy fino y muy preciso	5
7.	Nivel de ruido:	
a.	Continuo	0
b.	intermitente – fuerte	2
c.	intermitente - muy fuerte	5
d.	de tono alto – fuerte	5
8.	Estrés mental:	
a.	proceso bastante complejo	1
b.	atención compleja o amplia	4
c.	muy compleja	8
9.	Monotonía:	
a.	nivel bajo	0
b.	nivel medio	1
c.	nivel alto	4
10.	Tedio:	
a.	algo tedioso	0
b.	Tedioso	2
c.	muy tedioso	5

Fuente: (Organización Internacional del Trabajo 1996)

Los suplementos son esenciales en las operaciones, no vamos a omitir ningún detalle en el trabajo es por eso que vamos a tratar de calcular todos los parámetros.²⁵

CUADRO 10-17

La española: Resumen de suplementos por sección, 2014

SUPLEMENTO	MOLIDO	MEZCLADO	EMBUTIDO	COCINADO
A.				
1.	0	0	0	0
2.	4	4	4	4
B.				
1.	2	2	2	2
2.	2	2	2	2
3.	13	9	5	5
4.	2	2	2	2
5.	7	4	4	15
6.	2	2	5	2
7.	2	5	5	2
8.	1	4	4	1
9.	1	0	0	1
10.	0	0	0	0
TOTAL	36	34	33	36

Fuente: Elaborado en base al CUADRO 10-15

El suplemento por producto se muestra en el siguiente cuadro, esto según la dificultad de elaboración. Este suplemento se aplica al detalle del producto.

CUADRO 10-18

La española: Resumen de suplementos por dificultad de producto, 2014

SUPLEMENTO	SALCHICHA	CHORIZO	MORTADELA	JAMÓN	SALAME
DÍA	0%	5%	12%	12%	10%
NOCHE	28%	30%	36%	36%	30%

Fuente: Elaborado en base a gerencia de producción

²⁵ Suponer un suplemento general para todas las operaciones es un error, así como suponer que los suplementos son menores que un 20 %. Vamos a mencionar el ejemplo realizado por (Niebel y Freidvalds 2004), donde menciona un 300 % de suplemento por condiciones ambientales a temperatura húmeda de 303 K y un trabajo físico de 200 Kcal/min.

Sección de molido

Trozado

DIAGRAMA 10-6

La española: Diagrama de trozado, 2014



Fuente: Elaborado en base a observación de producción

Esta tarea compone de 5 elementos, el primero es recepción y selección de la carne, el segundo es cortado de las carnes por secciones amplias, el tercero es la separación del hueso de la carne, el cuarto es el almacenaje de desperdicios y el último es el almacenaje intermedio de carne trozada.²⁶

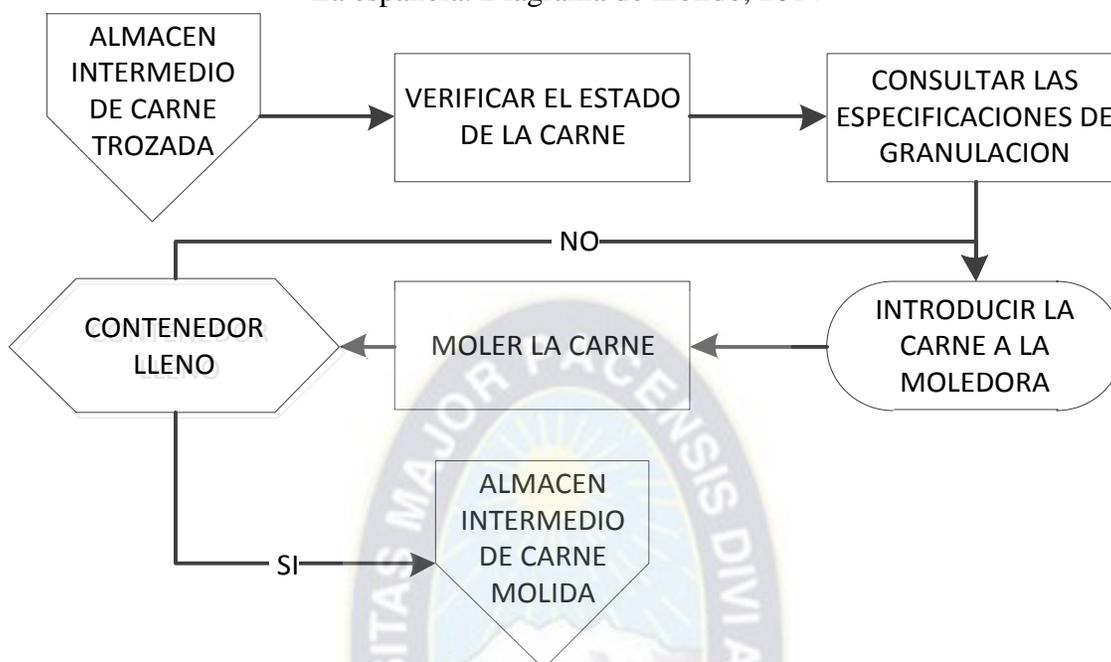
Esta tarea es llevada a cabo por un solo operario. El tiempo de suplemento calculado es 36%. Esta operación no contiene elementos extraños, por consiguiente su tiempo medido es su tiempo estándar.

²⁶ El formulario de estudio se muestra en la sección de ANEXO C.

Molido

DIAGRAMA 10-7

La española: Diagrama de molido, 2014



Fuente: Elaborado en base a observación de producción

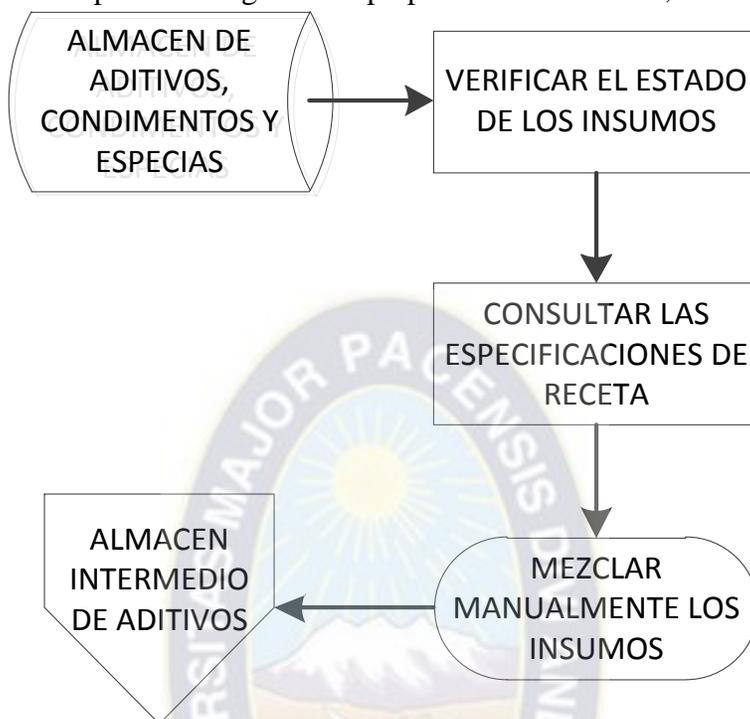
Esta tarea compone de 5 elementos; el primero es la recepción y verificación del estado de la carne del almacén intermedio de carne trozada; el segundo es la consulta de las especificaciones de granulación; el tercero es introducir la carne a la moledora; el cuarto es el molido manual, estos dos últimos componen de un ciclo, hasta llenar un contenedor de carne; y el quinto es el almacenaje intermedio de carne molida.

Esta tarea es llevada a cabo por un solo operario. El tiempo de suplemento para la operación es de 28%. La operación no tiene ningún elemento extraño.

Preparación de aditivos

DIAGRAMA 10-8

La española: Diagrama de preparación de aditivos, 2014



Fuente: Elaborado en base a observación de producción

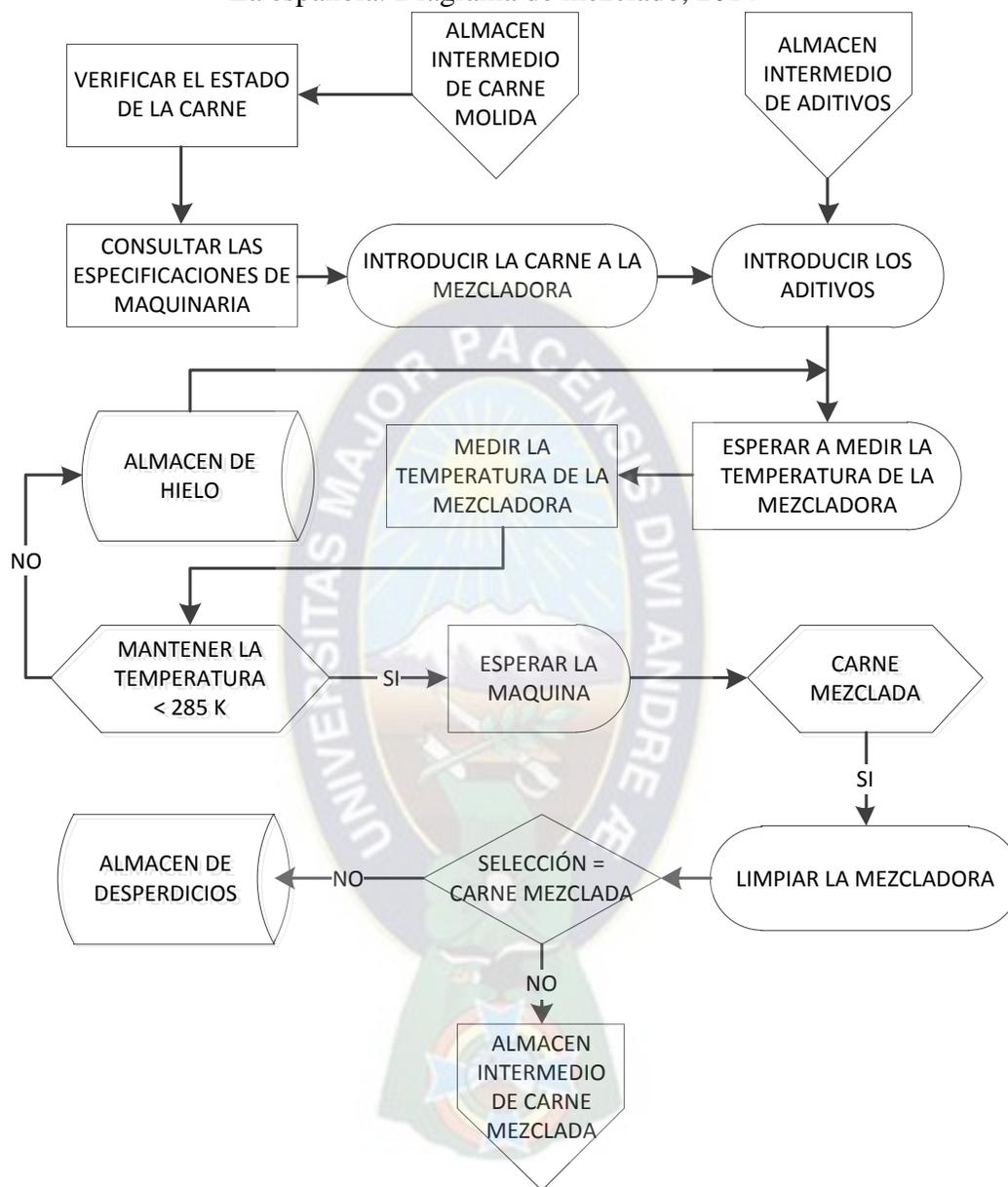
Esta tarea compone de 4 elementos; el primero es la recepción y verificación del estado de la carne; el segundo es la consulta de la receta; el tercero es la mezcla manual de los aditivos; y el cuarto es el almacenaje intermedio de aditivos.

Esta tarea es llevada a cabo por el ayudante de la sección de mezclado. El tiempo de suplemento para esta operación es de 24%. La operación no presenta ningún elemento extraño.

Mezclado

DIAGRAMA 10-9

La española: Diagrama de mezclado, 2014



Fuente: Elaborado en base a observación de producción

Esta tarea compone de 10 elementos; el primero es la recepción y verificación del estado de la carne; el segundo es la consulta de las especificaciones técnicas de la maquinaria; el tercero es la introducción de la carne a la mezcladora; el cuarto es la recepción e introducción de los aditivos en la mezcladora; el quinto es la espera para medir la temperatura de la mezcladora; el sexto es medir la temperatura de la mezcladora; el séptimo

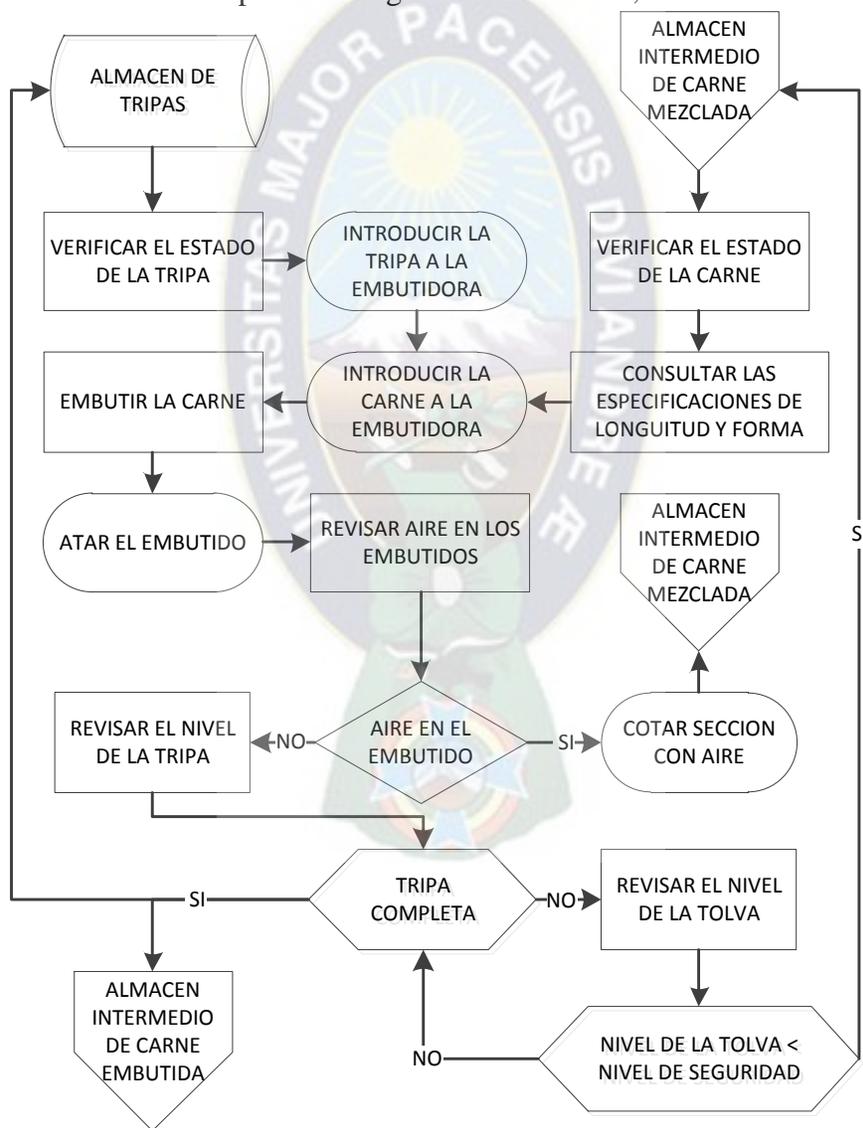
es un elemento extraño, en caso que la temperatura baje de los 285 [K] se ira al almacén de hielo y se extraerá una cantidad, luego se lo depositara en la mezcladora; el octavo es esperar a que la maquina mezcle totalmente; el noveno es extraer la limpiar la mezcladora usando el criterio de selección de desperdicio y carne bien mezclada; y el décimo es el almacenaje intermedio de carne mezclada y de desperdicios en sus respectivos lugares.

Sección de embutido

Embutido

DIAGRAMA 10-10

La española: Diagrama de embutido, 2014



Fuente: Elaborado en base a observación de producción

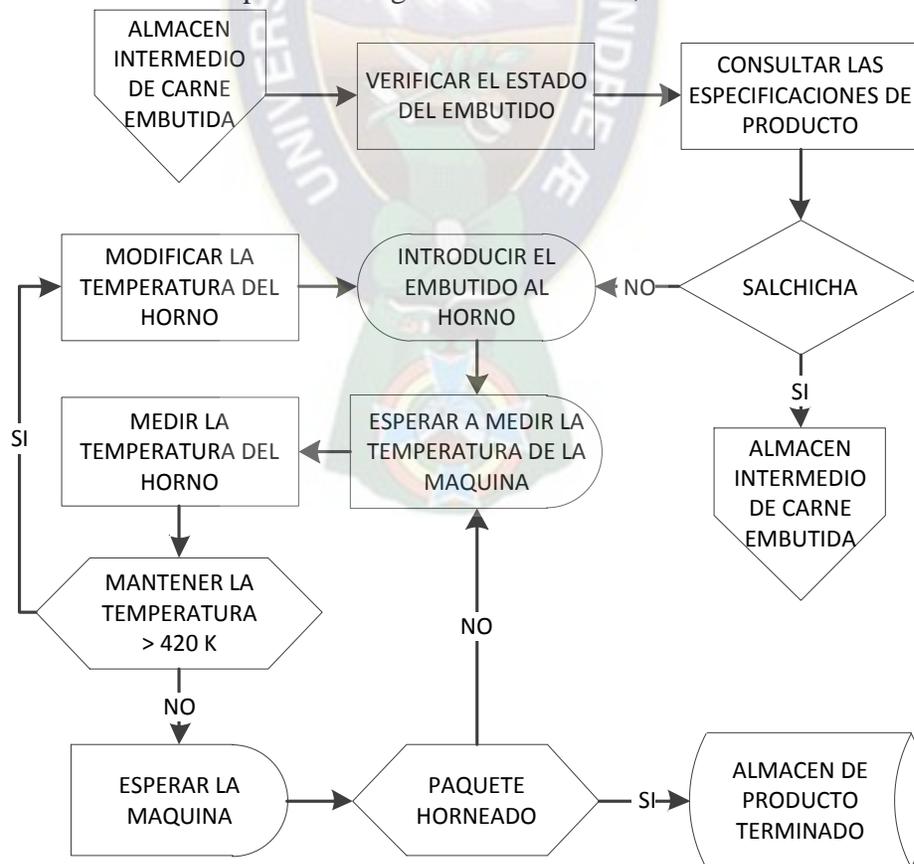
Esta tarea compone de 11 elementos; el primero es la recepción y verificación del estado de la tripa; el segundo es la introducción de la tripa en la embutidora; el tercero es la recepción y verificación del estado de la carne; el cuarto es la consulta de la especificaciones de longitud y forma; el quinto es la introducción de la carne en la embutidora; el sexto es embutir la carne; el séptimo es atar el embutido paralelamente a la anterior operación; el octavo es revisar el aire de los embutidos con un criterio selectivo de la carne que contiene aire va al almacén intermedio de carne mezclada y se atan los extremos para seguir embutiendo; el noveno es la revisión del nivel de la tripa hasta que esta concluya con un criterio de realizar la operación hasta que la tripa este llena completamente; el décimo es la revisión del nivel de la tolva hasta que el nivel sea menor al nivel de seguridad con el criterio que nunca este vacía la tolva de alimentación; y el onceavo es el almacenaje intermedio de la carne embutida.

Sección de cocinado

Ahumado

DIAGRAMA 10-11

La española: Diagrama de ahumado, 2014



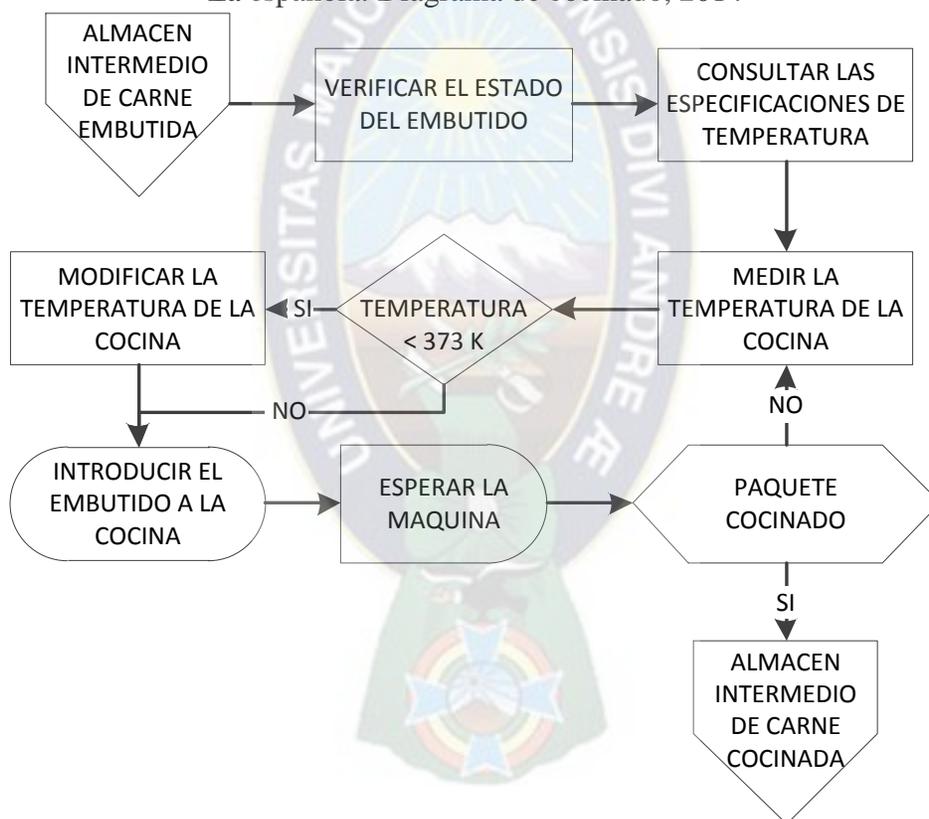
Fuente: Elaborado en base a observación de producción

Esta tarea compone de 6 elementos; el primero es la recepción y verificación del estado del embutido; el segundo es la consulta de las especificaciones de producto con el criterio selectivo de producto chorizo para ejecutar la siguiente operación; el tercero es esperar a medir la temperatura de la maquina; el cuarto es la medición de la temperatura de la maquina con el criterio de nunca dejar la temperatura del horno menor a 420 [K], modificando en su tablero la variable requerida; el quinto es la espera al producto ahumado con el criterio de volver a esperar para medir la temperatura; y el sexto es el almacenaje de producto terminado.

Cocinado

DIAGRAMA 10-12

La española: Diagrama de cocinado, 2014



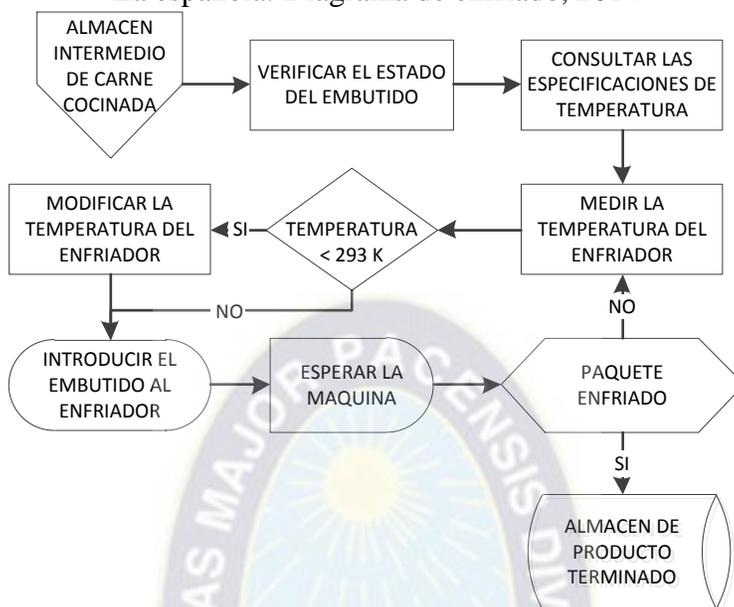
Fuente: Elaborado en base a observación de producción

Esta tarea compone de 6 elementos; el primero es la recepción y verificación del estado de la carne embutida; el segundo es la consulta de las especificaciones de temperatura; el tercero es la medición de la temperatura de la cocina con el criterio de mantener la temperatura mayor a 373 [K]; el cuarto es la introducción del embutido en la cocina; el quinto es la espera hasta que el paquete este totalmente cocinado; y el sexto es el almacenaje intermedio de carne cocinada.

Enfriado

DIAGRAMA 10-13

La española: Diagrama de enfriado, 2014



Fuente: Elaborado en base a observación de producción

Esta tarea compone de 6 elementos; el primero es la recepción y verificación del estado de la carne cocinada; el segundo es la consulta de las especificaciones de temperatura; el tercero es la medición de la temperatura del enfriador con el criterio de mantener la temperatura menor a 293 [K]; el cuarto es la introducción del embutido en el enfriador; el quinto es la espera hasta que el paquete este totalmente enfriado; y el sexto es el almacenaje de producto terminado.

Formulación de las observaciones

Para realizar el estudio de tiempos y movimientos vamos a preparar un formulario personalizado. Vamos a cruzar los requerimientos de la empresa con las sugerencias de la consultoría. El formulario será elaborado con la aplicación de (Microsoft Office Excel 2013) y (Microsoft Office Word 2013).²⁷ A continuación mostramos una impresión de pantalla de nuestro formulario.

²⁷ Estamos cambiando a la aplicación de MS Office Excel 2010 & Word 2010 para elaborar nuestros formularios, dado que existen complicaciones con la empresa al usar la aplicación más reciente de MS Office InfoPath Design & Filler 2013

También podemos mencionar que este nuevo formato no es flexible, las secciones no son distinguibles y no se puede actualizar directamente, pero pueden realizarse firmas digitales lo cual es una complicación para la empresa

FIGURA 10-1

La española: Formulario de observación de estudio de tiempos.2014



ESTUDIO DE TIEMPOS Y MOVIMIENTOS

OPERACIÓN: **MEZCLADO** SECCION: **MEZCLADO**
 FECHA: **5 DE MARZO DE 2014** N° DE OBSERVACIONES: **5**

ELEMENTO NUMERICO Y DESCRIPCION	C		TC		TN		TO		C		TC		TN		TO	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
NOTA																
1																
2																
3																
4																
5																
RESUMEN																
TO TOTAL																
CALIFICACION																
TN TOTAL																
NUM. DE OBSERV.																
TN PROMEDIO																
% SUPLEMENTOS																
TIEMPO ESTANDARE.																
NUM. OCURRENCIAS																
TIEMPO ESTANDAR																
TIEMPO ESTANDAR TOTAL DE TODOS LOS ELEMENTOS																

ELEMENTOS EXTRAÑOS			RESUMEN DE SUPLEMENTOS		
SIM	TC1	TO	DESCRIPCION	NECESIDADES PERSONALES	
A				FATIGA BASICA	
B				FATIGA VARIABLE	
C				ESPECIAL	

RESPONSABLE DE LA CONSULTORIA	SUPERVISOR A CARGO	DIRECTIVO A CARGO

Fuente: Elaborado en base a (Organizacion Internacional del Trabajo 1996)

Resumen de tiempos de tareas

El resumen del estudio de tiempos de los elementos se muestra a continuación.

CUADRO 10-19

La española: Resumen de estudio de tiempos de tareas, 2014

Operación	Tiempo calculado
Trozado	00:20:24
Molido	00:50:16
Preparación de aditivos	00:09:20
Mezclado	00:54:12
Embutido	00:40:08
Horneado	01:00:00
Cocinado	00:05:00
Enfriado	03:30:00

Fuente: Elaborado con base en el ANEXO C



10.5.2 Medición de los tiempos de elaboración de los productos

CUADRO 10-20

La española: Elaboración de Salchicha, 2014

Descripción de los elementos	Operación	Inspección	Transporte	Demora	Almacenaje	Cantidad	Tiempo [min]	Operarios
Recepción y almacén de carne		●	●			50		1
Selección de carne		●				50		1
Cortar por secciones amplias	●					50		1
Separar los huesos de la carne	●					50		1
Almacén intermedio de carne trozada			●			40		1
Moler la carne	●					40		1
Almacén intermedio de carne molida			●			40		1
Consultar especificaciones de receta		●				10		1
Mezclar manualmente los aditivos	●					10		1
Consultar especificaciones de mezclado		●				10		1
Introducir la carne a la mezcladora	●					40		2
Introducir los aditivos	●					50		1
Esperar a medir la temperatura de la mezcladora				●		50		2
Rellenar la mezcladora con hielo	●					50		1
Esperar la maquina				●		50		2
Limpiar la mezcladora	●					48		2
Almacén intermedio de carne mezclada			●			48		1
Introducir la tripa a la embutidora	●					4		2
Consultar especificaciones de longitud y forma		●				4		1
Introducir la carne a la embutidora	●					4		2
Embutir la carne	●					52		2
Atar el embutido	●					52		2
Revisar aire en los embutidos		●				50		2
Almacén intermedio de carne embutida			●			50		1
Modificar la temperatura de la cocina	●					50		1
Esperar la cocina				●		50		1
Introducir el embutido en la cocina	●					50		1
Almacén intermedio de carne cocinada			●			50		1
Modificar la temperatura del enfriador	●					50		1
Esperar la enfriadora				●		50		1
Introducir el embutido en la enfriadora	●					50		1
Almacén de producto terminado					●	50		1

Fuente: Elaborado en base al subtítulo 3.4.1 y (Niebel y Freidvalds 2004)

CUADRO 10-21

La española: Elaboración de Chorizo, 2014

Descripción de los elementos	Operación	Inspección	Transporte	Demora	Almacenaje	Cantidad	tiempo [min]	Operarios
Recepción y almacén de carne		●	●			50		1
Selección de carne		●				50		1
Cortar por secciones amplias	●					50		1
Separar los huesos de la carne	●					50		1
Almacén intermedio de carne trozada			●			40		1
Moler la carne	●					40		1
Almacén intermedio de carne molida			●			40		1
Consultar especificaciones de receta		●				10		1
Mezclar manualmente los aditivos	●					10		1
Consultar especificaciones de mezclado		●				10		1
Introducir la carne a la mezcladora	●					40		2
Introducir los aditivos	●					50		1
Esperar a medir la temperatura de la mezcladora				●		50		2
Medir la temperatura de la mezcladora	●					50		1
Rellenar la mezcladora con hielo	●					50		1
Esperar la maquina				●		50		2
Limpiar la mezcladora	●					48		2
Almacén intermedio de carne mezclada			●			48		1
Introducir la tripa a la embutidora	●					4		2
Consultar especificaciones de longitud y forma		●				4		1
Introducir la carne a la embutidora	●					48		2
Embutir la carne	●					52		2
Atar el embutido	●					52		2
Revisar aire en los embutidos		●				52		2
Revisar el nivel de la tripa		●				50		1
Revisar el nivel de la tolva		●				50		1
Almacén intermedio de carne embutida			●			50		1
Modificar la temperatura del horno	●					50		1
Esperar el ahumado del chorizo			●			50		1
Almacén de producto terminado				●		50		1

Fuente: Elaborado en base al subtítulo 3.4.1 y (Niebel y Freidvalds 2004)

CUADRO 10-22

La española: Elaboración de Mortadela, 2014

Descripción de los elementos	Op.	Ins.	Trans.	Demo.	Alm.	Cant.	Tiem.	Ope.
Recepción y almacén de carne		●	●			50		1
Selección de carne		●				50		1
Cortar por secciones amplias	●					50		1
Separar los huesos de la carne	●					50		1
Almacén intermedio de carne trozada			●			40		1
Moler la carne	●					40		1
Almacén intermedio de carne molida			●			40		1
Consultar especificaciones de receta		●				10		1
Mezclar manualmente los aditivos	●					10		1
Consultar especificaciones de mezclado		●				10		1
Introducir la carne a la mezcladora	●					40		2
Introducir los aditivos	●					50		1
Esperar a medir la temperatura de la mezcladora				●		50		2
Medir la temperatura de la mezcladora	●					50		1
Rellenar la mezcladora con hielo	●					50		1
Esperar la maquina				●		50		2
Limpiar la mezcladora	●					48		2
Almacén intermedio de carne mezclada			●			48		1
Introducir la tripa a la embutidora	●					4		2
Consultar especificaciones de longitud y forma		●				4		1
Introducir la carne a la embutidora	●					52		2
Embutir la carne	●					52		2
Atar el embutido	●					52		2
Revisar aire en los embutidos		●				50		2
Revisar el nivel de la tripa		●				50		1
Revisar el nivel de la tolva		●				50		1
Almacén intermedio de carne embutida			●			50		1
Medir la temperatura de la cocina	●					50		1
Modificar la temperatura de la cocina	●					50		1
Esperar la cocina				●		50		1
Introducir el embutido en la cocina	●					50		1
Almacén intermedio de carne cocinada			●			50		1
Modificar la temperatura del enfriador	●					50		1
Esperar la enfriadora				●		50		1
Introducir el embutido en la enfriadora	●					50		1
Almacén de producto terminado						50		1

Fuente: Elaborado en base al subtítulo 3.4.1 y (Niebel y Freidvalds 2004)

CUADRO 10-23

La española: Elaboración de Jamón, 2014

Descripción de los elementos	Oper.	Inspec.	Transp.	Demora	Alm.	Cant.	Tiempo	Ops
Recepción y almacén de carne		●	●			50		1
Selección de carne		●				50		1
Cortar por secciones amplias	●					50		1
Separar los huesos de la carne	●					50		1
Almacén intermedio de carne trozada			●			40		1
Moler la carne	●					40		1
Almacén intermedio de carne molida			●			40		1
Consultar especificaciones de receta		●				10		1
Mezclar manualmente los aditivos	●					10		1
Consultar especificaciones de mezclado		●				10		1
Introducir la carne a la mezcladora	●					40		2
Introducir los aditivos	●					50		1
Esperar a medir la temperatura de la mezcladora				●		50		2
Medir la temperatura de la mezcladora	●					50		1
Rellenar la mezcladora con hielo	●					50		1
Esperar la maquina				●		50		2
Limpiar la mezcladora	●					48		2
Almacén intermedio de carne mezclada			●			48		1
Introducir la tripa a la embutidora	●					4		2
Consultar especificaciones de longitud y forma		●				4		1
Introducir la carne a la embutidora	●					52		2
Embutir la carne	●					52		2
Atar el embutido	●					52		2
Revisar aire en los embutidos		●				50		2
Revisar el nivel de la tripa		●				50		1
Revisar el nivel de la tolva		●				50		1
Almacén intermedio de carne embutida			●			50		1
Medir la temperatura de la cocina	●					50		1
Modificar la temperatura de la cocina	●					50		1
Esperar la cocina				●		50		1
Introducir el embutido en la cocina	●					50		1
Almacén intermedio de carne cocinada			●			50		1
Modificar la temperatura del enfriador	●					50		1
Esperar la enfriadora				●		50		1
Introducir el embutido en la enfriadora	●					50		1
Almacén intermedio de embutido enfriado			●			50		1
Modificar la temperatura del horno	●					50		1
Esperar el ahumado del jamón				●		50		1
Almacén de producto terminado				●		50		1

Fuente: Elaborado en base al subtítulo 3.4.1 y (Niebel y Freidvalds 2004)

CUADRO 10-24

La española: Elaboración de Salame, 2014

Descripción de los elementos	Operación	Inspección	Transporte	Demora	Almacenaje	Cantidad	Tiempo [min]	Operarios
Recepción y almacén de carne		●	●			50		1
Selección de carne		●				50		1
Cortar por secciones amplias	●					50		1
Separar los huesos de la carne	●					50		1
Almacén intermedio de carne trozada			●			40		1
Moler la carne	●					40		1
Almacén intermedio de carne molida			●			40		1
Consultar especificaciones de receta		●				10		1
Mezclar manualmente los aditivos	●					10		1
Consultar especificaciones de mezclado		●				10		1
Introducir la carne a la mezcladora	●					40		2
Introducir los aditivos	●					50		1
Esperar a medir la temperatura de la mezcladora				●		50		2
Medir la temperatura de la mezcladora	●					50		1
Rellenar la mezcladora con hielo	●					50		1
Esperar la maquina				●		50		2
Limpiar la mezcladora	●					48		2
Almacén intermedio de carne mezclada			●			48		1
Introducir la tripa a la embutidora	●					4		2
Consultar especificaciones de longitud y forma		●				4		1
Introducir la carne a la embutidora	●					52		2
Embutir la carne	●					52		2
Atar el embutido	●					52		2
Revisar aire en los embutidos		●				50		2
Revisar el nivel de la tripa		●				50		1
Revisar el nivel de la tolva		●				50		1
Almacén intermedio de carne embutida			●			50		1
Modificar la temperatura del horno	●					50		1
Esperar el ahumado del salame			●			50		1
Almacén de producto terminado				●		50		1

Fuente: Elaborado en base al subtítulo 3.4.1 y (Niebel y Freidvalds 2004)

Las secciones de: Molido y Cocinado solo tienen un operador por lo que el tiempo calculado anteriormente es el tiempo estándar. Las secciones de: Mezclado y Embutido tienen dos operadores por lo que vamos a elaborar un diagrama hombre-máquina para estimar el tiempo estándar.

Las secciones de Molido y Mezclado son estándar para todas las operaciones por lo que vamos a tomarlos como un subproducto denominado “carne molida” y luego los vamos a completar para las operaciones de producto terminado.

DIAGRAMA 10-14

La española: Diagrama hombre-máquina para Carne molida, 2014

Descripcion del elemento	Operador 1	Operador 2	Operador 3	Mezclador	Moledora	Escala	Tiempo
Recepcion y almacen de carne	█						7
Selección de carne	█						15
Cortar por secciones amplias	█						2
Separar los huesos de la carne	█						2,4
Almacen intermedio de carne trozada	█						5
Moler la carne					█		25
Almacen intermedio de carne molida	█						3
Consultar especificaciones de receta		█					1
Mezclar manualmente los aditivos		█					3
Consultar especificaciones de mezclado			█				2
Introducir la carne a la mezcladora		█	█				20
Introducir los aditivos		█					3
Esperar a medir la temperatura de la mezcladora			█	█			2
Medir la temperatura de la mezcladora			█	█			1
Rellenar la mezcladora con hielo			█	█			3
Esperar la maquina			█	█			24
Limpiar la mezcladora		█	█				5
Almacen intermedio de carne mezclada		█					120
Tiempo de ciclo activo	107	107	107	107	107	minutos	
Accion	96,4	76,4	56,4	30	25	minutos	
Ocio	11	31	51	77	82	minutos	
Suplementos (Molido 36%, Mezclado 34%)	34,7	26	19,2			minutos	
Limpieza y mantenimiento	10	10	10	10	10	minutos	
Tiempo activo total	152	143	137	117	117	minutos	

Fuente: Elaborado en base a los tiempos de operaciones CUADRO 10-19

DIAGRAMA 10-15

La española: Diagrama Hombre-máquina para Salchicha, 2014

Descripcion del elemento	Operador 1	Operador 2	Operador 3	Embutidora	Cocina	Escala	Tiempo
Almacen intermedio de carne mezclada	█						2
Introducir la tripa a la embutidora	█						20
Consultar especificaciones de longitud y forma	█						7
Introducir la carne a la embutidora		█					10
Embutir la carne		█		█			2
Atar el embutido	█	█		█			15
Revisar aire en los embutidos	█	█		█			7
Revisar el nivel de la tripa	█	█		█			38
Revisar el nivel de la tolva	█	█		█			40
Almacen intermedio de carne embutida		█					2
Medir la temperatura de la cocina			█				20
Modificar la temperatura de la cocina			█				4
Esperar la cocina					█		45
Introducir el embutido en la cocina					█		8
Almacen intermedio de carne cocinada							2
Modificar la temperatura del enfriador			█				5
Esperar la enfriadora					█		120
Introducir el embutido en la enfriadora			█				4
Almacen de producto terminado			█				120
Tiempo de ciclo activo	192	192	192	192	192	minutos	
Accion	85	112	86	17	45	minutos	
Ocio	107	80	106	175	147	minutos	
Suplementos (Embutido 33%, Cocinado 36%)	28,1	37	31			minutos	
Limpieza y mantenimiento	10	10	10	10	10	minutos	
Tiempo activo total	230	239	233	202	202	minutos	

Fuente: Elaborado en base a los tiempos de operaciones CUADRO 10-20

DIAGRAMA 10-16

La española: Diagrama hombre-máquina para Chorizo, 2014

Descripcion del elemento	Operador 1	Operador 2	Operador 3	Embutidora	Cocina	Escala	Tiempo
Almacen intermedio de carne mezclada	█						2
Introducir la tripa a la embutidora	█						25
Consultar especificaciones de longitud y forma	█						1
Introducir la carne a la embutidora		█					5
Embutir la carne		█		█			2
Atar el embutido	█	█					15
Revisar aire en los embutidos		█					5
Revisar el nivel de la tripa	█	█					30
Revisar el nivel de la tolva	█	█					5
Almacen intermedio de carne embutida		█					2
Modificar la temperatura del horno			█		█		120
Esperar al horneado del chorizo			█		█		4
Almacen de producto terminado			█		█		120
Tiempo de ciclo activo	79	79	79	79	79	minutos	
Accion	15	19	66	6	35	minutos	
Ocio	64	60	13	73	44	minutos	
Suplementos (Embutido 33%, Cocinado 36%)	4,95	6,27	23,8			minutos	
Limpieza y mantenimiento	10	10	10	10	10	minutos	
Tiempo activo total	94	95	113	89	89	minutos	

Fuente: Elaborado en base a los tiempos de operaciones CUADRO 10-21

DIAGRAMA 10-17

La española: Diagrama hombre-máquina para Mortadela, 2014

Descripcion del elemento	Operador 1	Operador 2	Operador 3	Embutidora	Cocina	Escala	Tiempo
Almacen intermedio de carne mezclada	█						2
Introducir la tripa a la embutidora	█						15
Consultar especificaciones de longitud y forma	█						8
Introducir la carne a la embutidora		█					5
Embutir la carne		█		█			2
Atar el embutido	█	█		█			15
Revisar aire en los embutidos		█		█			10
Revisar el nivel de la tripa		█		█			80
Revisar el nivel de la tolva	█	█		█			15
Almacen intermedio de carne embutida		█					2
Medir la temperatura de la cocina			█				20
Modificar la temperatura de la cocina			█				4
Esperar la cocina			█		█		180
Introducir el embutido en la cocina			█				8
Almacen intermedio de carne cocinada			█				2
Modificar la temperatura del enfriador			█				5
Esperar la enfriadora			█		█		180
Introducir el embutido en la enfriadora			█				4
Almacen de producto terminado			█				240
Tiempo de ciclo activo	328	328	328	328	328	minutos	
Accion	45	42	122	13	90	minutos	
Ocio	283	286	206	315	238	minutos	
Suplementos (Embutido 33%, Cocinado 36%)	14,9	13,9	43,9			minutos	
Limpieza y mantenimiento	10	10	10	10	10	minutos	
Tiempo activo total	353	352	382	338	338	minutos	

Fuente: Elaborado en base a los tiempos de operaciones CUADRO 10-22

DIAGRAMA 10-18

La española: Diagrama hombre-máquina para Jamón, 2014

Descripcion del elemento	Operador 1	Operador 2	Operador 3	Embutidora	Cocina	Escala	Tiempo
Almacen intermedio de carne mezclada	█						2
Introducir la tripa a la embutidora	█						15
Consultar especificaciones de longitud y forma	█						8
Introducir la carne a la embutidora		█					5
Embutir la carne		█		█			2
Atar el embutido	█	█		█			15
Revisar aire en los embutidos		█		█			10
Revisar el nivel de la tripa		█		█			80
Revisar el nivel de la tolva	█	█		█			15
Almacen intermedio de carne embutida		█					2
Medir la temperatura de la cocina			█				20
Modificar la temperatura de la cocina			█		█		4
Esperar la cocina			█		█		180
Introducir el embutido en la cocina			█		█		8
Almacen intermedio de carne cocinada			█				2
Modificar la temperatura del enfriador			█		█		5
Esperar la enfriadora			█		█		180
Introducir el embutido en la enfriadora			█		█		4
Almacen de producto terminado			█				240
Tiempo de ciclo activo	410	410	410	410	410	minutos	
Accion	74	104	116	25	90	minutos	
Ocio	336	306	294	385	320	minutos	
Suplementos (Embutido 33%, Cocinado 36%)	24,4	34,3	41,8			minutos	
Limpieza y mantenimiento	10	10	10	10	10	minutos	
Tiempo activo total	444	454	462	420	420	minutos	

Fuente: Elaborado en base a los tiempos de operaciones CUADRO 10-23

DIAGRAMA 10-19

La española: Diagrama hombre-máquina para Salame, 2014

Descripcion del elemento	Operador 1	Operador 2	Operador 3	Embutidora	Cocina	Escala	Tiempo
Almacen intermedio de carne mezclada	█						2
Introducir la tripa a la embutidora	█						25
Consultar especificaciones de longitud y forma	█						1
Introducir la carne a la embutidora		█					5
Embutir la carne		█		█			2
Atar el embutido	█	█					15
Revisar aire en los embutidos	█	█					5
Revisar el nivel de la tripa	█	█					30
Revisar el nivel de la tolva	█	█					5
Almacen intermedio de carne embutida		█					2
Modificar la temperatura del horno			█		█		120
Esperar al horneado del chorizo			█				4
Almacen de producto terminado			█		█		120
Tiempo de ciclo activo	88	88	88	88	88	minutos	
Accion	15	19	45	6	20	minutos	
Ocio	73	69	43	82	68	minutos	
Suplementos (Embutido 33%, Cocinado 36%)	4,95	6,27	16,2			minutos	
Limpieza y mantenimiento	10	10	10	10	10	minutos	
Tiempo activo total	103	104	114	98	98	minutos	

Fuente: Elaborado en base a los tiempos de operaciones CUADRO 10-24

Resumen de tiempos de elaboración

CUADRO 10-25

La española: Tiempos de elaboración por producto [Hr], 2014

Producto	Tiempo de elaboración	
	Diurno	Nocturno
Salchicha	8,82	11,30
Chorizo	5,58	6,90
Mortadela	9,88	12,00
Jamón	9,88	12,00
Salame	5,84	6,90

Fuente: Elaborado en base a los diagramas hombre-máquina según producto

10.5.3 Medición de los rendimientos

Para medir los rendimientos vamos a distinguir entre los horarios diurno y nocturno. Los suplementos según dificultad están en el CUADRO 10-24.²⁸

Entonces dados los tiempos de elaboración, suplementos, tiempos de elaboración por sección, vamos a armar nuestra matriz de tiempos.

CUADRO 10-26

La española: Composición de los tiempos de elaboración [Hr], 2014

Producto	Salchicha		Chorizo		Mortadela		Jamón		Salame	
	0%	28%	5%	30%	12%	36%	12%	36%	10%	30%
Operación	Día	Noche	Día	Noche	Día	Noche	Día	Noche	Día	Noche
Trozado	00:27,6	00:35,7	00:29,9	00:36,0	00:31,4	00:37,9	00:31,4	00:37,9	00:30,1	00:36,0
Molido	01:08,8	01:27,3	01:11,8	01:28,3	01:16,0	01:32,4	01:16,0	01:32,4	01:15,9	01:28,3
Prep. de ad.	00:12,6	00:16,8	00:13,7	00:16,1	00:14,0	00:17,8	00:14,0	00:17,8	00:13,8	00:16,1
Mezclado	01:13,7	01:34,1	01:17,9	01:35,5	01:22,4	01:40,9	01:22,4	01:40,9	01:21,0	01:35,5
Embutido	00:54,9	01:09,8	00:57,6	01:10,3	01:01,9	01:14,8	01:01,9	01:14,8	01:00,4	01:10,3
Horneado			01:25,8	01:46,8					01:29,6	01:46,8
Cocinado	00:06,0	00:08,2			00:07,0	00:09,9	00:07,0	00:09,9		
Enfriado	04:45,0	06:05,1			05:19,3	06:28,0	05:19,3	06:28,0		
Total	08:49,6	11:17,1	05:34,7	06:54,1	09:53,0	12:00,7	09:53,0	12:00,7	05:50,8	06:54,1

Fuente: Elaborado en base a los tiempos de elaboración y el rendimiento anterior mencionado

Los rendimientos por kilogramo están en la siguiente tabla y con esto finalizamos es estudio de tiempos y movimientos.

²⁸ Esta es una estimación de Gerencia de producción.

CUADRO 10-27

La española: Rendimiento de los productos según horario [Hr/Kg], 2014

Producto	Salchicha		Chorizo		Mortadela		Jamón		Salame	
	0%	28%	5%	30%	12%	36%	12%	36%	10%	30%
	Día	Noche	Día	Noche	Día	Noche	Día	Noche	Día	Noche
Trozado	0,0092	0,0118	0,0097	0,0120	0,0104	0,0126	0,0104	0,0126	0,0102	0,0120
Molido	0,0228	0,0292	0,0239	0,0296	0,0255	0,0310	0,0255	0,0310	0,0251	0,0296
Prep. de ad.	0,0042	0,0054	0,0044	0,0055	0,0047	0,0058	0,0047	0,0058	0,0047	0,0055
Mezclado	0,0246	0,0315	0,0258	0,0319	0,0275	0,0334	0,0275	0,0334	0,0270	0,0319
Embutido	0,0182	0,0233	0,0191	0,0237	0,0204	0,0247	0,0204	0,0247	0,0200	0,0237
Horneado			0,0286	0,0354					0,0299	0,0354
Cocinado	0,0023	0,0029			0,0025	0,0031	0,0025	0,0031		
Enfriado	0,0952	0,1219			0,1066	0,1295	0,1066	0,1295		
Total	08:49,6	11:17,1	05:34,7	06:54,1	09:53,0	12:00,7	09:53,0	12:00,7	05:50,8	06:54,1

Fuente: Elaborado en base al CUADRO 10-25

10.6 Pronóstico de la demanda

La metodología utilizada para hacer el pronóstico de la demanda es el siguiente:

10.6.1 Bases para el pronóstico de la demanda

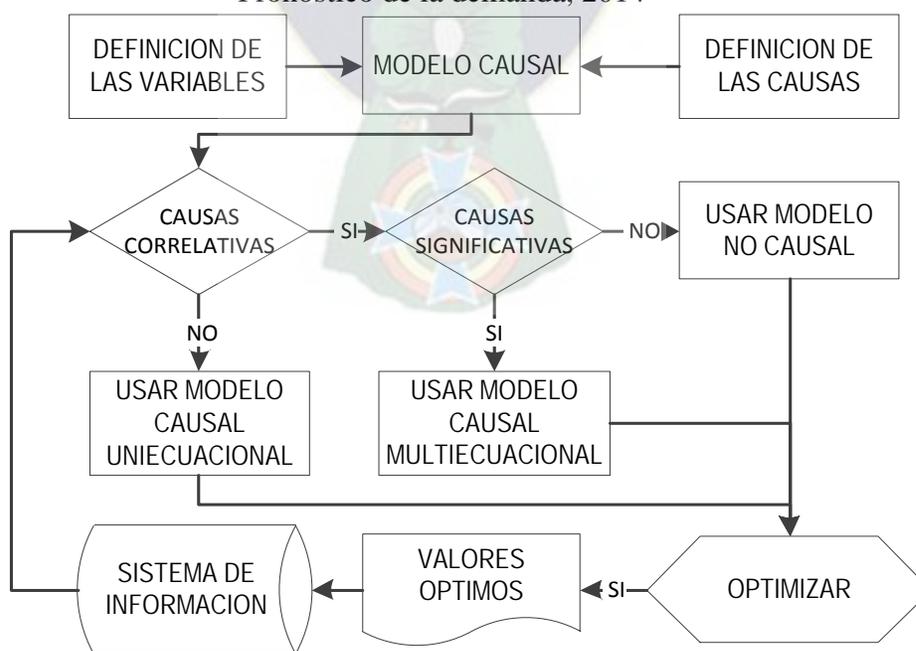
Heurística.- Es el procesos de mejora continua.

Simulación.- Es un proceso inductivo de cálculo que se basa en la realidad.

10.6.2 Fases del pronóstico de la demanda

DIAGRAMA 10-20

Pronostico de la demanda, 2014



Fuente: Elaborado en base a (Kinneary y Taylor 1998), (Gujarati y Porter 2010) y (Wayne L. 2003)

10.6.3 Definición de la variable endógena

La variable a pronosticar es la demanda por producto.

10.6.4 Medición de los valores históricos

Los valores de demanda histórica están sistematizados en la empresa, entonces solo vamos a compilarlas.

CUADRO 10-28

La española, Demanda histórica [Kg], 2012 - 2013

PERIODO	SALCHICHA	CHORIZO	MORTADELA	JAMÓN	SALAME	TOTAL
ene-12	13.384	7.154	1.363	596	390	19.108
feb-12	14.615	8.535	1.632	654	428	21.298
mar-12	20.816	10.114	1.772	664	435	23.196
abr-12	15.244	8.247	1.609	640	419	21.667
may-12	14.102	7.765	1.443	626	410	19.537
jun-12	25.094	14.948	1.578	687	418	19.512
jul-12	5.368	5.564	928	585	383	13.895
ago-12	14.080	10.699	1.384	658	431	17.788
sep-12	11.939	7.580	1.321	635	416	17.591
oct-12	17.059	10.045	2.006	689	485	26.619
nov-12	24.416	8.228	2.488	726	475	29.986
dic-12	22.482	7.400	2.309	715	468	30.582
ene-13	17.238	5.527	1.355	598	392	19.681
feb-13	15.052	8.876	1.677	677	443	21.936
mar-13	21.440	10.519	1.821	686	449	23.891
abr-13	15.701	8.577	1.652	657	430	22.317
may-13	14.525	8.076	1.481	642	420	20.123
jun-13	25.847	15.546	1.506	653	427	20.097
jul-13	5.529	5.787	1.020	590	386	14.312
ago-13	14.503	11.127	1.423	666	436	18.322
sep-13	12.297	7.883	1.356	640	419	18.119
oct-13	17.571	10.447	2.294	773	490	27.418
nov-13	25.148	8.557	2.305	732	479	30.886
dic-13	23.157	7.696	2.298	696	471	31.500

Fuente: Elaborado con base en los datos de Gerencia General de la empresa

10.6.5 Definición de las causas

Para este cometido vamos a explorar una serie de variables que consideramos causan significativamente a la demanda de productos de la española.²⁹

Las variables seleccionadas³⁰ son: Ingreso per cápita (si un habitante tiene mayor ingreso tendrá mayor poder de compra, entonces podrá comprar más productos embutidos), precio (si el precio del producto es alto el cliente preferirá otro producto esto en proporción a la elasticidad precio-demanda), Publicidad (si los cliente potenciales están informados es posible que compren los productos), Fuerza de ventas (esta variable afecta primero a la variable publicidad luego a la demanda, si existe mayor fuerza de ventas se tendrá una mayor demanda), Inflación (esta variable afecta primero al precio luego a la demanda, si existe mayor inflación el poder de compra disminuirá).

10.6.6 Modelo causal

Con la definición clara de las variables exógenas, el modelo está claro:

La demanda (D_t) es igual a una constante (β_0), más el coeficiente (β_1) multiplicado por el ingreso per cápita rezagado (PPC_{t-1}), más el coeficiente (β_2) multiplicado por el precio rezagado del producto (PRE_{t-1}), más el coeficiente (β_3) multiplicado por el presupuesto en publicidad (PUB), más el error de pronóstico (μ_t).

El precio de mercado (PRE) es igual a una constante (γ_0), más el coeficiente (γ_1) multiplicado por la inflación (π_t), más el error de pronóstico (ε_t).

El presupuesto de publicidad (PUB_t) es igual a una constante (δ_0), más el coeficiente (δ_1) multiplicado por la Fuerza de Ventas (FDV_t), más el error de pronóstico (ω_t).

$$D_t = \beta_0 + \beta_1 PPC_{t-1} + \beta_2 PRE_{t-1} + \beta_3 PUB_t + \mu_t \quad (10-1)$$

$$PRE_t = \gamma_0 + \gamma_1 \pi_t + \varepsilon_t \quad (10-2)$$

$$PUB_t = \delta_0 + \delta_1 FDV_t + \omega_t \quad (10-3)$$

El problema de la Identificación

Para entender las condiciones de orden y de rango, se introduce la siguiente notación:

M = número de variables endógenas en el modelo.

m = número de variables endógenas en una ecuación dada.

K = número de variables predeterminadas en el modelo, incluyendo el intercepto.

²⁹ Dado que estamos haciendo el pronóstico de la demanda no podemos usar variables causales que estén en el mismo periodo de la variable a pronosticar. Entonces vamos a hacer el estudio con un rezago de 1 y 2 periodos.

³⁰ Esto se elaboró en conjunto con la Gerencia General de la Empresa.

= número de variables predeterminadas en una ecuación dada.

La condición de identificación es:

$$K - k \geq m - 1$$

Para la ecuación (10-1), la condición de identificación es:

$$2 - 2 \geq 3 - 1$$

$$0 \geq 2$$

La ecuación (10-1) esta subidentificada, por lo tanto no existe algún método para resolver esta ecuación.

Para la ecuación (10-2), la condición de identificación es:

$$2 - 0 \geq 1 - 1$$

$$2 \geq 0$$

La ecuación (10-2) esta sobreidentificada, por lo tanto se puede resolver el modelo por el Método de Mínimos Cuadrados en dos Etapas (MC2E).

Para la ecuación (10-3), la condición de identificación es:

$$2 - 0 \geq 1 - 1$$

$$2 \geq 0$$

La ecuación (10-3) esta sobreidentificada, por lo tanto se puede resolver el modelo por el Método de Mínimos Cuadrados en dos Etapas (MC2E).

Considerando la ecuación (10-1) que es la ecuación principal, no puede ser resuelta, entonces podemos asegurar que no existe causalidad correlativa que puede ser resuelta. Por lo tanto vamos a resolver la ecuación (10-1) como modelo uniecuacional.

Modelo causal uniecuacional

Ahora vamos a determinar mediante una prueba de hipótesis si la ecuación (10-1) para todos los productos es significativa

- i. Especificación del modelo econométrico de la teoría

$$D_t = \beta_0 + \beta_1 PPC_{t-1} + \beta_2 PRE_{t-1} + \beta_3 PUB_t + \mu_t$$

- ii. Planteamiento de la hipótesis

$$H_0: \beta_0 = \beta_1 = \beta_2 = \beta_3 = \beta_4 = \beta_5 = \beta_6 = 0$$

$$H_a: \beta_0 \neq \beta_1 \neq \beta_2 \neq \beta_3 \neq \beta_4 \neq \beta_5 \neq \beta_6 \neq 0$$

iii. Estimación de los parámetros

CUADRO 10-29

La española, Estimación de Parámetros 2012 - 2013

β_t	-0,34	1091,90	4,90	3172,72
σ_{β_t}	1,63	10493,71	8,47	227339,86
$R^2 = 0,06290$		$\sigma_\varepsilon = 11997,77$		
$F = 0,44749$		$n - k = 20$		
$SCR = 193242722$		$SCE = 2878927861$		

Fuente: Elaboración propia

iv. Estadístico de prueba

$$F = \frac{SCR}{SCE} = 0,44749$$

v. Estadístico de prueba

$$F_{\alpha, n-k, n-1} = F_{0,05, 20, 23} = 2,05$$

vi. Contraste

$$F_{\text{calculado}} = 0,45 \quad F_{\alpha, n-k, n-1} = 2,05$$

vii. Conclusión

Para un nivel de significancia del 5%, podemos aceptar que todos los parámetros de estimación son iguales a cero por lo tanto el modelo no es significativo en lo global.

Conclusión final

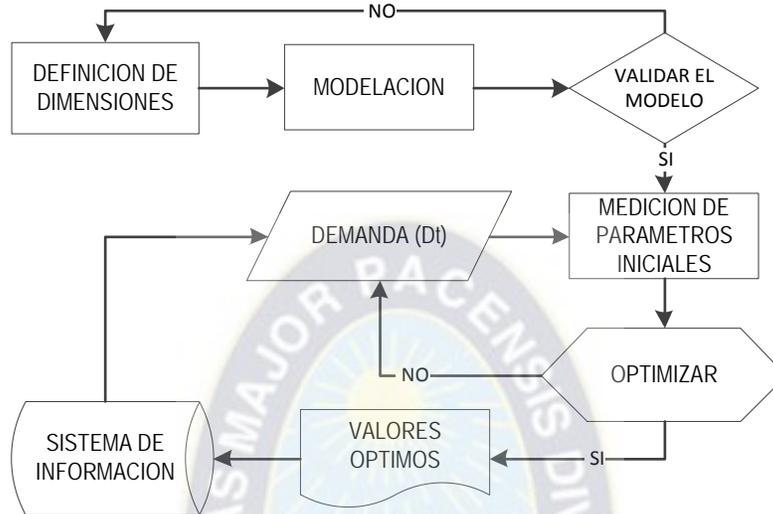
Con los datos anteriores podemos determinar que no existe causalidad significativa para la demanda de productos embutidos de la española o al menos no podemos determinar la combinación lineal de estas variables. Entonces vamos a seguir un proceso de no causalidad para determinar la demanda de productos embutidos de “La española”.

10.6.7 Modelo no causal

Los procesos no causales más completos son: el proceso de Holt-Winters aditivo, y el promedio móvil en dos fases. Considerando la experiencia de los autores vamos a

determinar que el promedio móvil en dos fases requieren datos de tres ciclos o más. Entonces el método a utilizar es el método de Holt-Winters aditivo.³¹

DIAGRAMA 10-21
Método de Holt-Winters, 2014



Fuente: Elaborado en base a (Wayne L. 2003)

Modelo de Holt-Winters

La demanda del producto (Y_t) es igual a el parámetro de nivel inicial (A_t) más el periodo estacional (s) multiplicado por el parámetro de tendencia inicial (T_t), más el parámetros de estacionalidad iniciales (R_{t-s+1});

$$Y_{t+1} = (A_t + s * T_t) + R_{t-s+1}$$

El parámetro de nivel (A_t) es igual al coeficiente de nivel (α) multiplicado por la demanda del producto (Y_t) entre el parámetro de estacionalidad de estación pasada (R_{t-s}) más el complemento del coeficiente de nivel ($1 - \alpha$), multiplicado por el parámetro de nivel anterior (A_{t-1}) más el parámetro de tendencia anterior (T_{t-1})

$$A_t = \alpha \frac{Y_t}{R_{t-s}} + (1 - \alpha)(A_{t-1} + T_{t-1})$$

El parámetro de tendencia (T_t) es igual al coeficiente de tendencia (β) multiplicado por la diferencia del parámetro de nivel (A_t) y el parámetro de nivel anterior (A_{t-1}) más el complemento del coeficiente de tendencia ($1 - \beta$) multiplicado por el parámetro de tendencia anterior (T_{t-1})

$$T_t = \beta(A_t - A_{t-1}) + (1 - \beta)T_{t-1}$$

³¹ Los conceptos de nivel, tendencia y estacionalidad se determinan en base a los resultados del modelo, entonces no vamos a tocar ninguno de estos conceptos hasta conseguir el modelo final.

El parámetro de estacionalidad (R_t) es igual al coeficiente de estacionalidad (γ) multiplicado por la demanda (Y_t) entre el parámetro de nivel (A_t) más el complemento del coeficiente de estacionalidad ($1 - \gamma$) multiplicado por el parámetro de estacionalidad de la anterior estación (R_{t-s})

$$R_t = \gamma \frac{Y_t}{A_t} + (1 - \gamma)R_{t-s}$$

El modelo de Holt-Winters ya está bien definido, entonces solo nos falta calcular los parámetros iniciales y optimizarlos para obtener los valores óptimos.

Los parámetros iniciales son: Cantidad de periodos en un año (L) son 12; Cantidad de periodos en un mes (K); Cantidad de periodos en la serie (r); Parámetro de nivel inicial (A_0); Parámetro de tendencia inicial (T_0); Parámetros de estacionalidad iniciales (R_{i0}); Coeficiente de nivel (α); Coeficiente de tendencia (β); Coeficiente de estacionalidad (γ); Promedio del año ($Prom_{rt}$)

Calculo de los parámetros iniciales

La cantidad de periodos en un año son 12, la cantidad de periodos en un mes es 1, la cantidad de periodos en la serie son 2. El coeficiente de nivel es 0, el coeficiente de tendencia es 0, el coeficiente de estacionalidad es 0.

El Parámetro de tendencia inicial (T_0) es igual a el promedio del segundo año ($Prom_{r2}$) menos el promedio del primer año ($Prom_{r1}$) dividido entre la Cantidad de periodos en un año (L) multiplicado por la cantidad de periodos de la serie (r) menos 1.

$$T_0 = \frac{Prom_{r2} - Prom_{r1}}{L * (r - 1)}$$

Para el pronóstico de la salchicha el valor del Parámetro de tendencia inicial (T_0) es:

$$T_0 = \frac{17334,33 - 16550,25}{12 * (2 - 1)}$$

$$T_0 = 65,34028$$

El parámetro de nivel inicial (A_0) es igual a el Promedio del primer año ($Prom_{r1}$) menos la Cantidad de periodos en una año (L) entre 2 por el parámetro de tendencia inicial (T_0).

$$A_0 = Prom_{r1} - L/2 * T_0$$

Para el pronóstico de la salchicha el valor del parámetro de tendencia inicial (A_0) es:

$$A_0 = 16550,25 - 12/2 * 65,34$$

$$A_0 = 8759,62$$

Los parámetros de estacionalidad iniciales (R_{i0}) es igual a la demanda (D_t) dividido entre el promedio del primer año ($Prom_{r1}$) menos la Cantidad de periodos en un año (L) más 1 entre 2 menos el periodo del año (i) los dos términos anteriores multiplicados por el Parámetro de tendencia inicial (T_0).

$$R_{i0} = \frac{D_t}{Prom_{rj} - \left(\frac{L+1}{2} - i\right) * T_0}$$

Entonces tenemos dos valores del parámetro de estacionalidad inicial para cada mes, lo que vamos a hacer es promediarlos y ajustarlos para evitar el problema del sesgo.

Para el pronóstico de la salchicha el valor del parámetro de estacionalidad inicial del primer mes del primer año (R_{10}) es:

$$R_{10} = \frac{7154}{16550,25 - \left(\frac{12+1}{2} - 1\right) * 65,34}$$

$$R_{10} = 0,81595$$

Para el pronóstico de la salchicha el valor del parámetro de estacionalidad inicial del primer mes del segundo año (R_{130}) es:

$$R_{130} = \frac{D_t}{Prom_{r2} - \left(\frac{L+1}{2} - i\right) * T_0}$$

$$R_{130} = \frac{5528}{17334,33 - \left(\frac{12+1}{2} - 1\right) * 65,34}$$

$$R_{130} = 0,61678$$

El promedio del Parámetro de estacionalidad inicial ($\overline{R_{i0}}$) es igual a el promedio entre el Parámetro de estacionalidad inicial del primer mes del primer año (R_{10}) y el Parámetro de estacionalidad inicial del primer mes del segundo año (R_{20}).

$$\overline{R_{i0}} = \frac{R_{10} + R_{20}}{2}$$

El valor del promedio del Parámetro de estacionalidad inicial ($\overline{R_{10}}$) es:

$$\overline{R_{10}} = \frac{0,81595 + 0,61678}{2}$$

$$\overline{R_{10}} = 0,71636$$

El valor ajustado del Parámetro de estacionalidad (\widehat{R}_{10}) es igual al promedio del Parámetro de estacionalidad inicial (\overline{R}_{10}), multiplicado por la Cantidad de periodos en un año (L) entre la sumatoria del promedio del Parámetro de estacionalidad inicial ($\sum \overline{R}_{10}$).

$$\widehat{R}_{10} = \overline{R}_{10} * \frac{L}{\sum \overline{R}_{10}}$$

El valor ajustado del Parámetro de estacionalidad (\widehat{R}_{10}) es:

$$\widehat{R}_{10} = 0,71636 * \frac{12}{11,99985}$$

$$\widehat{R}_{10} = 0,71637$$

La siguiente tabla muestra el proceso anterior para los doce periodos:

CUADRO 10-30

La española: Salchicha, parámetro de estacionalidad, 2014

R_{i0}	Primer año	Segundo año	Promedio	Ajustado
Enero	0,81595	0,61678	0,71636	0,71637
Febrero	0,97177	0,98865	0,98021	0,98022
Marzo	1,14940	1,16941	1,15940	1,15942
Abril	0,93541	0,95191	0,94366	0,94367
Mayo	0,87923	0,89459	0,88691	0,88692
Junio	1,68924	1,71895	1,70410	1,70412
Julio	0,62762	0,63873	0,63318	0,63319
Agosto	1,20476	1,22604	1,21540	1,21542
Septiembre	0,85202	0,86708	0,85955	0,85956
Octubre	1,12689	1,14691	1,13690	1,13691
Noviembre	0,92137	0,93774	0,92956	0,92957
Diciembre	0,82726	0,84200	0,83463	0,83464
Sumatoria			11,99985	

Fuente: Elaborado en base a la ecuación del parámetro estacional

La siguiente tabla resume los parámetros iniciales de los diferentes productos.

CUADRO 10-31

La española: Parámetros Iniciales del modelo Holt-Winters, 2014

Producto	Salchicha	Chorizo	Mortadela	Jamón	Salame
At	16158,2083	8759,6250	1638,8333	651,4583	426,9583
Tt	65,3403	16,2431	2,4444	0,9375	0,5903
Rt1	0,9219	0,7164	0,8222	0,9096	0,9090
Rt2	0,8920	0,9802	0,9993	1,0123	1,0120
Rt3	1,2655	1,1594	1,0834	1,0252	1,0245
Rt4	0,9231	0,9437	0,9816	0,9829	0,9826
Rt5	0,8506	0,8869	0,8794	0,9596	0,9605
Rt6	1,5079	1,7041	0,9262	1,0137	0,9765
Rt7	0,3213	0,6332	0,5842	0,8876	0,8877
Rt8	0,8396	1,2154	0,8403	0,9970	0,9969
Rt9	0,7092	0,8596	0,7999	0,9603	0,9600
Rt10	1,0094	1,1369	1,2830	1,0988	1,1204
Rt11	1,4393	0,9296	1,4292	1,0948	1,0948
Rt12	1,3203	0,8346	1,3714	1,0582	1,0751

Fuente: Elaborado en base al cálculo de parámetros iniciales.

Ya calculados nuestros parámetros iniciales ahora vamos a implementarlos a el modelo de Holt-Winters. El modelo solo puede ser resuelto por un algoritmo de simulación. En este caso vamos a utilizar (Microsoft Office Excel 2013).

Calculo de los coeficientes del modelo Holt-Winters

Como mencionamos anteriormente el cálculo de los coeficientes del modelo es complejo, no vamos a incurrir en temas que son tecnología de simulación. El algoritmo para realizar esta simulación es muy complejo hasta para un programador experto. Los programas aptos para optimizar el modelo de Holt-Winters son: el complemento SOLVER de (Microsoft Office Excel 2013) y el complemento COM CristalBall para (Microsoft Office Excel 2013) de ORACLE.

En este caso vamos a utilizar el complemento SOLVER de ya este no está establecido como proceso estándar, es decir que tiene mayor flexibilidad para cambiar el método de resolución por ejemplo, el único problema es que no muestra los estadísticos necesarios para validar el modelo como la prueba de Durbin-Watson para la autocorrelación (lo cual por cierto es bastante complejo corregir). Pero esto no afecta de manera significativa al modelo, ya que según (Gujarati y Porter 2010) para una serie de tiempo se suponen que los datos son independientes, aunque claro el autor lleva el tema desde una perspectiva

netamente econométrico, es decir no reconoce al modelo de Holt – Winters, ya que se desvía del tema, sino que utiliza el modelo Autoregresivo de Media móvil Integrado (ARIMA) con la metodología de Box-Jenkins.

Sin salir tanto del tema aquí presentamos los coeficientes calculados en el complemento SOLVER de (Microsoft Office Excel 2013). La siguiente figura muestra la ubicación de los parámetros.

FIGURA 10-2
La española: Modelo de Holt- Winters para la salchicha.2014

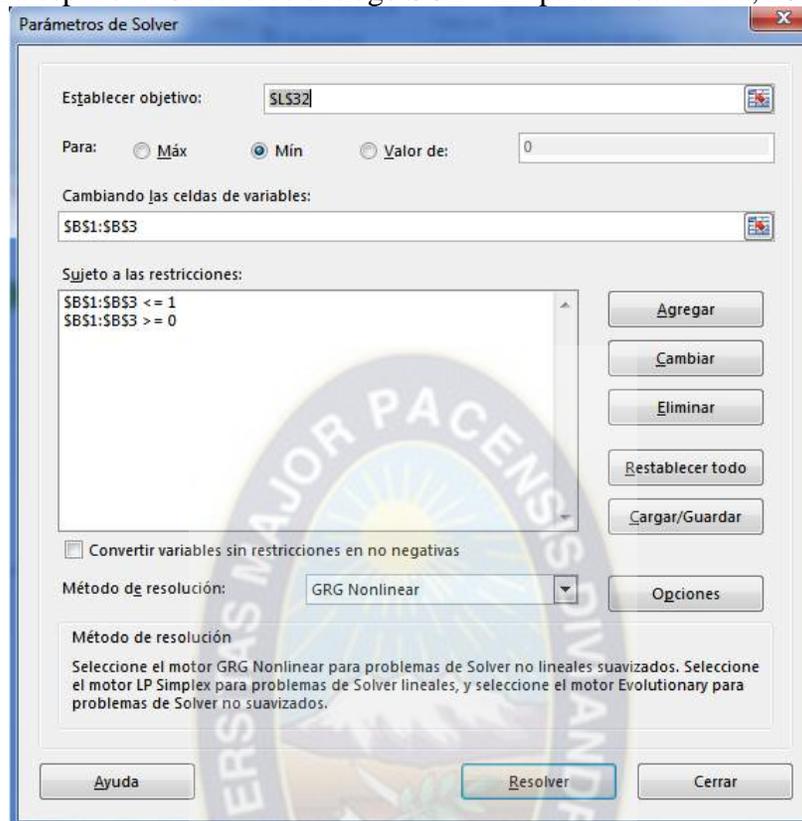
	A	B	C	D	E	F	G	H	I	J	K	L
1	Alfa	0										
2	Beta	0										
3	Gama	0										
4	L	12										
5	K	1										
6	r	2										
METODO DE WINTERS: PRONOSTICO DE UNA SERIE DE TIEMPO DIVIDIDA EN MESES												
7	Año	Mes	D _t	Prom.	R _t	R _t (Prom.)	R _t (Norm.)	Periodo	F _t	A ₀	T ₀	ECM
8	2000	1	13384		0,82664	0,92110	0,92187	-11		16158,20833	65,34028	796993,11
9		2	14615		0,89904	0,89121	0,89196	-10				
10		3	20817		1,27543	1,26441	1,26548	-9				
11		4	15244		0,93026	0,92232	0,92310	-8				
12		5	14102		0,85715	0,84992	0,85064	-7				
13		6	25095		1,51929	1,50660	1,50787	-6				
14		7	5368		0,32371	0,32103	0,32131	-5				
15		8	14081		0,84579	0,83888	0,83959	-4				
16		9	11939		0,71433	0,70858	0,70918	-3				
17		10	17059		1,01669	1,00857	1,00942	-2				
18		11	24416		1,44951	1,43807	1,43928	-1				
19		12	22483	16550,25	1,32960	1,31918	1,32030	0		16158,20833	65,34028	
20	2001	1	17239		1,01555	Suma de Rt Prom.	0,92187	1	14956,07	16223,54861	65,34028	5211749,7
21		2	15053		0,88338	11,98987	0,89196	2	14529,06	16288,88889	65,34028	274516,41
22		3	21440		1,25339		1,26548	3	20695,89	16354,22917	65,34028	553693,99
23		4	15701		0,91439		0,92310	4	15156,94	16419,56944	65,34028	295996,87
24		5	14525		0,84270		0,85064	5	14022,73	16484,90972	65,34028	252276,47
25		6	25847		1,49390		1,50787	6	24955,61	16550,25000	65,34028	794577,79
26		7	5529		0,31836		0,32131	7	5338,68	16615,59028	65,34028	36220,815
27		8	14503		0,83196		0,83959	8	14005,06	16680,93056	65,34028	247941,65
28		9	12298		0,70284		0,70918	9	11876,13	16746,27083	65,34028	177973,47
29		10	17571		1,00045		1,00942	10	16970,05	16811,61111	65,34028	361138,74
30		11	25149		1,42662		1,43928	11	24290,69	16876,95139	65,34028	736702,66
31		12	23157	17334,333	1,30877		1,32030	12	22368,88	16942,29167	65,34028	621128,73

Fuente: Elaborado en (Microsoft Office Excel 2013) en base al cálculo de parámetros y coeficientes iniciales en cero

Entonces vamos a abrir el complemento SOLVER que se encuentra en la pestaña DATOS en la sección ANALISIS. Vamos a utilizar el Error mínimo cuadrado como objetivo del modelo. La figura siguiente muestra el cuadro del complemento con los parámetros llenados.

FIGURA 10-3

La española: Cuadro de dialogo SOLVER para la salchicha, 2014



Fuente: Elaborado en (Microsoft Office Excel 2013) en base a la FIGURA 10-2

Entonces cambiamos el Método de Resolución a EVOLUTIONARY oprimimos en RESOLVER y luego aceptamos los cambios. El siguiente cuadro muestra los coeficientes calculados por el programa que optimizan el objetivo del modelo.

CUADRO 10-32

La española: Coeficientes del modelo Holt-Winters, 2014

Producto	Salchicha	Chorizo	Mortadela	Jamón	Salame
Alfa	0,23677	0,21909	0,02555	0,02922	1,00000
Beta	0,00164	0,00000	0,27252	1,00000	0,00000
Gama	0,66359	0,19602	0,34494	0,41379	0,19667

Fuente: Elaborado en base al cálculo de coeficientes

Entonces tenemos el pronóstico para el año 2014 de todos los productos aplicando el modelo de Holt-Winters Aditivo. El siguiente cuadro muestra el pronóstico de la demanda para todos los productos.

CUADRO 10-33

La española: Pronostico de la demanda [Kg], 2014

Periodo	Salchicha	Chorizo	Mortadela	Jamón	Salame
ene-14	95.763	19.682	13.593	9.200	4.595
feb-14	88.489	21.937	13.763	9.398	4.695
mar-14	106.620	23.892	13.580	9.571	4.782
abr-14	101.351	22.318	13.510	9.477	4.735
may-14	98.909	20.123	13.296	9.335	4.663
jun-14	170.137	20.097	12.807	9.358	4.674
jul-14	89.558	14.451	13.193	8.968	4.478
ago-14	96.011	18.500	13.186	9.303	4.647
sep-14	88.323	18.295	14.664	9.312	4.651
oct-14	76.885	27.684	14.593	10.057	5.027
nov-14	84.438	31.186	14.127	10.346	5.172
dic-14	83.886	31.806	13.929	10.415	5.208

Fuente: Elaborado en base al modelo de Holt-Winters.



CAPÍTULO 11. DISEÑO DE MÉTODOS

11.1 Introducción

La programación lineal es la herramienta que nos permite optimizar un sistema, transformado el sistema real en un modelo matemático.

11.2 Fases de la programación lineal

DIAGRAMA 11-1

Fases de la programación entera, 2014



Fuente: Elaborado en base a (Wayne L. 2003)

11.2.1 Definición

La programación entera es un método de programación lineal para optimizar un modelo de variables discretas, de lo cual se obtienen una o más opciones de optimización, con la cual podemos alterar parámetros para descubrir un nuevo valor óptimo.

11.2.2 Alcance

Las dimensiones de la programación son: recursos temporales, recursos materiales, recursos monetarios, mercado y recursos de informática. Siendo el objetivo la dimisión monetaria.

Vamos a mencionar en claro que no está dentro del alcance del trabajo un sistema de información geográfico.

11.2.3 Estructura del sistema de información

Ahora ya calculamos todos nuestros parámetros en el CAPÍTULO 11. DISEÑO DE MÉTODOS Solo nos falta llenar

Perspectiva financiera

Esta perspectiva es reflejada en el módulo de contabilidad. Este módulo contiene la administración de la facturación, administración de libros contables, administración de estados financieros, administración de presupuesto e indicadores financieros.

Perspectiva mercado

Esta perspectiva es reflejada en tres módulos. Primero está el módulo de mercado donde están los cuadros de dialogo donde se calculan los parámetros de mercado como ser demanda y precio, luego tenemos el módulo de aprovisionamiento donde se gestiona los proveedores y tercero está el módulos de comercialización donde se gestiona los clientes.

Estos módulos alimentan a los módulos de perspectiva de procesos.

Perspectiva de procesos

Esta perspectiva es reflejada en el módulo de producción. Este módulo contiene la programación de requerimientos empresariales (ERP), administración de inventarios, administración de productos, programación detallada por producto y la administración de la cadena de valor.

Perspectiva de personal

Esta perspectiva es reflejada en el módulo de administración. Este módulo contiene la administración de personal, el manual de funciones y el organigrama.

11.3 Modelación

La variable dependiente es el Ingreso total (Z), la cual está definida por una combinación lineal de las variables dependientes: Cantidad de Kilogramos a producir de Salchichas en Horario Diurno (A_d), Cantidad de Kilogramos a producir de Salchichas en Horario Nocturno (A_n), Cantidad de Kilogramos a producir de Chorizo en el Horario Diurno (B_d), Cantidad de Kilogramos a producir de Chorizo en Horario Nocturno (B_n), Cantidad de Kilogramos a producir de Mortadela en Horario Diurno (C_d), Cantidad de Kilogramos a producir de Mortadela en Horario Nocturno (C_n), Cantidad de Kilogramos a producir de Jamón en Horario Diurno (D_d), Cantidad de Kilogramos a producir de Jamón en Horario

Nocturno (Dn), Cantidad de Kilogramo a producir de Salame en Horario Diurno (Ed), Cantidad de Kilogramo a producir de Salame en Horario Nocturno (En).

Las variable dependiente esta expresado en [boliviano/año] y las variables secundarias en [Kg].

11.3.1 Definición de la función objetivo

Los parámetros de la función objetivo están expresados en [bolivianos/Kg], este factor de transformación el coeficiente tecnológico (el cual no es más que el valor agregado por horario). El siguiente cuadro muestra el precio, costo diurno, costo nocturno y coeficiente tecnológico.

CUADRO 11-1

La española: Coeficientes tecnológicos [Bolivianos/Kg], 2014

	Precio	Costo Diurno	Costo Nocturno	Coeficiente día	Coeficiente noche
Salchicha	38,00	21,00	27,30	17,00	10,70
Chorizo	41,00	20,00	26,00	21,00	15,00
Mortadela	35,00	23,00	29,90	12,00	5,10
Jamón	36,00	24,00	31,20	12,00	4,80
Salame	47,00	25,00	32,50	22,00	14,50

Fuente: Elaborado en base a Gerencia General

Entonces la combinación lineal está dada por la siguiente ecuación:

$$Z = 17Ad + 21Bd + 12Cd + 12Dd + 22Ed + 10,7An + 15Bn + 5,1Cn + 4,8Dn + 14,5En$$

11.3.2 Definición de las restricciones

En la dimensión de recursos temporales tenemos la mano de obra expresada en [hora/año] y los coeficientes tecnológicos expresados en [hora/Kg], la restricción es “menor o igual que”. Este coeficiente tecnológico es el tiempo empleado o tiempo estándar. Para Producción de Salchichas (A) el tiempo estándar es 0,3 [hora/Kg], para Producción de Chorizo (B) el tiempo estándar es 0,35 [hora/Kg] y para Producción de otros productos secundarios (C) el precio medio es 5 [hora/Kg]. La limitación está dada por la cantidad de horas de trabajo que se tienen al año:

$$15 [\text{hombre}] * \frac{44 \left[\frac{\text{hora}}{\text{semana}} \right]}{1[\text{hombre}]} * \frac{48,143 [\text{semana}]}{1[\text{año}]} = 31775 \left[\frac{\text{hora}}{\text{año}} \right]$$

Entonces la combinación lineal está dada por la siguiente ecuación:

$$0,30A + 0,35B + 0,50C \leq 31775$$

La dimensión de recursos materiales compone de tres restricciones.

La primera restricción material es el requerimiento de carne de cerdo expresado en [Kg de I/año] y los coeficientes tecnológicos expresados en [Kg de I/Kg de P], la restricción es “menor o igual que”. Este coeficiente tecnológico es conocido como el coeficiente de Insumo-Producto. Para Producción de Salchichas (A) el coeficiente de insumo-producto es 1,2 [Kg de I/Kg de P], para Producción de Chorizo (B) el coeficiente de insumo-producto es 1,3 [Kg de I/Kg de P] y para Producción de otros productos secundarios (C) el coeficiente de insumo-producto es 1,5 [Kg de I/Kg de P]. La limitación está dada por la cantidad de requerimientos de carne de cerdo por año:

$$3000 \left[\frac{\text{Kg de I}}{\text{año}} \right]$$

Entonces la combinación lineal está dada por la siguiente ecuación:

$$1,2A + 1,3B + 1,5C \leq 30000$$

La segunda restricción material es el requerimiento de carne de res expresado en [Kg de I/año] y los coeficientes tecnológicos expresados en [Kg de I/Kg de P], la restricción es “menor o igual que”. Este coeficiente tecnológico es conocido como el coeficiente de Insumo-Producto. Para Producción de Salchichas (A) el coeficiente de insumo-producto es 1,25 [Kg de I/Kg de P], para Producción de Chorizo (B) el coeficiente de insumo-producto es 1,34 [Kg de I/Kg de P] y para Producción de otros productos secundarios (C) el coeficiente de insumo-producto es 1,40 [Kg de I/Kg de P]. La limitación está dada por la cantidad de requerimientos de carne de cerdo por año:

$$4000 \left[\frac{\text{Kg de I}}{\text{año}} \right]$$

Entonces la combinación lineal está dada por la siguiente ecuación:

$$1,25A + 1,34B + 1,40C \leq 40000$$

La tercera restricción material es el requerimiento de carne de pollo expresado en [Kg de I/año] y los coeficientes tecnológicos expresados en [Kg de I/Kg de P], la restricción es “menor o igual que”. Este coeficiente tecnológico es conocido como el coeficiente de Insumo-Producto. Para Producción de Salchichas (A) el coeficiente de insumo-producto es 1,12 [Kg de I/Kg de P], para Producción de Chorizo (B) el coeficiente

de insumo-producto es 1,22 [Kg de I/Kg de P] y para Producción de otros productos secundarios (C) el coeficiente de insumo-producto es 1,38 [Kg de I/Kg de P]. La limitación está dada por la cantidad de requerimientos de carne de cerdo por año:

$$5000 \left[\frac{\text{Kg de I}}{\text{año}} \right]$$

Entonces la combinación lineal está dada por la siguiente ecuación:

$$1,12A + 1,22B + 1,38C \leq 50000$$

En la dimensión de recursos monetarios tenemos el presupuesto anual expresado en [boliviano/año] y los coeficientes tecnológicos expresados en [boliviano/Kg], la restricción es “menor o igual que”. Este coeficiente tecnológico es establecido al principio de la gestión. Para Producción de Salchichas (A) el presupuesto es 30 [boliviano/Kg], para Producción de Chorizo (B) el presupuesto es 35 [boliviano/Kg] y para Producción de otros productos secundarios (C) el presupuesto es 40 [boliviano/Kg]. La limitación está dada por la cantidad de cantidad de dinero disponible para esa gestión:

$$150000 \left[\frac{\text{boliviano}}{\text{año}} \right]$$

Entonces la combinación lineal está dada por la siguiente ecuación:

$$30A + 35B + 40C \leq 150000$$

En la dimensión de mercado tenemos a la demanda general del mercado expresado en [boliviano/año] y los coeficientes tecnológicos expresados en [boliviano/Kg], la restricción podría ser “igual que” pero con una perspectiva optimista vamos a ponerla como “mayor o igual que”. Este coeficiente es el precio de mercado que es muy similar al precio medio³². Para Producción de Salchichas (A) el precio de mercado es 37,35 [boliviano/Kg], para Producción de Chorizo (B) el precio de mercado es 42,50 [boliviano/Kg] y para Producción de otros productos secundarios (C) el precio medio es 40,30 [boliviano/Kg]. La limitación está dada por la cantidad de mercado proyectada según el historial:

$$26000 \left[\frac{\text{boliviano}}{\text{año}} \right]$$

Entonces la combinación lineal está dada por la siguiente ecuación:

$$37,35A + 42,5B + 40,30C \geq 26000$$

³² Para no incurrir en problemas de soluciones múltiples no vamos a utilizar los mismos coeficientes tecnológicos.

Para esta dimensión también tenemos requerimientos que podemos aplicarlos en forma de restricción o en forma de objetivo, por esta ocasión vamos a emplearlo como dos restricciones más. La producción de los productos principales A y B tienen requerimientos mínimos. Entonces tenemos las restricciones directas en función a la demanda anual que se tiene:

$$A \geq 2000$$

$$B \geq 1800$$

Y por último en la dimensión de recursos informáticos, que en esta etapa no podremos formular su restricción, por falta de información de la ecuaciones de transferencia.

Entonces el modelo se resume como:

$$Z = 17Ad + 21Bd + 12Cd + 12Dd + 22Ed + 10,7An + 15Bn + 5,1Cn + 4,8Dn + 14,5En$$

$$0,30A + 0,35B + 0,50C \leq 31775$$

$$1,2A + 1,3B + 1,5C \leq 30000$$

$$1,25A + 1,34B + 1,40C \leq 40000$$

$$1,12A + 1,22B + 1,38C \leq 50000$$

$$30A + 35B + 40C \leq 150000$$

$$37,35A + 42,5B + 40,30C \geq 26000$$

$$A \geq 2000$$

$$B \geq 1800$$

$$C \geq 0$$

11.3.3 Validación del modelo

En esta etapa del trabajo no hemos validado ninguno de los datos que presentamos porque no estamos seguros de los procesos que vamos a ejecutar. Es por esta razón que no validamos el modelo general ni los resultados, pero tomamos como referencia los datos.

11.3.4 Solución del modelo

Para solucionar el problema hemos empleado la aplicación Lingo 80 para win32. Ahora mencionamos algunas de las virtudes que requerimos del programa. El reporte que presenta la aplicación nos presenta la solución de las variables y la satisfacción de las restricciones,

además de los intervalos de confianza para las variables y las restricciones para análisis posteriores, como el análisis de sensibilidad y analizar la probabilidad de éxito.

Ahora vamos a mostrar el reporte como impresión de pantalla:

CUADRO 11-2

Solución de las variables y satisfacción de las restricciones, 2014

```

LP OPTIMUM FOUND AT STEP      0

      OBJECTIVE FUNCTION VALUE

    1)      177800.0

      VARIABLE                VALUE                REDUCED COST
      A                2200.000000                0.000000
      B                2400.000000                0.000000
      C                 0.000000                3.000000

      ROW    SLACK OR SURPLUS    DUAL PRICES
    1)      30275.000000                0.000000
    2)      24240.000000                0.000000
    3)      34034.000000                0.000000
    4)      44608.000000                0.000000
    5)           0.000000                1.200000
    6)      158170.000000                0.000000
    7)           0.000000               -1.000000
    8)         600.000000                0.000000
    9)           0.000000                0.000000

NO. ITERATIONS=          2
  
```

Fuente: Elaboración propia

CUADRO 11-3

La española: Análisis de sensibilidad, 2014

```

RANGES IN WHICH THE BASIS IS UNCHANGED:

      OBJ COEFFICIENT RANGES
      VARIABLE      CURRENT      ALLOWABLE      ALLOWABLE
                   COEF      INCREASE      DECREASE
      A              35.000000      1.000000      INFINITY
      B              42.000000      INFINITY      1.166667
      C              45.000000      3.000002      INFINITY

      RIGHTHAND SIDE RANGES
      ROW      CURRENT      ALLOWABLE      ALLOWABLE
              RHS      INCREASE      DECREASE
    1          31775.000000      INFINITY      30275.000000
    2          30000.000000      INFINITY      24240.000000
    3          40000.000000      INFINITY      34034.000000
    4          50000.000000      INFINITY      44608.000000
    5          150000.000000      652615.375000      21000.000000
    6          26000.000000      158170.000000      INFINITY
    7           2200.000000      700.000000      2200.000000
    8           1800.000000      600.000000      INFINITY
    9              0.000000      0.000000      INFINITY
  
```

Fuente: Elaboración propia

Por ahora tenemos nuestra solución inicial y la mayoría de las restricciones insatisfechas. Las restricciones más importantes son la de tiempo de mano de obra y la de presupuesto, y cómo podemos ver la restricción del tiempo esta insatisfecha.

Para solucionar este inconveniente vamos a segmentar la restricción por proceso y el modelo nos mostrara si es adecuado o no ejecutar el proceso, además incluiremos la opción de ejecutar o no con proceso de control mediante variables estocásticas. Y en esa etapa completaremos la dimensión de recursos informáticos.



CAPÍTULO 12. DISEÑO DE LA INTERFAZ

12.1 Introducción

12.1.1 Definiciones

Aplicación informática, es el desarrollo de un sistema asistido por computadora para ejecutar una tarea específica.

Interface, es un punto de encuentro entre dos sistemas diferentes, en el que interactúan.

Algoritmo, es un conjunto prescrito de instrucciones o reglas, ordenadas y finitas que permite realizar una actividad mediante pasos sucesivos

Lenguaje de programación, es un sistema de códigos usados para la interacción del usuario y el computador.

12.1.2 Bases para el desarrollo de la aplicación

Código fuente.- Es la transcripción del algoritmo desarrollado en el lenguaje de programación y la presentación de la aplicación.

Entorno de desarrollo integrado.- Es el programa que desarrolla el algoritmo y la presentación de una aplicación informática.

12.1.3 Entorno de desarrollo integrado

Un entorno de desarrollo integrado (IDE) es un programa informático compuesto por un conjunto de herramientas de programación. Un IDE es un entorno que ha sido empaquetado como un programa de aplicación; es decir, que consiste en un editor de código, un compilador, un depurador y un constructor de interfaz gráfica (GUI). Los IDE pueden ser aplicaciones por sí solas o pueden ser parte de aplicaciones existentes. Los IDE proveen un marco de trabajo amigable para la mayoría de los lenguajes de programación tales como C++, PHP, Python, Java, C#, Delphi, Visual Basic, Gambas, etc. En algunos lenguajes, un IDE puede funcionar como un sistema en tiempo de ejecución en donde se permite usar el lenguaje de programación en forma interactiva, sin necesidad de trabajo orientado a archivos de texto, como es el caso de Smalltalk y Objective-C.

12.1.4 Lenguaje de programación

Un lenguaje de programación es un lenguaje formal diseñado para expresar procesos que pueden ser llevados a cabo por máquinas como las computadoras.

Pueden usarse para crear programas que controlen el comportamiento físico y lógico de una máquina, para expresar algoritmos con precisión, o como modo de comunicación humana. Está formado por un conjunto de símbolos y reglas sintácticas y semánticas que definen su estructura y el significado de sus elementos y expresiones. Al proceso por el cual se escribe, se prueba, se depura, se compila y se mantiene el código fuente de un programa informático se le llama programación.

12.1.5 Editor de texto

Un editor de texto es un programa que permite crear y modificar archivos digitales compuestos únicamente por un texto sin formato, conocidos comúnmente como archivos de texto o texto plano. El programa lee el archivo e interpreta los bytes leídos según el código de caracteres que usa el editor. Hoy en día es comúnmente de 7 u 8-bits en ASCII o UTF-8, rara vez EBCDIC.

Por ejemplo, un editor ASCII de 8 bits que lee el número binario 0110 0001 (decimal 97 o hexadecimal 61) en el archivo lo representará en la pantalla por la figura a, que el usuario reconoce como la letra "a" y ofrecerá al usuario las funciones necesarias para cambiar el número binario en el archivo.

Los editores de texto son incluidos en el sistema operativo o en algún paquete de software instalado y se usan cuando se deben crear o modificar archivos de texto como archivos de configuración, scripts o el código fuente de algún programa.

El archivo creado por un editor de texto incluye por convención en DOS y Microsoft Windows la extensión ".txt", aunque pueda ser cambiada a cualquier otra con posterioridad. Tanto Unix como Linux dan al usuario total libertad en la denominación de sus archivos.

Al trasladar archivos de texto de un sistema operativo a otro se debe considerar que existen al menos dos convenciones diferentes para señalar el término de una línea: Unix y Linux usan sólo retorno de carro en cambio Microsoft de carro y salto de línea.

12.1.6 Compilador

Un compilador es un programa informático que traduce un programa escrito en un lenguaje de programación a otro lenguaje de programación, generando un programa equivalente que la máquina será capaz de interpretar. Usualmente el segundo lenguaje es lenguaje de máquina, pero también puede ser un código intermedio (bytecode), o simplemente texto. Este proceso de traducción se conoce como compilación.

Un compilador es un programa que permite traducir el código fuente de un programa en lenguaje de alto nivel, a otro lenguaje de nivel inferior (típicamente lenguaje de máquina). De esta manera un programador puede diseñar un programa en un lenguaje mucho más cercano a cómo piensa un ser humano, para luego compilarlo a un programa más manejable por una computadora.

Como parte importante de este proceso de traducción, el compilador informa a su usuario de la presencia de errores en el programa fuente.

12.1.7 Interprete

En ciencias de la computación, intérprete o interpretador es un programa informático capaz de analizar y ejecutar otros programas, escritos en un lenguaje de alto nivel. Los intérpretes se diferencian de los compiladores en que mientras estos traducen un programa desde su descripción en un lenguaje de programación al código de máquina del sistema, los intérpretes sólo realizan la traducción a medida que sea necesaria, típicamente, instrucción por instrucción, y normalmente no guardan el resultado de dicha traducción.

Usando un intérprete, un solo archivo fuente puede producir resultados iguales incluso en sistemas sumamente diferentes (ejemplo. una PC y un PlayStation 3). Usando un compilador, un solo archivo fuente puede producir resultados iguales solo si es compilado a distintos ejecutables específicos a cada sistema.

Los programas interpretados suelen ser más lentos que los compilados debido a la necesidad de traducir el programa mientras se ejecuta, pero a cambio son más flexibles como entornos de programación y depuración (lo que se traduce, por ejemplo, en una mayor facilidad para reemplazar partes enteras del programa o añadir módulos completamente nuevos), y permiten ofrecer al programa interpretado un entorno no dependiente de la máquina donde se ejecuta el intérprete, sino del propio intérprete (lo que se conoce comúnmente como máquina virtual).

Para mejorar el desempeño, algunas implementaciones de programación de lenguajes de programación pueden interpretar o compilar el código fuente original en una más compacta forma intermedia y después traducir eso al código de máquina (ej. Perl, Python, MATLAB, y Ruby). Algunos aceptan los archivos fuente guardados en esta representación intermedia (ej. Python, UCSD Pascal y Java).

Comparando su actuación con la de un ser humano, un compilador equivale a un traductor profesional que, a partir de un texto, prepara otro independiente traducido a otra lengua,

mientras que un intérprete corresponde al intérprete humano, que traduce de viva voz las palabras que oye, sin dejar constancia por escrito.

En la actualidad, uno de los entornos más comunes de uso de los intérpretes informáticos es Internet, debido a la posibilidad que estos tienen de ejecutarse independientemente de la plataforma.

12.1.8 Depurador

Un depurador es un programa usado para probar y depurar (eliminar) los errores de otros programas (el programa "objetivo"). El código a ser examinado puede alternativamente estar corriendo en un simulador de conjunto de instrucciones (ISS), una técnica que permite gran potencia en su capacidad de detenerse cuando son encontradas condiciones específicas pero será típicamente algo más lento que ejecutando el código directamente en el apropiado (o el mismo) procesador. Algunos depuradores ofrecen dos modos de operación - la simulación parcial o completa, para limitar este impacto.

Si es un depurador de nivel de fuente o depurador simbólico, comúnmente ahora visto en entornos de desarrollo integrados, cuando el programa "se estrella" o alcanza una condición predefinida, la depuración típicamente muestra la posición en el código original. Si es un depurador de bajo nivel o un depurador de lenguaje de máquina, muestra la línea en la fuente desensamblado (a menos que también tenga acceso en línea al código fuente original y pueda exhibir la sección apropiada del código del ensamblador o del compilador). Un "estrellamiento" sucede cuando el programa no puede continuar normalmente debido a un error de programación. Por ejemplo, el programa pudo haber intentado usar una instrucción no disponible en la versión actual del CPU o haber intentado tener acceso a memoria protegida o no disponible.

Típicamente, los depuradores también ofrecen funciones más sofisticadas tales como correr un programa paso a paso (un paso o animación del programa), parar el programa (breaking), es decir, pausar el programa para examinar el estado actual en cierto evento o instrucción especificada por medio de un breakpoint, y el seguimiento de valores de algunas variables. Algunos depuradores tienen la capacidad de modificar el estado del programa mientras que está corriendo, en vez de simplemente observarlo. También es posible continuar la ejecución en una posición diferente en el programa bypaseando un estrellamiento o error lógico.

La importancia de un buen depurador no puede ser exagerada. De hecho, la existencia y la calidad de tal herramienta para un lenguaje y una plataforma dadas a menudo puede ser el factor de decisión en su uso, incluso si otro lenguaje/plataforma es más adecuado para la tarea.[cita requerida] La ausencia de un depurador, una vez estando acostumbrado a usar uno, se ha dicho que "hace que usted se sienta como un hombre ciego en un cuarto oscuro buscando un gato negro que no está allí".¹ Sin embargo, el software puede (y a menudo) se comporta diferentemente corriendo bajo un depurador que normalmente, debido a los cambios inevitables que la presencia de un depurador hará a la temporización interna de un programa de software. Como resultado, incluso con una buena herramienta de depuración, es a menudo muy difícil rastrear problemas de tiempo de corrida en complejos sistemas distribuidos con múltiples hilos.

La misma funcionalidad que hace a un depurador útil para eliminar errores permite ser usado como herramienta de craqueo de software para evadir la protección anticopia, la gestión digital de derechos, y otras características de protección de software. A menudo también lo hace útil como herramienta general de verificación de pruebas, cobertura de fallas, o analizador de desempeño, especialmente si son mostradas las longitudes de trayectoria de instrucción.

La mayoría de los motores de depuración actuales, tales como gdb y dbx proporcionan interfaces basadas en línea de comandos. Los frontales de depuración son extensiones populares a los motores de depuración, que proporcionan integración al IDE, animación del programa, y características de visualización. Algunos depuradores de los primeros mainframes tales como el Oliver y el SIMON proporcionaron esta misma funcionalidad para el IBM System/360 y posteriores sistemas operativos, hasta los años 1970.

12.1.9 Cliente

El cliente es una aplicación informática o un ordenador que consume un servicio remoto en otro ordenador conocido como servidor, normalmente a través de una red de telecomunicaciones.

El término se usó inicialmente para los llamados terminales tontos, dispositivos que no eran capaces de ejecutar programas por sí mismos, pero podían conectarse e interactuar con computadores remotos por medio de una red y dejar que éste realizase todas las operaciones requeridas, mostrando luego los resultados al usuario. Se utilizaban sobre todo porque su

costo en esos momentos era mucho menor que el de un computador. Estos terminales tontos eran clientes de un computador mainframe por medio del tiempo compartido.

Actualmente se suele utilizar para referirse a programas que requieren específicamente una conexión a otro programa, al que se denomina servidor y que suele estar en otra máquina. Ya no se utilizan por criterios de costo, sino para obtener datos externos (por ejemplo páginas web, información bursátil o bases de datos), interactuar con otros usuarios a través de un gestor central (como por ejemplo los protocolos BitTorrent o IRC), compartir información con otros usuarios (servidores de archivos y otras aplicaciones Groupware) o utilizar recursos de los que no se dispone en la máquina local (por ejemplo impresión)

Uno de los clientes más utilizados, sobre todo por su versatilidad, es el navegador web. Muchos servidores son capaces de ofrecer sus servicios a través de un navegador web en lugar de requerir la instalación de un programa específico.

12.1.10 Interfaz gráfica de usuario

La interfaz gráfica de usuario (GUI) es un programa informático que actúa de interfaz de usuario, utilizando un conjunto de imágenes y objetos gráficos para representar la información y acciones disponibles en la interfaz. Su principal uso, consiste en proporcionar un entorno visual sencillo para permitir la comunicación con el sistema operativo de una máquina o computador.

Habitualmente las acciones se realizan mediante manipulación directa, para facilitar la interacción del usuario con la computadora. Surge como evolución de las interfaces de línea de comandos que se usaban para operar los primeros sistemas operativos y es pieza fundamental en un entorno gráfico. Como ejemplos de interfaz gráfica de usuario, cabe citar los entornos de escritorio Windows, el X-Window de GNU/Linux o el de Mac OS X, Aqua.

En el contexto del proceso de interacción persona-ordenador, la interfaz gráfica de usuario es el artefacto tecnológico de un sistema interactivo que posibilita, a través del uso y la representación del lenguaje visual, una interacción amigable con un sistema informático.

12.1.11 Metodología para el diseño de interfaz del sistema

Existen varias metodologías para crear un sistema informático pero el que tomaremos será el método del prototipo de sistemas ya que es una estrategia de desarrollo utilizada cuando no es posible determinar todos los requerimientos del usuario. Es por ello que incluye el

desarrollo interactivo o en continua evolución, donde el usuario participa de forma directa en el proceso.

Este método contiene condiciones únicas de aplicación, en donde los encargados del desarrollo tienen poca experiencia o información, o donde los costos y riesgos de que se cometa un error pueden ser altos.

Así mismo este método resulta útil para probar la facilidad del sistema e identificar los requerimientos del usuario, evaluar el diseño de un sistema o examinar el uso de una aplicación. El método del prototipo de sistemas consta de 5 etapas:

1). Identificación de requerimientos conocidos: La determinación de los requerimientos de una aplicación es tan importante para el método de desarrollo de prototipos como lo es para el ciclo de desarrollo de sistemas o análisis estructurado. Por consiguiente, antes de crear un prototipo, los analistas y usuario deben de trabajar juntos para identificar los requerimientos conocidos que tienen que satisfacer.

2). Desarrollo de un modelo de trabajo: Es fácil comenzar el proceso de construcción del prototipo con el desarrollo de un plan general que permita a los usuarios conocer lo que se espera de ellas y del proceso de desarrollo. Un cronograma para el inicio y el fin de la primera interacción es de gran ayuda. En el desarrollo del prototipo se preparan los siguientes componentes:

- El lenguaje para el dialogo o conversación entre el usuario y el sistema (visual studio.net)
- Pantallas y formatos para la entrada de datos(Diseño de la base de datos)
- Módulos esenciales de procesamiento(Interfaz para el usuario)
- Salida del sistema(Reportes o informes)

3). Utilización del prototipo: Es responsabilidad del usuario trabajar con el prototipo y evaluar sus características y operación. La experiencia del sistema bajo condiciones reales permite obtener la familiaridad indispensable para determinar los cambios o mejoras que sean necesarios, así como las características inadecuadas

4). Revisión del prototipo: Durante la evaluación los analistas de sistemas desean capturar información sobre los que les gusta y lo que les desagrada a los usuarios.

Los cambios al prototipo son planificados con los usuarios antes de llevarlos a cabo, sin embargo es el analista responsable de tales modificaciones.

5). **Repetición del proceso las veces que sea necesarias:** El proceso antes descrito se repite varias veces, el proceso finaliza cuando los usuarios y analistas están de acuerdo en que el sistema ha evolucionado lo suficiente como para incluir todas las características necesarias.

12.2 Código fuente

El código fuente para un sistema de este tamaño abarca varias líneas de código que se detallaron en el ANEXO E.³³ Es necesario detallarlo ya que si se habla de un sistema informático y no escribir su código fuente es como estudiar ingeniería y no titularse.

12.3 Diseño de la base de datos

Como se mencionó anteriormente se utilizó el programa MSSQL Server 2012 y a continuación se adjuntan las impresiones de pantalla de programa para que tengan una idea de cómo es la interfaz del programa

Impresiones de pantalla

FIGURA 12-1
La española: Tabla de empleados, 2014

Cod_empleado	Nombre	Direccion_empl	Telefono_empl	Cargo	Fecha_contrat...	Salario	Contraseña	Foto	CI
101	Julio L. Mendoza...	Calle 4 2 Obraje...	22112348	Gerente General	1980-04-20 00:...	7000,0000	JMGG101	<Binary data>	2345674
102	Pedro Villegas R...	Calle Los Tajibos...	22791371	Contador	1990-04-30 00:...	2500,0000	PVRCC102	<Binary data>	2342554
103	Marco Quispe A...	Calle Diaz Roman...	22222995	Encargado de Al...	1981-04-29 00:...	2500,0000	JJEA103	<Binary data>	4988475
105	Johnny Tronca A...	Plan 52 Mz G N I...	23834787	Cortador de Carne	1981-04-29 00:...	2500,0000	JPC105	<Binary data>	4345354
106	Oribidal Apaza ...	Calle Batallon Co...	2503106	Operador Cultur...	1985-08-16 00:...	2500,0000	CATOC106	<Binary data>	5312935
107	Gregorio Placido ...	Av. Holandesa N...	23817176	Gerente de Prod...	1985-06-03 00:...	6000,0000	CPAGP107	<Binary data>	5236648
108	Silvia Inea Barroso	Calle Juan Ruffo...	224086258	Secretaria	2005-07-04 00:...	2500,0000	SBRSE108	<Binary data>	2401942
109	Amyra Bultron C...	Pjs Maria Auxilia...	22420843	Asistente de Ger...	2009-12-07 00:...	2500,0000	ANCA6109	NUL	6928374
110	Eliado Cantuta C...	Av Huayna Potos...	22851148	Encargado de Al...	2000-02-07 00:...	2500,0000	ECCAM110	<Binary data>	2524246
111	Gonzalo Momani ...	Av 12 N J de Vill...	22860395	Ayudante Almac...	2004-06-07 00:...	2500,0000	GMLAA111	<Binary data>	6834441
112	Vicente Vargas ...	Calle San Salvador...	22841362	Jefe de Produccion	1999-02-01 00:...	2500,0000	VWAP112	NUL	6918734
113	Jesanto Condori ...	Calle Heri N 342...	22840894	Ayudante Opera...	2005-11-07 00:...	2500,0000	JCFAC113	NUL	5209964
114	Sergio Trujillo Yupa	Cjn JJ Perez N 4...	22844158	Embalador	1999-12-04 00:...	2500,0000	STVOE114	<Binary data>	5249666
115	Juan Triconá Yucra	Av. Franco Valle...	22823900	Encargado de H...	2005-01-03 00:...	2500,0000	JTYEO115	NUL	3657136
116	Desiderio Yanah...	Av. A Plan 528 ...	22851351	Encargado de C...	1999-11-01 00:...	2500,0000	DYVEC116	NUL	2943667
117	Pedro Lima Herrera	Calle 13 N 1578 ...	22384321	Ayudante Almac...	2009-10-05 00:...	2500,0000	PLHAM117	NUL	6982764
118	Carlos Espinoza ...	Calle Isaac Arias...	22834529	Ayudante Embut...	2009-03-02 00:...	2500,0000	CEAAE118	NUL	2398273
* NUL	NUL	NUL	NUL	NUL	NUL	NUL	NUL	NUL	NUL

Fuente: Elaborado en base a la base de datos en MSSQL Server 2012

³³ En la codificación no se encuentran los códigos de diseño y se borró una línea de código a propósito

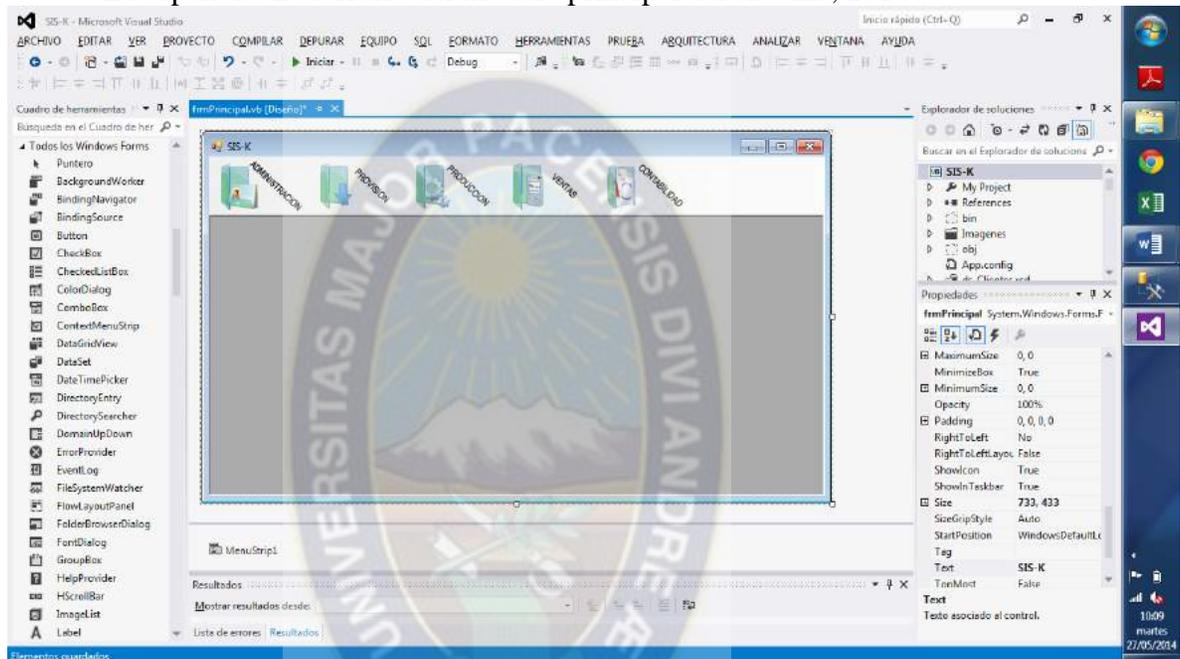
12.4 Interfaz para el usuario

Como se mencionó anteriormente utilizamos el programas (Microsoft Visual Studio 2012) para el desarrollo del sistema informático y después de determinar los requerimientos de la empresa se crearon los siguientes módulos.

Impresiones de pantalla

FIGURA 12-2

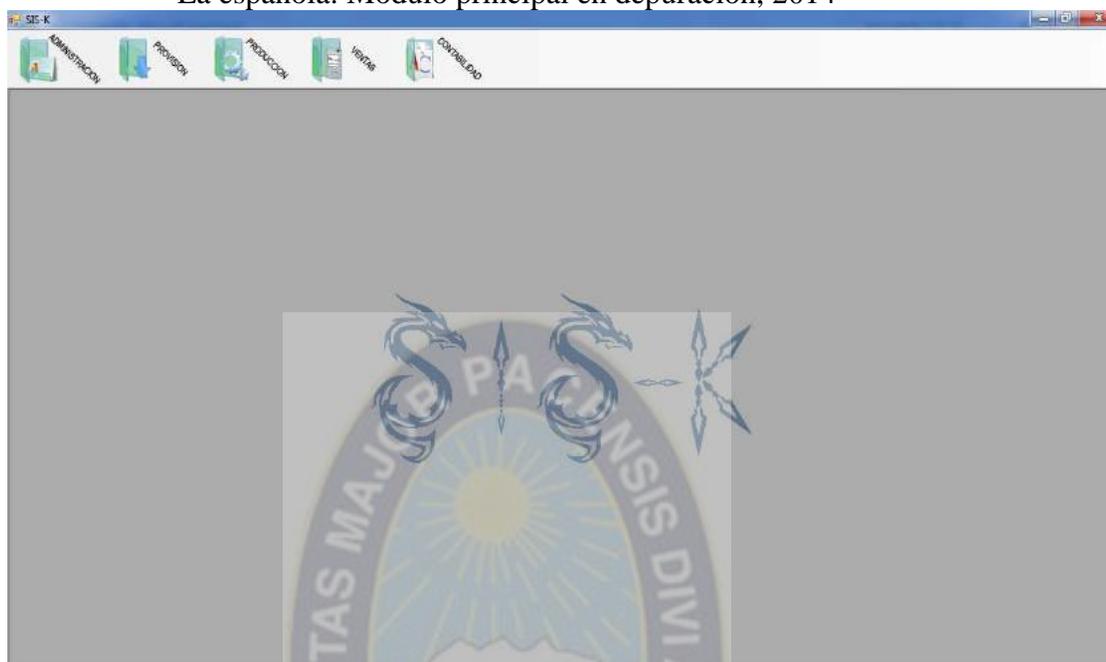
La española: Desarrollo del módulo principal del sistema, 2014



Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-3

La española: Modulo principal en depuración, 2014



Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

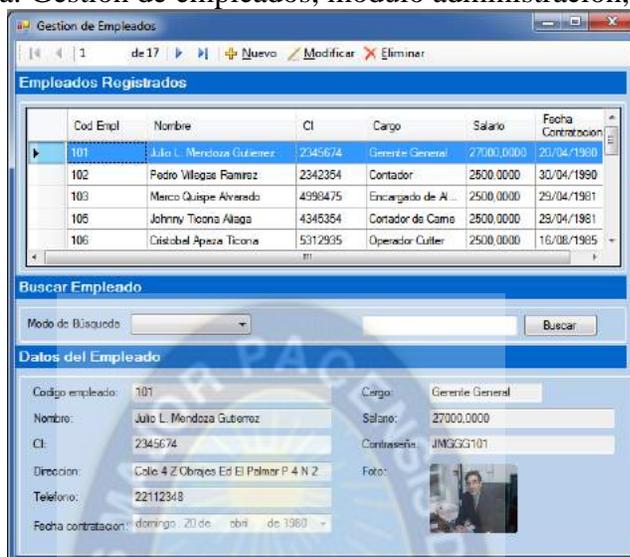
12.4.1 Administración

Este módulo realiza las funciones de: Gestión de empleados, Manual de funciones, Boletas de pago, Planilla de sueldos patronales. A continuación se muestran las impresiones de pantalla de las funciones.

Impresiones de pantalla

FIGURA 12-4

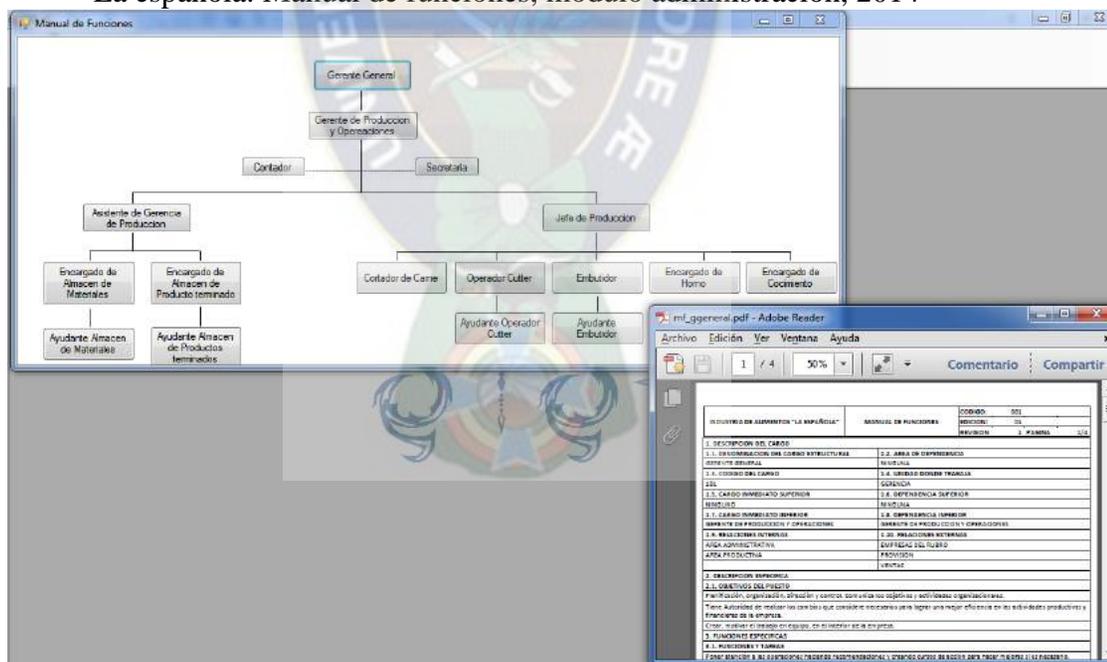
La española: Gestión de empleados, modulo administración, 2014



Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-5

La española: Manual de funciones, modulo administración, 2014



Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-6

La española: Boletas de Pago, modulo administración, 2014

Boleta de Pago

Correspondiente a **Marzo** de **2014**

(en Bolivianos)

Apellidos y Nombres: **Julio L. Mendoza Gutierrez** C.I.: **2345674**

Fecha de Ingreso: **20/04/1980** Cargo: **Gerente General**

INGRESOS		DESCUENTOS	
Sueldo Base:	27000 por 25 dias/mes	AFP:	5555,55
Bono de Produccion:	5000	RC-IVA:	4880,78
Monto Horas extra:	0 # HE: 0	Otros Descuentos:	0
Bono de Antigüedad:	13500		
Dominicales:			
Otros Bonos:	0		
Total Ganado:	45500,00	Total Descuentos:	10436,33
		Liquido Pagable:	35063,67

Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-7

La española: Planilla de sueldos patronales, 2014

Razon Social: **Industria de Alimentos de fiabres y Embutidos "La Española"**

No. Empleador:

Correspondiente al mes de **Marzo** de **2014**

PLANILLA DE SUELDOS Y SALARIOS
PERSONAL PERMANENTE
(en Bolivianos)

Nombre	CI	Cargo	FechaCont	Dias Trab	Sueldo	Bono Antig
Raul Gonzalez	234567	Gerente General	20/04/1980	30	7000,0000	0,0000
Silvia Ines Barroso	2401942	Secretaria	04/07/2005	25	2500,0000	13500,0...
Eladio Cantuta Cano	2624346	Encargado de Almacen de Mate...	07/02/2000	25	2500,0000	13500,0...
Johnny Ticona Aliaga	4345354	Cortador de Carne	29/04/1981	25	2500,0000	13500,0...
Gregorio Placido Asu...	5236648	Gerente de Produccion y Opere...	03/06/1985	25	6000,0000	13500,0...
Cristobal Apaza Ticona	5312935	Operador Cutter	16/08/1985	25	2500,0000	13500,0...
Gonzalo Mamani Lipe	6834441	Ayudante Almacen de Materiales	07/06/2004	25	2500,0000	13500,0...
Amaya Butron Cahuna	6928374	Asistente de Gerencia de Produ...	07/12/2009	25	2500,0000	13500,0...
Jose Jose	9998475	Encargado de Almacen de Prod...	29/04/1980	30	2500,0000	300,0000
Pedro Villenas	234235423	Contador	30/04/1980	30	2500,0000	300,0000

Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

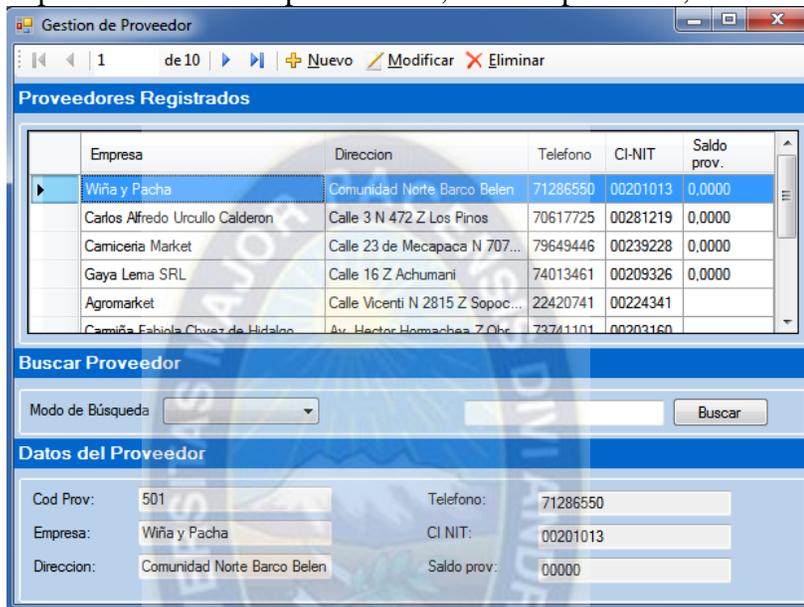
12.4.2 Provisión

Este módulo realiza las funciones de: Gestión de proveedores y Compras. A continuación de muestran las impresiones de pantalla de las funciones.

Impresiones de pantalla

FIGURA 12-8

La española: Gestión de proveedores, modulo provisión, 2014



Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-9

La española: Compras, modulo provisión, 2014



Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

12.4.3 Producción

Este módulo realiza las funciones de: Insumos, Productos terminados y Programa maestro de producción. A continuación de muestran las impresiones de pantalla de las funciones.

Impresiones de pantalla

FIGURA 12-10

La española: Productos terminados, modulo producción, 2014

The screenshot shows the 'Gestion de Productos' application window. At the top, there are navigation buttons: 'Nuevo', 'Modificar', and 'Eliminar'. Below this is a table titled 'Productos Registrados' with the following data:

Cod Producto	Producto	Precio	Existencias
201	Salchicha viena	35,7000	50000
202	Chorizo	40,0000	2988
203	Mortadela	41,0000	1000
204	Jamon	46,0000	2000
205	Salame	49,0000	500

Below the table is a 'Buscar Producto' section with a 'Modo de Búsqueda' dropdown and a 'Buscar' button. At the bottom is the 'Datos del Producto' section with the following fields:

Cod Producto:	201	Existencias:	50000
Descripcion:	Salchicha viena	Saldo Inventario:	1785000,0000
Precio:	35,7000		

Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-11

La española: Insumos, modulo producción, 2014

The screenshot shows the 'Gestion de Insumos' application window. At the top, there are navigation buttons: 'Nuevo', 'Modificar', and 'Eliminar'. Below this is a table titled 'Insumos Registrados' with the following data:

Cod Insumo	Descripcion	Precio	Existencias
601	Conservantes	20,0000	1224
602	Came de res	30,0000	263
603	Came de pollo	14,0000	426
604	Came de cerdo	27,0000	634
605	Tripa Natural	13,0000	1045

Below the table is a 'Buscar Insumo' section with a 'Modo de Búsqueda' dropdown and a 'Buscar' button. At the bottom is the 'Datos del Insumo' section with the following fields:

Cod Insumo:	601	Existencia fisica:	1224
Descripcion:	Conservantes	Saldo Inventario:	4480000
Costo:	200000		

Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-12

La española: Pronostico, Programa maestro de prod., mod. Prod., 2014

The screenshot shows a software window titled 'Programa Maestro de Producción' with tabs for 'Pronostico', 'Programacion Lineal', and 'MRP'. It displays two tables: 'Datos Historicos' and 'Pronostico'.

Periodo	Salchicha	Chorizo	Mortadela	Jamon	Salame	Total
01/01/2012	267,67	143,08	27,27	11,93	7,81	382,16
01/02/2012	292,29	170,71	32,64	13,09	8,57	425,96
01/03/2012	416,32	202,28	35,44	13,29	8,7	463,92
01/04/2012	304,87	164,94	32,18	12,79	8,37	433,34

Periodo	Salchicha	Chorizo	Mortadela	Jamon	Salame	Total
01/03/2014	106620	23892	13580	9571	4782	158445
01/04/2014	101351	22318	13510	9477	4735	151391
01/05/2014	98909	20123	13296	9335	4663	146326

Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-13

La española: Programación lineal, Programa maestro de prod., mod. Prod., 2014

The screenshot shows the 'Programa Maestro de Producción' software with the 'Programacion Lineal' tab selected. It displays a table of costs and prices, a calendar for 'Dias hab. al mes', and a 'Solucion del Modelo' table.

	Costos HO Dia	Costos HO Noche	Precios
Salchicha	21	27,3	38
Chorizo	21	27,3	41
Mortadela	21	27,3	35
Jamon	33	42,9	36
Salame	31	40,3	47

Dias hab. al mes	Ene	Feb	Mar	Abr	May	Jun	Jul	Ago	Sep	Oct	Nov	Dic
	22	19	21	21	22	20	21	22	23	22	22	23

Descripcion	Valores
Margen esperado	3047
Variables	
Cantidad de Lotes a producir de Salchicha en Horario Diurno	65
Cantidad de Lotes a producir de Salchicha en Horario Nocturno	0
Cantidad de lotes a producir de Chorizo en Horario Diurno	75
Cantidad de Lotes a producir de Chorizo en Horario Nocturno	0
Cantidad de Lotes a producir de Mortadela en Horario Diurno	40
Cantidad de Lotes a producir de Mortadela en Horario Nocturno	0
Cantidad de Lotes a producir de Jamón en Horario Diurno	18
Cantidad de Lotes a producir de Jamón en Horario Nocturno	0

Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-14
La española: MRP, 2014

Descripcion	Enero	Febrero	Marzo	Abril	Mayo	Junio	Total
Dias laborales por mes	22	19	21	21	22	20	125
Horas de produccion disponibles	1760	1520	1680	1680	1760	1600	10000
Produccion real	3385	2923	3231	3231	3385	3077	19232
Demanda pronosticada	2857	2766	3169	3028	2927	4341	19088
Inventario Final	528	685	747	950	1408	144	
Existencia de reserva	0	0	0	0	0	0	
Unidades sobrantes	528	685	747	950	1408	144	
Costo de faltantes	0	0	0	0	0	0	0
Costo de inventario	5280	6854	7473	9502	14083	1438	44630
Costo de tiempo corrido	22000	19000	21000	21000	22000	20000	125000
Costo Total							169630

Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

12.4.4 Ventas

Este módulo realiza las funciones de: Gestión de clientes y ventas. A continuación de muestran las impresiones de pantalla de las funciones.

Impresiones de pantalla

FIGURA 12-15
La española: Gestión de clientes, modulo ventas, 2014

Cod Cliente	Empresa	Telefono	CI-NIT	Deuda
101	Hipemaxi	22831283	15356340014	5,0000
102	MegaMarket	22252093	19872464626	5,0000
103	Ketal	22423630	19875923847	23,0000
104	Fidalga	22244732	17284947846	1500,0000
105	Fernando Contreras	22358575	12359861014	0,0000
106	Valentin Escobar Cam...	22470070	12077051225	0,0000

Datos del Cliente	
Cod Cliente:	101
Empresa:	Hipemaxi
Direccion:	Alpacoma Esq Av de Policia N 500 Z
Telefono:	22831283
CI-NIT:	153563400
Deuda:	50000

Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-18

La española: Cuentas, modulo contabilidad, 2014

The screenshot shows the 'Gestion de Cuentas' window with a table of registered accounts. Below the table is a search section and a 'Datos de la Cuenta' section with input fields for account details.

Cod Cuenta	Cuenta	Saldo
10001	Caja	1300,0000
10002	Caja Chica	0,0000
10003	Banco Mercantil	0,0000
10004	Cuentas por Cobrar	10,0000
10005	Hipemaxi	0,0000
10006	Entrega de Fondos a Rendir	0,0000

Buscar Cuenta

Modo de Búsqueda: [dropdown] [Buscar]

Datos de la Cuenta

Cod Cuenta:	100	Rubro:	1
Cuenta:	Caja	Grupo:	1
Saldo:	13000000	Titulo:	1100

Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-19

La española: Indicadores, modulo contabilidad, 2014

The screenshot displays the 'Indicadores Financieros' window with a grid of financial indicators. Each indicator has a name, a value, and a unit. A large watermark of the University of the Pacific is visible in the background.

INDICADORES FINANCIEROS	
Indicadores de liquidez	Indicadores de actividad
Capital de trabajo: 2960,00 Bs.	Rotacion de cuentas por cobrar: 5,14 dias
Indice Comiente: 1,23	Rotacion de cuentas por pagar: 9,47 dias
Prueba acida: 0,00	Rotacion de inventarios: 18,95 dias
Indicadores de endeudamiento	Rotacion de activos: 52,24 %
Endeudamiento total: 123,51 %	Indicadores de rentabilidad
Endeudamiento a largo plazo: 0,01	Margen Bruto: 45,71 %
Indice de apalancamiento: 1,66	Margen Neto: 12,00 %
Capacidad de pago de intereses: 0,80	Rentabilidad sobre activos: 0,06
	Rentabilidad sobre patrimonio: 0,08

Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-20

La española: Balance General, contabilidad, 2014

ACTIVO		PASIVO	
ACTIVO CORRIENTE		PASIVO CORRIENTE	
Caja	1300,00	Cuentas por Pagar	10,00
Caja chica	0,00	Proveedores por Pagar	0,00
Banco Mercantil	0,00	Sueldos y salarios por pagar	0,00
Cuentas por Cobrar	10,00	Impuesto IVA debito fiscal	290,00
Impuesto IVA credito fiscal	0,00	AFP Futuro de Bolivia	10,00
IUE por Compensar	0,00	AFP BBV Prevision	20,00
	1310,00		330,00
INVENTARIOS		PASIVO NO CORRIENTE	
Inventario de prod. term.	20,00	Prestamo Hipotecario	0,00
Inventario de Insumos	0,00	Prestamo documentario	10,00
	20,00	Prevision beneficios sociales	0,00
		Provision aguinaldos	0,00
ACTIVO NO CORRIENTE		TOTAL PASIVO	
ACTIVO FIJO			340,00
Terrenos	10,00	CAPITAL	
Edificios y construcciones	0,00	Capital	1000,00
Dep. Acum de edif y cons.	0,00	Aportes por capitalizar	0,00
Muebles y enseres	0,00	Reserva legal	0,00
Dep. Acum de muebl y ens	0,00		
Maquinaria y equipo	0,00	TOTAL CAPITAL	1000,00
Dep. Acum de Maq. y eq.	0,00		
Vehiculos	0,00		
Dep. Acum. de Vehiculos	0,00		
Herramientas	0,00		
Dep. Acum. de Herramientas	0,00		
Equipos de computacion	0,00		
Dep. Acum. de Eq. de Comp.	0,00		
	10,00		
ACTIVO DIFERIDO			
Gastos pagados x adelant.	0,00		
	0,00		
TOTAL ACTIVO	1340,00	TOTAL PASIVO Y CAPITAL	1340,00

Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-21

La española: Estado de Resultados, modulo contabilidad, 2014

ESTADO DE RESULTADOS	
Del 1 de Enero de 2014 al 27 de Mayo de 2014 (Expresado en Bs.)	
Ventas	700,00
- Costo de Produccion	380,00
Utilidad Bruta	320,00
- Gastos Administrativos	238,00
- Gastos de Ventas	60,00
Utilidad en Operaciones	22,00
+ Ingresos extraordinarios	0,00
- Gastos extraordinarios	6,00
Utilidad antes de Impuestos	16,00
- Impuestos	4,00
Utilidad Neta	16,00

Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

12.5 Salidas del sistema

El sistema puede generar los siguientes reportes o comprobantes. A continuación se muestran las impresiones de pantalla.

Impresiones de pantalla

FIGURA 12-22

La española: Reporte boleta de pago, modulo administración, 2014

Boleta de Pago

Correspondiente a de (en Bolivianos)

Apellidos y Nombres: C.I.:

Fecha de Ingreso: Cargo:

INGRESOS				DESCUENTOS	
Sueldo Base:	<input type="text" value="2500"/>	por <input type="text" value="25"/>	dias/mes	AFP:	<input type="text" value="2002,44"/>
Bono de Produccion:	<input type="text" value="0"/>			RC-IVA:	<input type="text" value="1559,68"/>
Monto Horas extra:	<input type="text" value="0"/>	# HE:	<input type="text" value="0"/>	Otros Descuentos:	<input type="text" value="0"/>
Bono de Antigüedad:	<input type="text" value="13500"/>				
Dominicales:	<input type="text" value="400,00"/>	<input checked="" type="checkbox"/>			
Otros Bonos:	<input type="text" value="0"/>				
Total Ganado:	<input type="text" value="16400,00"/>			Total Descuentos:	<input type="text" value="3562,12"/>
				Líquido Pagable:	<input type="text" value="12837,88"/>

Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-23

La española: Planilla de sueldos y salarios, modulo administración, 2014

Planilla de Sueldos y Salarios

Razon Social: Industria de Alimentos de fiabres y Embutidos "La Española"

No. Empleador:

Correspondiente al mes de de

PLANILA DE SUELDOS Y SALARIOS
PERSONAL PERMANENTE
(en Bolivianos)

Nombre	CI	Cargo	FechaCont	Dias Trab	Sueldo	Bono Antig	#HE	Costo HE	Bono Prod	Domi-nicales	Otros Bonos	Total Ganado
Raul Gonzalez	234567	Gerente General	20/04/1980	30	7000,0000	0,0000	0	0,0000	0,0000	12,0000	8,0000	7020,0000
Raul Gonzalez	234567	Gerente General	20/04/1980	30	7000,0000	50,0000	4	80,0000	300,0000	56,0000	0,0000	7486,0000
Pedro Villegas	234235423	Contador	30/04/1980	30	2500,0000	300,0000	5	100,0000	30,0000	0,0000	0,0000	2930,0000
Jose Jose	9998475	Encargado de Almacen de P...	29/04/1980	30	2500,0000	300,0000	5	100,0000	5,0000	0,0000	0,0000	2905,0000
Pedro Villegas	234235423	Contador	30/04/1980	30	2500,0000	1250,0000	4	3,6085	8,0000	333,3333	0,0000	4094,9419
Pedro Villegas	234235423	Contador	30/04/1980	25	2500,0000	1250,0000	0	3,6085	0,0000	400,0000	0,0000	4153,6085
Julio L. Mendoza Guti...	2345674	Gerente General	20/04/1980	22	7000,0000	3500,0000	0	0,0000	0,0000	0,0000	0,0000	10500,0000
Marco Quispe Alvarado	4998475	Encargado de Almacen de P...	29/04/1981	25	2500,0000	1250,0000	0	0,0000	0,0000	400,0000	0,0000	4150,0000
Johnny Ticona Aliaga	4345354	Cortador de Carne	29/04/1981	25	2500,0000	13500,0...	0	0,0000	0,0000	400,0000	0,0000	16400,0000
Cristobal Apaza Ticona	5312935	Operador Cutter	16/08/1985	25	2500,0000	13500,0	0	0,0000	0,0000	400,0000	0,0000	16400,0000

Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-24

La española: Registro de compra, modulo provisión, 2014



REGISTRO DE COMPRA

Codigo Compra:

Proveedor: 

Descripcion: 

Cantidad:

Importe:



Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-25

La española: Registro de venta, modulo ventas, 2014



LA ESPAÑOLA
C. URZULA GOYZUETA NO. 558
EL ALTO - BOLIVIA

F A C T U R A
ORIGINAL

NIT: 2703795012
FACTURA NO.: 13
AUTORIZACION NRO.: 390400921080

FECHA: 27/05/2014 10:59:00

SEÑOR (ES): Hipermaxi
NIT/CI: 15356340014

CANT	CONCEPTO	PRECIO	TOTAL
100	Salchicha viena	35,7000	3570,0000
200	Chorizo	40,0000	8000,0000
100	Mortadela	41,0000	4100,0000

TOTAL A PAGAR BS.: 15670,0000
EFFECTIVO: 15700
CAMBIO BS.: 30,0000

SON: 00/100 BS.



"La reproduccion total o parcial y/o el uso no autorizado de esta Nota Fiscal constituye un delito a ser sancionado conforme a Ley"

Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-26

La española: Indicadores, modulo contabilidad, 2014



INDICADORES FINANCIEROS



Indices de liquidez

Capital de trabajo:	2960,00	Bs.
Indice Corriente:	1,23	
Prueba acida:	0,00	

Indices de actividad

Rotacion de cuentas por cobrar:	5,14	dias
Rotacion de cuentas por pagar:	9,47	dias
Rotacion de inventarios:	18,95	dias
Rotacion de activos:	52,24	%

Indices de endeudamiento

Endeudamiento total:	123,51	%
Endeudamiento a largo plazo:	0,01	
Indice de apalancamiento:	1,66	
Capacidad de pago de intereses:	0,80	

Indices de rentabilidad

Margen Bruto:	45,71	%
Margen Neto:	12,00	%
Rentabilidad sobre activos:	0,06	
Rentabilidad sobre patrimonio:	0,08	

Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-27

La española: Estado de Resultados, modulo contabilidad, 2014

"Industria de Alimentos de Fiambres y embutidos La Española SRL"



ESTADO DE RESULTADOS

Del 1 de Enero de 2014 al 27 de Mayo de 2014
(Expresado en Bs.)

Ventas	700,00	
- Costo de Produccion	380,00	
Utilidad Bruta		320,00
- Gastos Administrativos	238,00	
- Gastos de Ventas	60,00	
Utilidad en Operaciones		22,00
+ Ingresos extraordinarios	0,00	
- Gastos extraordinarios	6,00	
Utilidad antes de Impuestos		16,00
- Impuestos	4,00	
Utilidad Neta		16,00

Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

FIGURA 12-28

La española: Balance General, modulo contabilidad, 2014

"Industria de Alimentos de Fiambres y embutidos La Española SRL"

BALANCE GENERAL
al 27 de Mayo de 2014
(Expresado en Bs.)



<u>ACTIVO</u>		<u>PASIVO</u>	
ACTIVO CORRIENTE		PASIVO CORRIENTE	
Caja	1300,00	Cuentas por Pagar	10,00
Caja chica	0,00	Proveedores por Pagar	0,00
Banco Mercantil	0,00	Sueldos y salarios por pagar	0,00
Cuentas por Cobrar	10,00	Impuesto IVA debito fiscal	290,00
Impuesto IVA credito fiscal	0,00	AFP Futuro de Bolivia	10,00
IUE por Compensar	0,00	AFP BBV Prevision	20,00
	1310,00		330,00
INVENTARIOS		PASIVO NO CORRIENTE	
Inventario de prod. term.	20,00	Prestamo Hipotecario	0,00
Inventario de Insumos	0,00	Prestamo documentario	10,00
	20,00	Prevision beneficios sociales	0,00
		Provision aguinaldos	0,00
ACTIVO NO CORRIENTE			10,00
ACTIVO FIJO		TOTAL PASIVO	340,00
Terrenos	10,00		
Edificios y construcciones	0,00	<u>CAPITAL</u>	
Dep. Acum de edif y cons.	0,00	Capital	1000,00
Muebles y enseres	0,00	Aportes por capitalizar	0,00
Dep. Acum de muebl y ens	0,00	Reserva legal	0,00
Maquinaria y equipo	0,00		
Dep. Acum de Maq. y eq.	0,00	TOTAL CAPITAL	1000,00
Vehiculos	0,00		
Dep. Acum. de Vehiculos	0,00		
Herramientas	0,00		
Dep. Acum. de Herramientas	0,00		
Equipos de computacion	0,00		
Dep. Acum. de Eq. de Comp.	0,00		
	10,00		
ACTIVO DIFERIDO			
Gastos pagados x adelant.	0,00		
	0,00		
TOTAL ACTIVO	1340,00	TOTAL PASIVO Y CAPITAL	1340,00

Fuente: Elaborado en base al proyecto SIS-K de Visual Studio 2012

12.6 Seguridad del sistema

Para cumplir los requerimientos de la empresa en seguridad del sistema. Se creó un módulo de autenticación que se inicia antes del módulo principal, para gestionar los permisos según tipo de usuario, por ejemplo, el Contador solo puede acceder al módulo de contabilidad y no a los otros módulos. A continuación se muestran las impresiones de pantalla.

FIGURA 12-29

La española: Autenticación, 2014



Fuente: Elaborado en base al módulo de autenticación

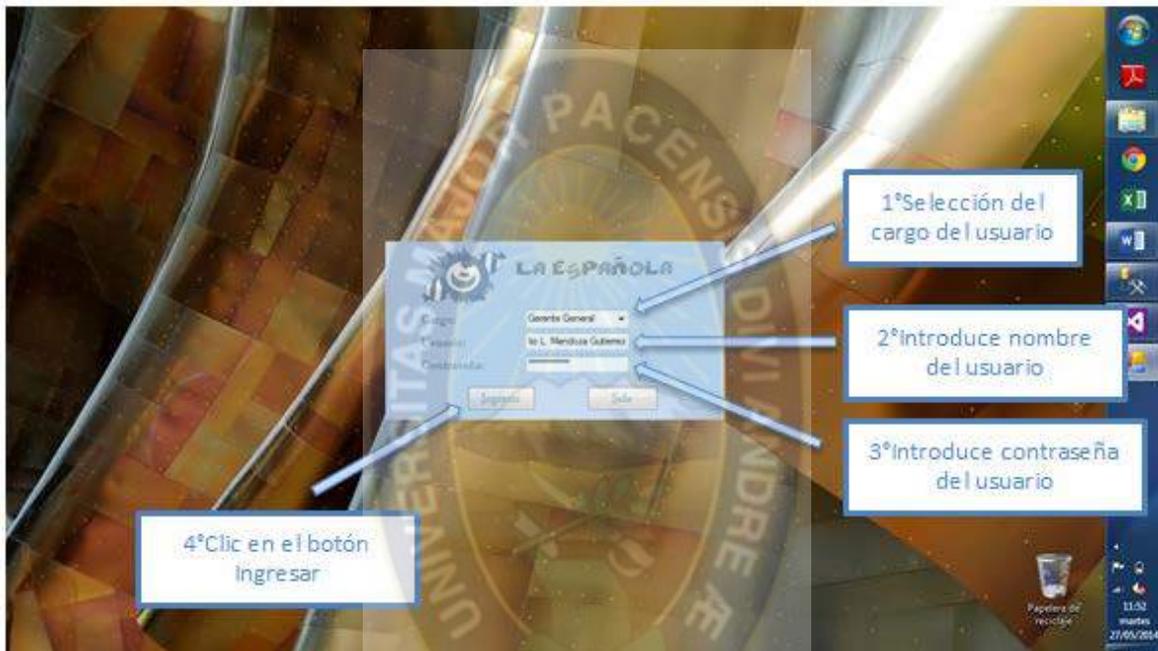
CAPÍTULO 13. MANUAL DEL USUARIO

13.1 Ingreso al sistema.

Para ingresar al sistema SIS-K primero debe llenar correctamente los campos requeridos de la autenticación de programa para ello se deben seguir los siguientes pasos:

FIGURA 13-1

La española: Ingreso al Sistema, 2014



Fuente: Elaborado en base al módulo de autenticación

13.2 Entorno del sistema

El sistema SIS-K tiene 5 módulos mediante los cuales administra la empresa estos son:

FIGURA 13-2

La española: Entorno del Sistema, 2014



Fuente: Elaborado en base sistema central SIS-K

13.3 Módulo de administración

Al dar clic en la pestaña de administración se muestran cuatro opciones la primera es la pestaña de administración, la segunda el manual de funciones, la tercera boletas de pago y la ultima la planilla de sueldos y salarios.

FIGURA 13-3

La española: Modulo de Administración, 2014



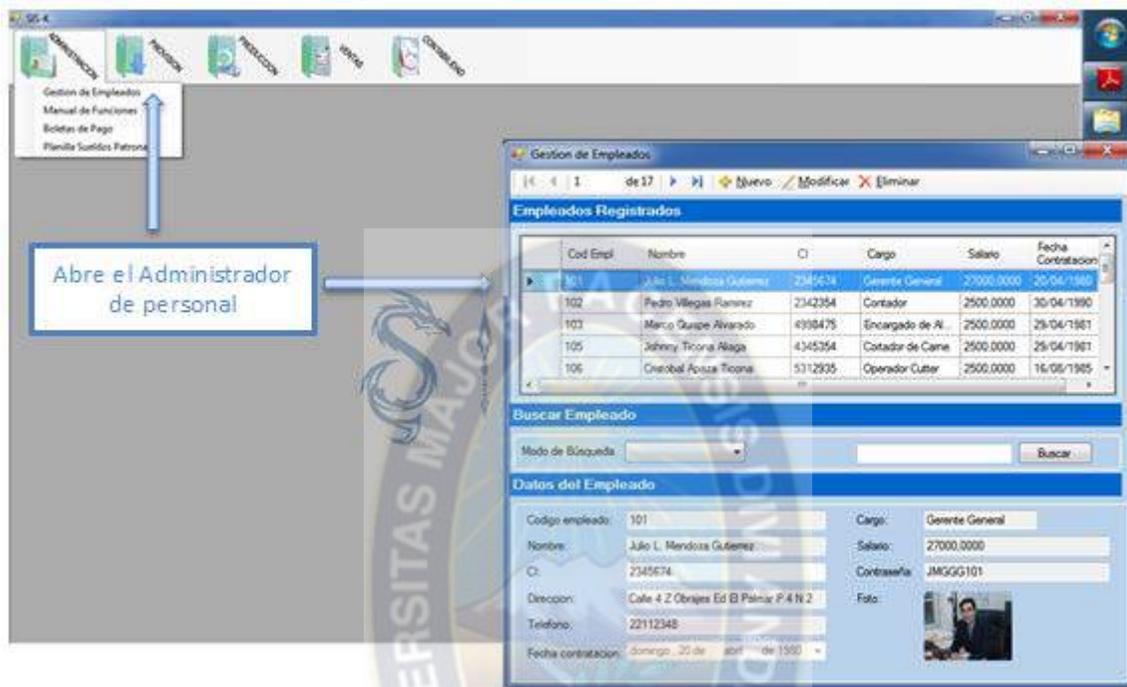
Fuente: Elaborado en base sistema central SIS-K

13.3.1 Gestión de empleados.

La forma de ingresar a la gestión de empleados es la siguiente:

FIGURA 13-4

La española: Modulo de Administración, 2014

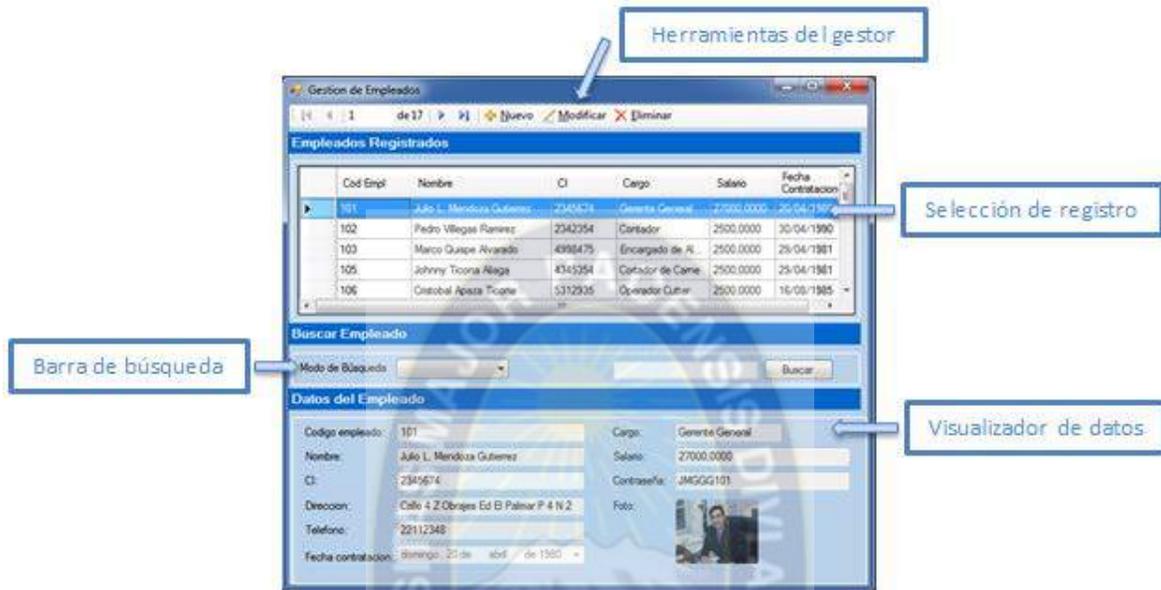


Fuente: Elaborado en base al módulo de administración

13.3.2 Entorno del gestor de empleados

El entorno del gestor de empleados se indica en la siguiente figura:

FIGURA 13-5
La española: Modulo de Administración, 2014



Fuente: Elaborado en base al módulo de administración

13.3.3 Crear nuevo registro

Los pasos para crear un nuevo registro es el siguiente:

FIGURA 13-6

La española: Módulo de Administración- Nuevo Registro, 2014



Fuente: Elaborado en base al módulo de administración

13.3.4 Modificar un registro

Para Modificar un registro se siguen los siguientes pasos:

- 1° Seleccione un registro
- 2° Clic en el botón modificar
- 3° En el visualizador de datos seleccione el campo que quiera modificar
- 4° Al finalizar las modificaciones dar clic en guardar

13.3.5 Eliminar un registro

Para eliminar un registro se siguen los siguientes pasos:

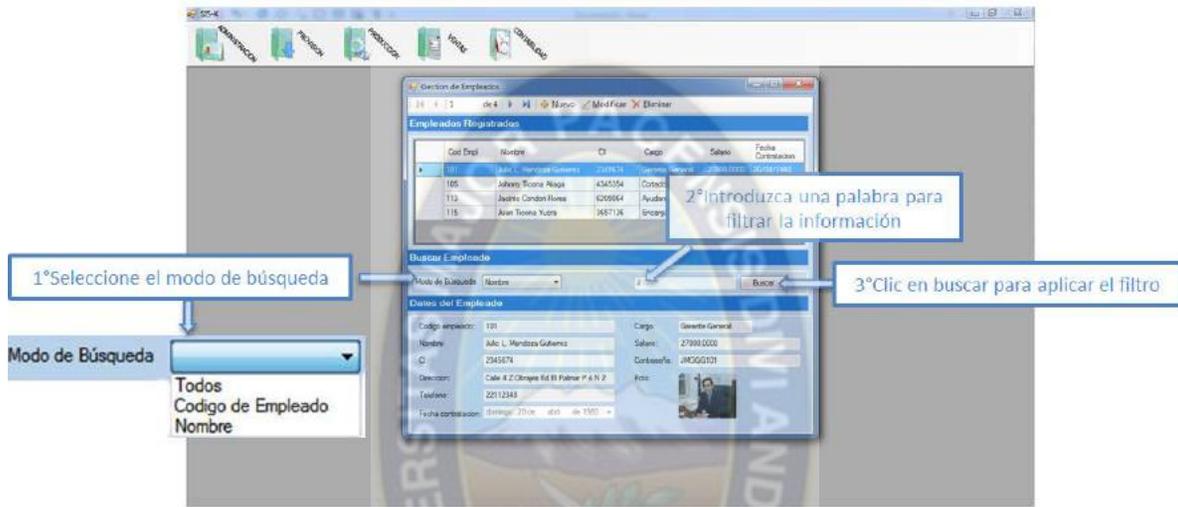
- 1° Seleccione un registro
- 2° Clic en el botón eliminar
- 3° Confirme la supresión del registro

13.3.6 Modo de búsqueda

Para realizar búsquedas en la base de datos se debe:

FIGURA 13-7

La española: Modulo de Administración-Búsqueda de registros, 2014



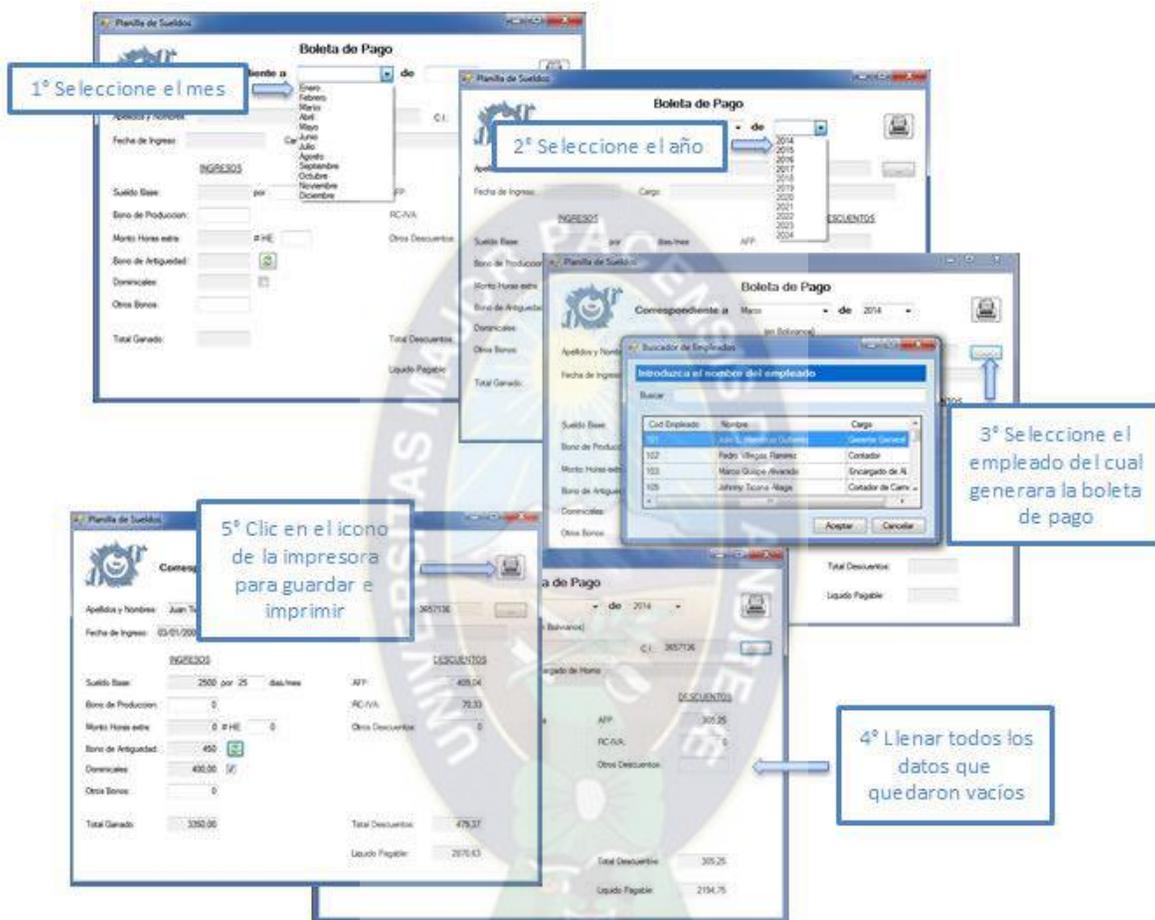
Fuente: Elaborado en base al módulo de administración

13.3.8 Boletas de pago

Para generar las boletas de pago se debe:

FIGURA 13-9

La española: Módulo de Administración-Boletas de Pago, 2014



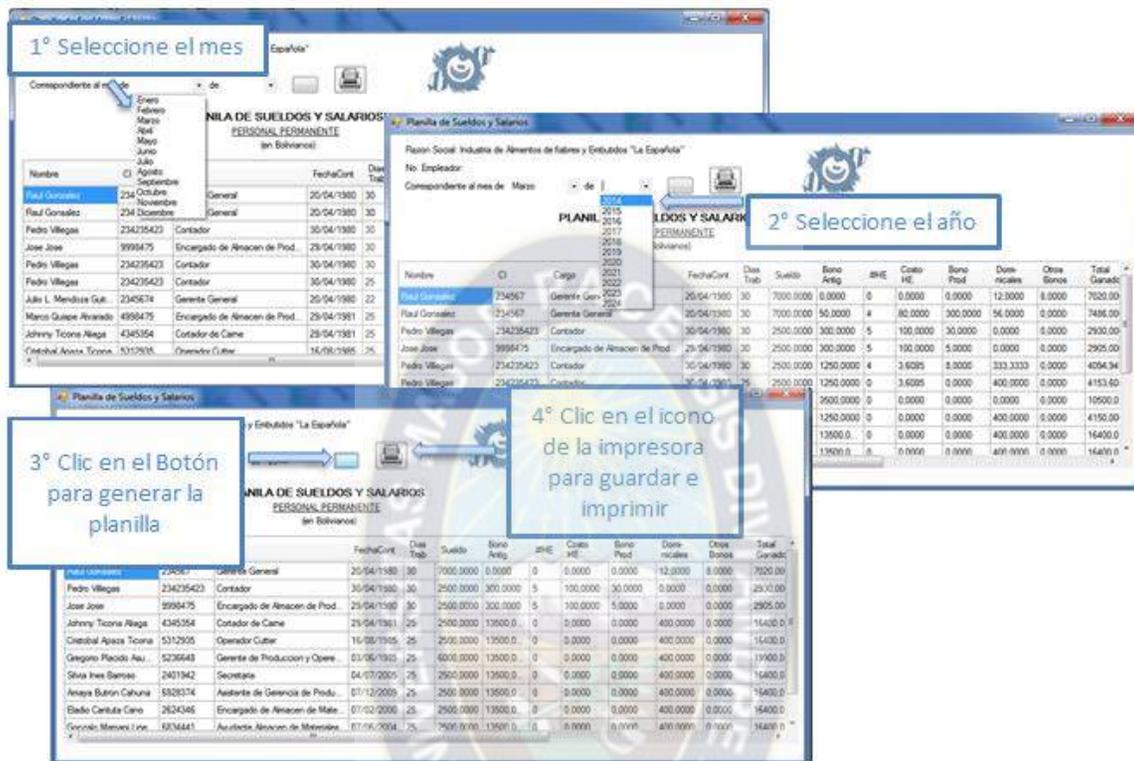
Fuente: Elaborado en base al módulo de administración

13.3.9 Planilla de sueldos personal

La planilla de sueldos personal se la genera como se muestra a continuación:

FIGURA 13-10

La española: Modulo de Administración-Planilla de Sueldos, 2014



Fuente: Elaborado en base al módulo de administración

13.4 Módulo de provisión

Al dar clic en la pestaña de provisión se muestran dos opciones la primera es la pestaña de gestión de proveedores y la segunda que es la de compras.

FIGURA 13-11

La española: Modulo de Provisión, 2014



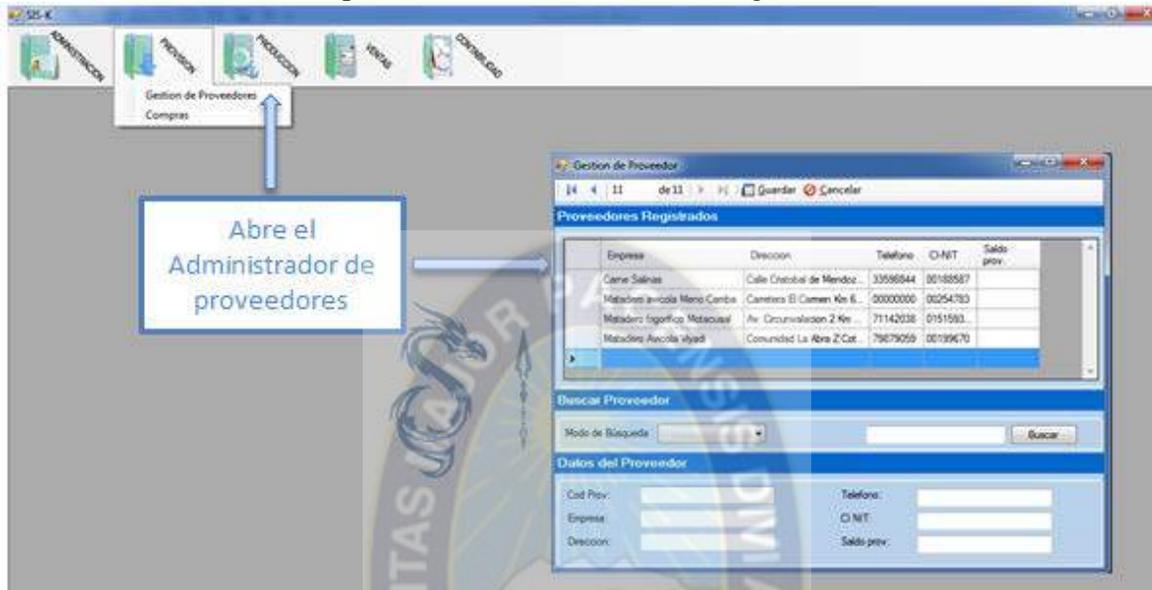
Fuente: Elaborado en base al módulo de provisión

13.4.1 Gestión de proveedores

La forma de ingreso a la gestión de proveedores es la siguiente:

FIGURA 13-12

La española: Modulo de Provisión-Ingreso, 2014



Fuente: Elaborado en base al módulo de provisión

13.4.2 Entorno del administrador de proveedores

El entorno del gestor de provisión se muestra a continuación.

FIGURA 13-13

La española: Modulo de Provisión-Entorno, 2014



Fuente: Elaborado en base al módulo de provisión

13.4.2.1 Crear nuevo registro

Para crear un registro nuevo se debe:

- 1° Clic en el botón nuevo
- 2° Llene los datos respectivos del visualizador de datos
- 3° Clic en el botón guardar

13.4.3 Modificar un registro

Para modificar un registro se debe:

- 1° Seleccione un registro
- 2° Clic en el botón modificar
- 3° En el visualizador de datos seleccione el campo que quiera modificar
- 4° Al finalizar las modificaciones dar clic en guardar

13.4.4 Eliminar un registro

Para eliminar un registro se debe:

- 1° Seleccione un registro
- 2° Clic en el botón eliminar
- 3° Confirme la supresión del registro

13.4.5 Modo de búsqueda

Al igual que en el módulo anterior se sigue la misma metodología para iniciar la búsqueda teniendo como filtros:

FIGURA 13-14

La española: Modulo de Provisión-Filtros de Búsqueda, 2014



Fuente: Elaborado en base al módulo de provisión

13.4.6 Comprobante de compra

Para generar el comprobante de compra se deben llenar los siguientes campos:

FIGURA 13-15

La española: Modulo de Provisión-Comprobantes de compra, 2014



Fuente: Elaborado en base al módulo de provisión

13.5 Módulo de producción

Al dar clic en la pestaña de producción se muestran tres opciones las primeras dos son las pestaña de gestión de inventarios de productos terminados e insumos y la tercera es la del programa maestro de producción.

FIGURA 13-16

La española: Modulo de Producción, 2014



Fuente: Elaborado en base al módulo de producción

13.5.1 Entorno de administrador de productos terminados

El entorno del administrador de productos terminados es el siguiente:

FIGURA 13-17

La española: Modulo de Producción-Entorno, 2014



Fuente: Elaborado en base al módulo de producción

13.5.2 Crear nuevo registro

Para crear un nuevo registro se debe:

- 1° Clic en el botón nuevo
- 2° Llene los datos respectivos del visualizador de datos
- 3° Clic en el botón guardar

13.5.3 Modificar un registro

Para modificar un registro se debe:

- 1° Seleccione un registro
- 2° Clic en el botón modificar
- 3° En el visualizador de datos seleccione el campo que quiera modificar
- 4° Al finalizar las modificaciones dar clic en guardar

13.5.4 Eliminar un registro

Para eliminar un registro se debe:

- 1° Seleccione un registro
- 2° Clic en el botón eliminar
- 3° Confirme la supresión del registro

13.5.5 Modo de búsqueda

Al igual que en el módulo anterior se sigue la misma metodología para iniciar la búsqueda teniendo como filtros:

FIGURA 13-18

La española: Módulo de Producción-Filtros de búsqueda, 2014



Fuente: Elaborado en base al módulo de producción

13.5.6 Entorno administrador de insumos

El entorno del administrador de insumos es el siguiente:

FIGURA 13-19

La española: Módulo de Producción, 2014



Fuente: Elaborado en base al módulo de producción

13.5.7 Crear nuevo registro

Para crear un nuevo registro se debe:

- 1° Clic en el botón nuevo
- 2° Llene los datos respectivos del visualizador de datos
- 3° Clic en el botón guardar

13.5.8 Modificar un registro

Para modificar un registro se debe:

- 1° Seleccione un registro
- 2° Clic en el botón modificar

3° En el visualizador de datos seleccione el campo que quiera modificar

4° Al finalizar las modificaciones dar clic en guardar

13.5.9 Eliminar un registro

Para eliminar un registro se debe:

1° Seleccione un registro

2° Clic en el botón eliminar

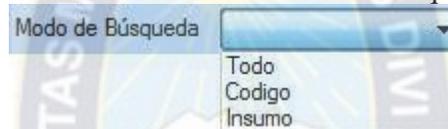
3° Confirme la supresión del registro

13.5.10 Modo de búsqueda

Al igual que en el módulo anterior se sigue la misma metodología para iniciar la búsqueda teniendo como filtros:

FIGURA 13-20

La española: Modulo de Producción-Filtros de búsqueda Insumos, 2014



Fuente: Elaborado en base al módulo de producción

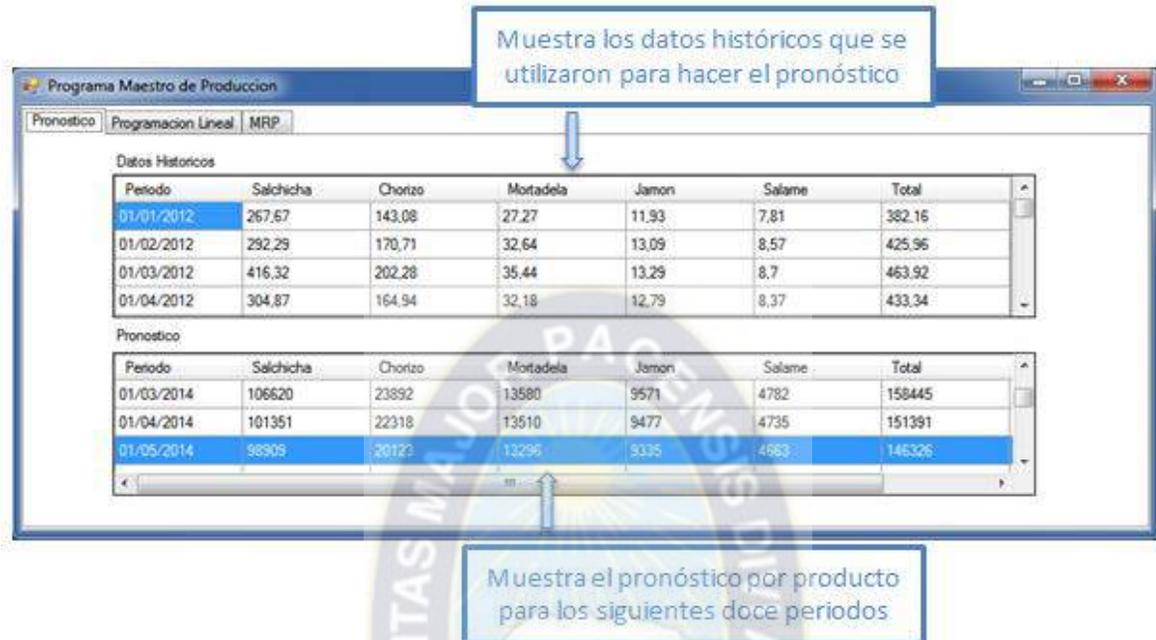
13.5.11 Programa maestro de producción

El programa maestro de producción está compuesto por tres pestañas, pronostico, programación lineal y MRP.

PRONOSTICO

FIGURA 13-21

La española: Modulo de Producción-Pronostico, 2014



Fuente: Elaborado en base al módulo de producción

PROGRAMACION LINEAL

FIGURA 13-22

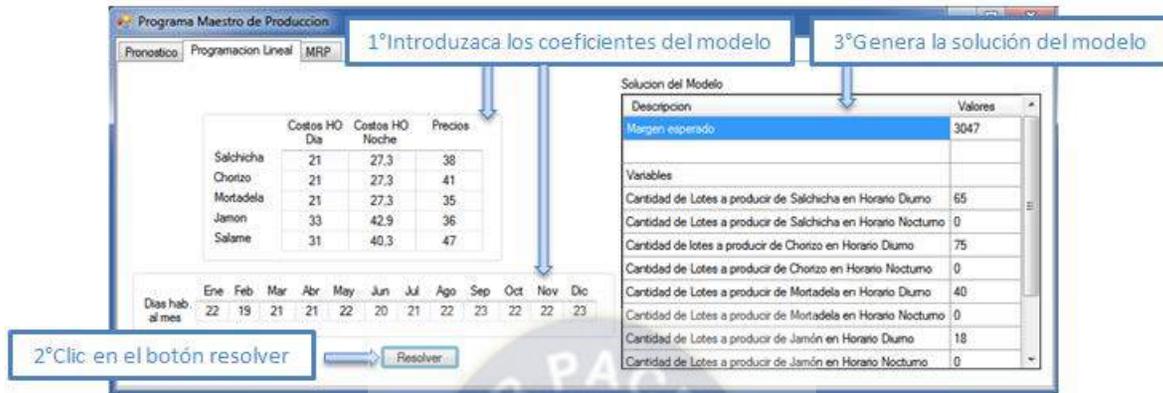
La española: Modulo de Producción-Entorno Programación Lineal, 2014



Fuente: Elaborado en base al módulo de producción

FIGURA 13-23

La española: Modulo de Producción-Modo de uso, 2014



Fuente: Elaborado en base al módulo de producción

MRP

Una vez realizado la programación lineal se pasa a la pestaña MRP que ya genero un programa de producción.

FIGURA 13-24

La española: Modulo de Producción-MRP, 2014

Descripcion	Enero	Febrero	Marzo	Abril	Mayo	Junio	Total
Dias laborales por mes	22	19	21	21	22	20	125
Horas de produccion disponibles	1760	1520	1680	1680	1760	1600	10000
Produccion real	3385	2923	3231	3231	3385	3077	19232
Demanda pronosticada	2857	2766	3169	3028	2927	4341	19088
Inventario Final	528	685	747	950	1408	144	
Existencia de reserva	0	0	0	0	0	0	
Unidades sobrantes	528	685	747	950	1408	144	
Costo de faltantes	0	0	0	0	0	0	0
Costo de inventario	5280	6854	7473	9502	14083	1438	44630
Costo de tiempo comido	22000	19000	21000	21000	22000	20000	125000
Costo Total							169630

Fuente: Elaborado en base al módulo de producción

13.6 Módulo de ventas

Al dar clic en la pestaña de ventas se muestran dos opciones la primera muestra un administrador de clientes y la segunda un módulo para realizar las ventas.

FIGURA 13-25

La española: Modulo de Ventas, 2014



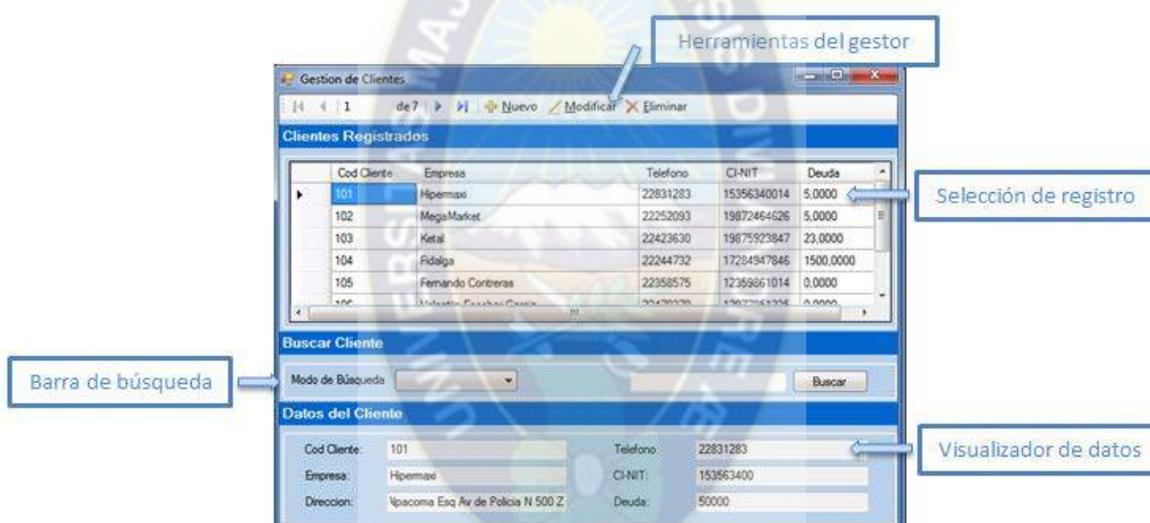
Fuente: Elaborado en base al módulo de ventas

13.6.1 Entorno gestión de clientes

El entorno del administrador de clientes es el siguiente:

FIGURA 13-26

La española: Modulo de Ventas-Entorno, 2014



Fuente: Elaborado en base al módulo de ventas

13.6.2 Crear nuevo registro

- 1° Clic en el botón nuevo
- 2° Llene los datos respectivos del visualizador de datos
- 3° Clic en el botón guardar

13.6.3 Modificar un registro

- 1° Seleccione un registro
- 2° Clic en el botón modificar
- 3° En el visualizador de datos seleccione el campo que quiera modificar
- 4° Al finalizar las modificaciones dar clic en guardar

13.6.4 Eliminar un registro

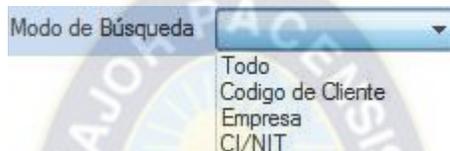
- 1° Seleccione un registro
- 2° Clic en el botón eliminar
- 3° Confirme la supresión del registro

13.6.5 Modo de búsqueda

Al igual que en el módulo anterior se sigue la misma metodología para iniciar la búsqueda teniendo como filtros:

FIGURA 13-27

La española: Modulo de Ventas-Filtros de búsqueda, 2014



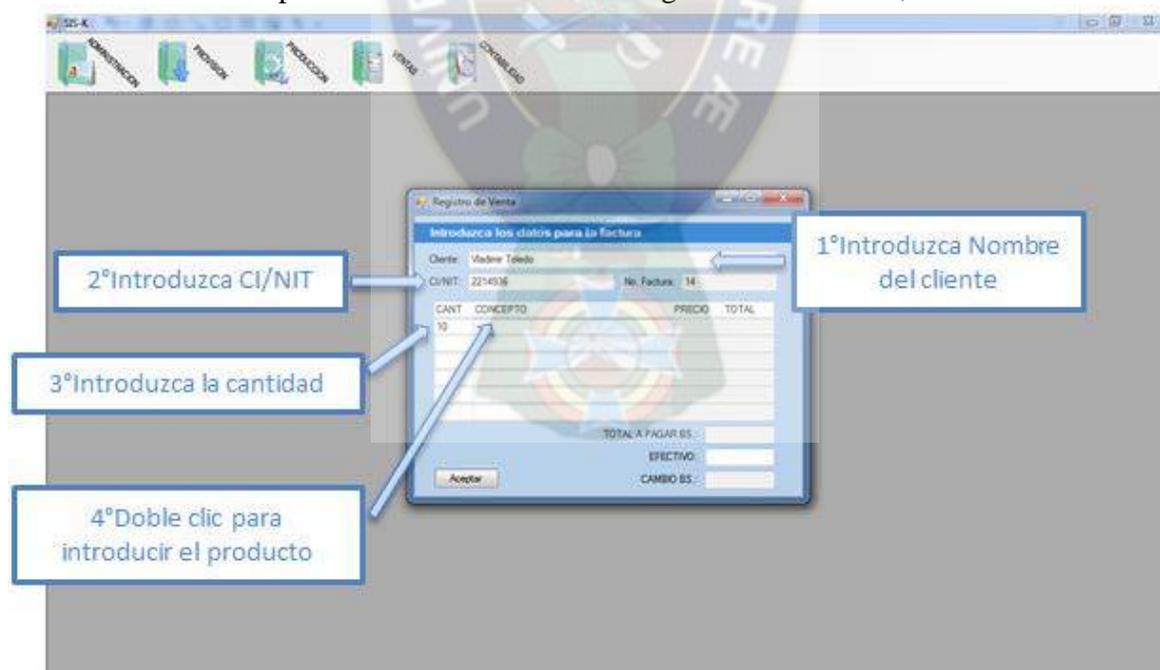
Fuente: Elaborado en base al módulo de ventas

13.6.6 Registro de ventas

Para realizar algún registro de venta se debe:

FIGURA 13-28

La española: Modulo de Ventas-Registro de venta 1-3, 2014

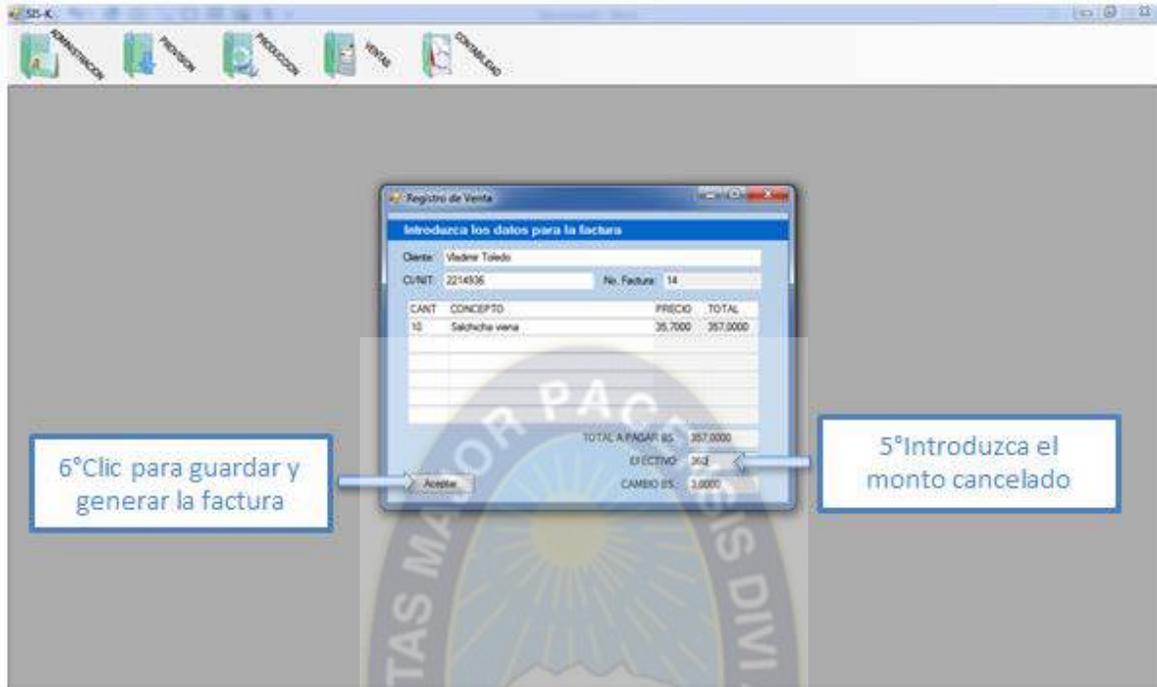


Fuente: Elaborado en base al módulo de ventas

Repita los dos últimos pasos si se registra más de un producto.

FIGURA 13-29

La española: Modulo de Ventas-Registro de venta 2-3, 2014



Fuente: Elaborado en base al módulo de ventas

Finalmente se imprime la factura:

FIGURA 13-30

La española: Modulo de Ventas-Registro de venta 3-3, 2014

FACTURA
ORIGINAL

NIT: 2703795012
FACTURA NO.: 13
AUTORIZACION NRO.: 390400921080

FECHA: 27/05/2014 10:59:00
SEÑOR (ES): Hipermaxi
NIT/CI: 15356340014

CANT	CONCEPTO	PRECIO	TOTAL
100	Salchicha viena	35,7000	3570,0000
200	Chorizo	40,0000	8000,0000
100	Mortadela	41,0000	4100,0000

TOTAL A PAGAR BS.: 15670,0000
EFECTIVO: 15700
CAMBIO BS.: 30,0000
SON: 00/100 BS.

7°Clic para imprimir factura

"La reproducción total o parcial y/o el uso no autorizado de esta Nota Fiscal constituye un delito a ser sancionado conforme a Ley"

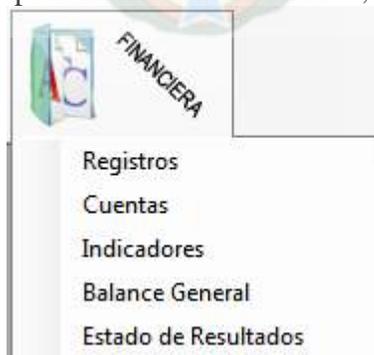
Fuente: Elaborado en base al módulo de ventas

13.7 Modulo financiero

Al dar clic en la pestaña de ventas se muestran cinco opciones la primera realiza los registros contables de la empresa, la segunda administra las cuentas, la tercera muestra los indicadores financieros de la empresa, la cuarta genera un balance general y la quinta un estado de resultados.

FIGURA 13-31

La española: Modulo Financiero, 2014



Fuente: Elaborado en base al módulo de ventas

13.7.1 Registros contables

Para realizar los registros contables se debe seguir los siguientes pasos:

FIGURA 13-32

La española: Modulo Financiero-Registros Contables, 2014



Fuente: Elaborado en base al módulo financiero

13.7.2 Entorno del administrador de cuentas

El entorno del administrador de cuentas es el siguiente:

FIGURA 13-33

La española: Modulo Financiero-Entorno Administrador de cuentas, 2014



Fuente: Elaborado en base al módulo financiero

13.7.3 Crear nuevo registro

Para crear un nuevo registro se debe:

- 1° Clic en el botón nuevo
- 2° Llene los datos respectivos del visualizador de datos
- 3° Clic en el botón guardar

13.7.4 Modificar un registro

Para modificar un registro se debe:

- 1° Seleccione un registro
- 2° Clic en el botón modificar
- 3° En el visualizador de datos seleccione el campo que quiera modificar
- 4° Al finalizar las modificaciones dar clic en guardar

13.7.5 Eliminar un registro

Para eliminar un registro se debe:

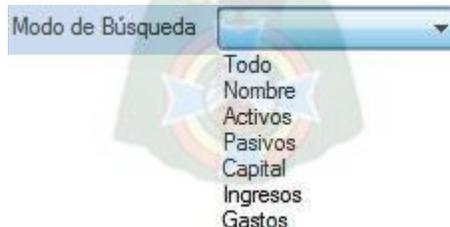
- 1° Seleccione un registro
- 2° Clic en el botón eliminar
- 3° Confirme la supresión del registro

13.7.6 Modo de búsqueda

Al igual que en el módulo anterior se sigue la misma metodología para iniciar la búsqueda teniendo como filtros:

FIGURA 13-34

La española: Modulo Financiero-Filtros de Búsqueda, 2014



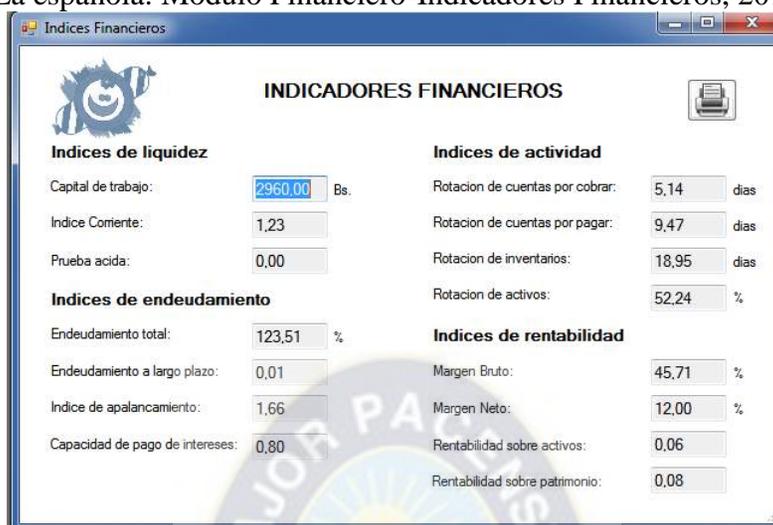
Fuente: Elaborado en base al módulo financiero

13.7.7 Indicadores financieros

Al dar clic en la pestaña de indicadores se generan los indicadores financieros de la empresa solo se tiene la opción de imprimir debido a que los demás proceso se los realiza de forma automática por el programa.

FIGURA 13-35

La española: Modulo Financiero-Indicadores Financieros, 2014



Fuente: Elaborado en base al módulo financiero

13.7.8 Balance general

Al dar clic en la pestaña de balance general se genera un balance del mes actual de la empresa solo se tiene la opción de imprimir debido a que los demás proceso se los realiza de forma automática por el programa.

FIGURA 13-36

La española: Modulo Financiero-Balance general, 2014

ACTIVO		PASIVO	
ACTIVO CORRIENTE		PASIVO CORRIENTE	
Caja	1300,00	Cuentas por Pagar	10,00
Caja chica	0,00	Proveedores por Pagar	0,00
Banco Mercantil	0,00	Sueldos y salarios por pagar	0,00
Cuentas por Cobrar	10,00	Impuesto IVA debito fiscal	290,00
Impuesto IVA credito fiscal	0,00	AFP Futuro de Bolivia	10,00
IUE por Compensar	0,00	AFP BBV Prevision	20,00
	1310,00		330,00
INVENTARIOS		PASIVO NO CORRIENTE	
Inventario de prod. tem.	20,00	Prestamo Hipotecario	0,00
Inventario de Insumos	0,00	Prestamo documentario	10,00
	20,00	Prevision beneficios sociales	0,00
		Provision aguinaldos	0,00
ACTIVO NO CORRIENTE			10,00
ACTIVO FIJO		CAPITAL	
Terrenos	10,00	Capital	1000,00
Edificios y construcciones	0,00	Aportes por capitalizar	0,00
Dep. Acum de edif y cons.	0,00	Reserva legal	0,00
Muebles y enseres	0,00		
Dep. Acum de muebl y ens	0,00		
Maquinaria y equipo	0,00		
Dep. Acum de Maq. y eq.	0,00		
Vehiculos	0,00		
Dep. Acum. de Vehiculos	0,00		
Herramientas	0,00		
Dep. Acum. de Herramientas	0,00		
Equipos de computacion	0,00		
Dep. Acum. de Eq. de Comp.	0,00		
	10,00		
ACTIVO DIFERIDO			
Gastos pagados x adelant.	0,00		
	0,00		
TOTAL ACTIVO	1340,00	TOTAL PASIVO Y CAPITAL	1340,00

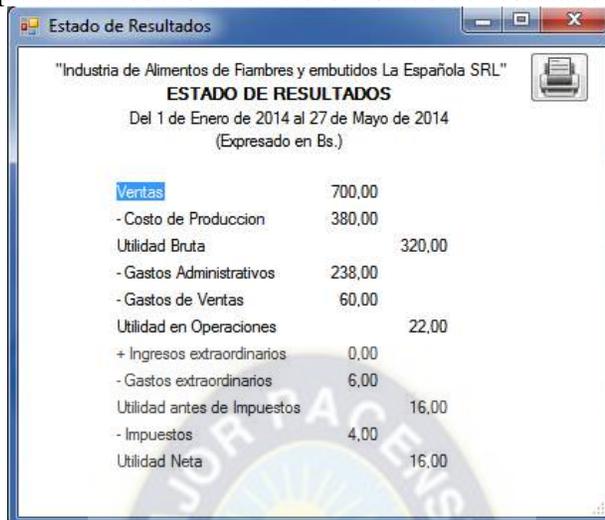
Fuente: Elaborado en base al módulo financiero

13.7.9 Estado de resultados

Al dar clic en la pestaña estados financieros se genera un estado de resultados del mes actual de la empresa solo se tiene la opción de imprimir debido a que los demás proceso se los realiza de forma automática por el programa.

FIGURA 13-37

La española: Modulo Financiero-Estado de resultados, 2014



"Industria de Alimentos de Fiambres y embutidos La Española SRL"		
ESTADO DE RESULTADOS		
Del 1 de Enero de 2014 al 27 de Mayo de 2014		
(Expresado en Bs.)		
Ventas	700,00	
- Costo de Produccion	380,00	
Utilidad Bruta		320,00
- Gastos Administrativos	238,00	
- Gastos de Ventas	60,00	
Utilidad en Operaciones		22,00
+ Ingresos extraordinarios	0,00	
- Gastos extraordinarios	6,00	
Utilidad antes de Impuestos		16,00
- Impuestos	4,00	
Utilidad Neta		16,00

Fuente: Elaborado en base al módulo financiero



CAPÍTULO 14. EVALUACIÓN ECONÓMICA Y FINANCIERA

14.1 Definición

Es un análisis que se hace al costo de oportunidad mediante un indicador financiero para tomar una decisión acerca del proyecto.

14.1.1 Bases para la evaluación financiera

Costo de oportunidad, es la relación de un beneficio social y un beneficio económico.

Indicador de rentabilidad, es el indicador que mide la oportunidad a largo plazo sobre la cual se toma la decisión.

14.1.2 Fases para la evaluación financiera



Fuente: Elaborado en base a (Castro Ordoñez 2010)

14.2 Beneficios del sistema

A continuación vamos a especificar una lista de beneficios que tendría la empresa si implementaría el sistema:

- Acceso rápido a la información
- Menor redundancia de la información
- Mayor seguridad en los datos
- Se tienen datos más documentados

- Toma de decisión más eficiente
- Menor espacio de almacenamiento
- Mayor control en almacenes
- Evita pérdida de tiempo en la recopilación de la información
- Menor error en la planeación de la producción
- Menor error en el uso de materiales
- Proporciona, facilita y ejecuta automáticamente procesos que constantemente se realizaban manualmente
- Mayor capacidad de análisis con los nuevos reportes
- Acceso a indicadores financiero de manera inmediata

14.3 Costos de oportunidad

14.3.1 Acceso rápido a la información

Se ha reducido el tiempo en la búsqueda de la información en un 80%. El cual tiene un costo según gerencia de: 80 [Bs/mes]. Entonces el beneficio es de: 768 [Bs/año].

14.3.2 Menor redundancia de la información

Este beneficio va ligado al anterior costo de oportunidad. Por lo tanto comparten el mismo costo de oportunidad.

14.3.3 Mayor seguridad en los datos

Anteriormente se tenía una vulnerabilidad en la información, el cual según gerencia tenía un costo de: 300 [Bs/mes] por pérdida de clientes, 80 [Bs/mes] en el agotamiento de insumos de los proveedores, estos costos se anulan por el sistema implementado ya que cuenta con un sistema de seguridad. Entonces el beneficio total es de 4560 [Bs/año].

14.3.4 Se tienen datos más documentados

Anteriormente no se podían contactar a clientes y proveedores antiguos, porque no se guardaba la información, el cual según gerencia tenía un costo de buscar nuevamente cliente o proveedor de: 500 [Bs/año] para los dos casos, esto se anula por el sistema que cuenta con una base de datos. Entonces el beneficio es de 500 [Bs/año].

14.3.5 Toma de decisión más eficiente

Anteriormente se tenía un deficiente toma de decisiones debido a que no se tenía acceso rápido a la información, ni tampoco se contaba con información confiable, los errores que

se cometían por estas causas son: entrega a destiempo por mala programación de la producción en 4000 [Bs/año], vencimiento de inventario por mal pronóstico en 50 [Kg/año] y el costo de mantener 50 [Kg] es de 2500 [Bs/año], estos costos son anulados por el sistema. Entonces el beneficio es de 6500 [Bs/año].

14.3.6 Menor espacio de almacenamiento

Anteriormente se tenían ambiente lleno de repisas en los cuales se almacenaban las carpetas con datos de la empresa. Ahora ese espacio es usado para una cafetería la cual mejora la calidad de trabajo, esto mejora el rendimiento del personal en: 5%, entonces se tiene un costo de 15 [Bs/mes - trabajador]. Entonces el beneficio es de: 1800 [Bs/año].

14.3.7 Mayor control en almacenes

Anteriormente se tenía una pérdida de insumos y productos frecuente el cual causaba una pérdida de: 2500 [Bs/año], el sistema aporta inicialmente este problema en un 50% y a partir del segundo año en un 100%. Entonces el beneficio para el primer año es de: 1250 [Bs/año], para el segundo año es de: 2500 [Bs/año].

14.3.8 Evita pérdida de tiempo en la recopilación de la información

Se ha reducido el tiempo en la recopilación de la información en un 90%. El cual tiene un costo según gerencia de: 100 [Bs/mes]. Entonces el beneficio es de: 1620 [Bs/año].

14.3.9 Menor error en la planeación de la producción

Este beneficio va ligado a la toma de decisiones más eficiente. Por lo tanto comparten el mismo costo de oportunidad.

14.3.10 Proporciona, facilita y ejecuta automáticamente procesos

Estos procesos se los realizaba manualmente. Este beneficio va ligado a evitar pérdida de tiempo en la recopilación de la información. Por lo tanto comparten el mismo costo de oportunidad.

14.3.11 Mayor capacidad de análisis con los nuevos reportes

Este beneficio va ligado a la toma de decisiones más eficiente, debida a que los reportes que genera el sistema son una herramienta fundamental en la toma de decisiones. Por lo tanto comparten el mismo costo de oportunidad.

14.4 Beneficio total

El beneficio total para el horizonte del sistema es de:

CUADRO 14-1

La española. Beneficios del sistema [Bolivianos], 2014

Año	2015	2016	2017	2018	2019
Beneficios	16998	18248	18248	18248	18248

Fuente: Elaborado en base a los costos de oportunidad anteriormente analizados

14.5 Costo de implementar el sistema

Para determinar los costos de elaboración del programa o herramienta informática, se lo realiza a través de las horas utilizadas en la programación desde el diseño hasta la elaboración del código fuente y su empaquetado. Otra forma para determinar el costo es por Megabytes elaborados. Y la tercera forma sería un contrato por todo el proyecto. Para el proyecto nosotros elegimos el contrato por proyecto. Un proyecto de este tipo esta cotizado alrededor de: 8000 [bolivianos]. Los costos que la empresa debe incurrir se detallan a continuación:

CUADRO 14-2

La española. Cotización de la instalación [Bolivianos], 2014

Descripción	Cantidad	Precio	Total
Computadoras	5	2590	12950
Enrutador	1	840	840
Instalación de la red	1	2500	2500
Instalación del sistema	1	8000	8000
Total			24290

Fuente: Elaborado en base a la cotización en el mercado popular Eloy Salmon

14.6 Flujo de fondos

14.6.1 Tasa de oportunidad

El dinero de inversión pudo ser invertido de la siguiente manera según el gerente general:

CUADRO 14-3

La española. Tasa de oportunidad de la inversión, 2014

Producción	Bonos	Total
14290	10000	24290
12,00%	6,50%	
1714,8	650	2364,8
		9,74%

Fuente: Elaborado en base a datos de gerencia general

El horizonte de vida según la programación es de 5 años.

14.6.2 Indicador de rentabilidad

El flujo de fondos para el horizonte de vida del sistema es de:

CUADRO 14-4

La española. Flujo de fondos [Bolivianos], 2014

Año	2014	2015	2016	2017	2018	2019
Flujo de fondos	-24290	16998	18248	18248	18248	18248

Fuente: Elaborado en base a los beneficios y la inversión del sistema

$$VAN = -INVERSION + \sum \frac{BENEFICIOS}{(1+i)^n}$$

$$VAN = -24290 + \frac{16998}{(1+0,0974)} + \frac{16998}{(1+0,0974)^2} + \frac{16998}{(1+0,0974)^3} + \frac{16998}{(1+0,0974)^4} + \frac{16998}{(1+0,0974)^5}$$

$$VAN = 40.292,08$$

Con estos resultados podemos asegurar la implementación del sistema.



Conclusiones

Al finalizar el presente proyecto de grado se llegaron a las siguientes conclusiones:

Se logró crear un sistema informático que optimiza la gestión de la producción de la empresa, para ellos utiliza diversos módulos que ayudan a sus usuarios a realizar sus tareas más rápida y fácilmente. Yendo desde el departamento administrativo, provisión, producción, ventas y contabilidad.

Para ello se creó con éxito la base de datos digital con la actualización de los datos, el cual es usado hoy por hoy en la empresa. Esto ayudo a que se tenga información confiable y oportuna, que era un requerimiento imprescindible de la gerencia general, ya que la información era vulnerable.

Se realizaron los estudios pertinentes para encontrar los parámetros de producción y comercialización, el cual es validado por la empresa. Esto ayudo a que se tenga una base para la programación de la producción en el sistema informático, ya que no se contaba con ningún parámetro para programar la producción.

Se desarrolló un modelo de programación lineal para programar la producción. Esto ayudo a maximizar las ganancias en función de la producción óptima de productos. Se transformó el modelo de programación lineal en un método informático para tener un acceso más rápido a este programa. Esto da una capacidad de cambiar la producción en función del cambio de los parámetros calculados, este era un requerimiento imprescindible de la gerencia general.

En cuanto al desarrollo de la aplicación informática en primer lugar se analizaron los requerimientos de los departamentos de la empresa para crear varios módulos que ayuden a cumplir estos requerimientos. El modulo administración ayuda a gestionar los recursos humanos, el módulo de provisión ayuda a gestionar las compras, el módulo de producción ayuda a gestionar el inventario y programar la producción, el módulo de ventas ayuda a gestionar los clientes y la facturación de productos, y el módulo de contabilidad ayuda a llevar registros y cuentas de la empresa y también genera los indicadores financieros, balance general y estado de resultados, necesarios para la toma de decisiones.

Los informes y las salidas producidas por el sistema son precisos, confiables y completos.

La función del Análisis puede dar soporte a las actividades de la empresa.

Se desarrolló un sistema de autenticación que protege toda la información de la empresa y otorga acceso diferenciado según el tipo de usuario.

Al ver los resultados de la evaluación financiera del proyecto se pudo observar que este llega a ser muy rentable debido a que se puede llegar a cubrir el costo de inversión del proyecto hasta el segundo año, obteniéndose un beneficio económico para la empresa considerable. Uno de los principales beneficios es la disminución del incumplimiento de los pedidos que varían entre un 25% y 50%.



Recomendaciones

Después de ver que el incremento en la cantidad de datos requeridos por las empresas y la interacción cada día más compleja entre personas, exige que este sistema se actualice cada dos años como mínimo, para actualizar los parámetros, mejorar los módulos, siempre buscando una mejora continua del sistema SIS-K. En la actualidad para la empresa, los sistemas de información basados en computadoras serán el corazón de las actividades cotidianas y objeto de gran consideración en la toma de decisiones, considerara con mucho cuidado las capacidades del sistema de información cuando decidan ingresar o no en nuevos mercados o cuando planeen la respuesta que darán a la competencia.

Al implementar el sistema informático puede traer consigo muchos beneficios para la empresa y a la vez complicaciones si este no es administrado de manera correcta, es decir que los usuarios deben estar capacitados, caso contrario el programa puede llegar a causar conflictos grandes, por lo que se recomienda capacitar al personal constantemente.

Lo que se busca con el programa además de tener un mejor control de la gestión de la producción es realizar un cambio en la cultura laboral, cambiando su forma de trabajo actual.

Al establecer los sistemas de información basados en computadoras se debe tener la certeza de que se logren dos objetivos principales: que sea un sistema correcto y que este correcto el sistema. Si el sistema satisface ambos objetivos será completamente útil para la gerencia u organización.

El proceso de gestión para la creación de un sistema informático que ayude a optimizar la gestión de la producción para la empresa, encierra un conjunto de actividades, una de las cuales es la estimación, que es una actividad importante que no debe llevarse a cabo de forma descuidada. Existen técnicas útiles para la estimación de costes de tiempo. Y dado que la estimación es la base de todas las demás actividades de planificación del proyecto y sirve como guía para una buena Ingeniería de Sistemas.

Si bien el sistema SIS-K responde a todos los requerimientos que la empresa nos solicitó, es necesario que se capacite al personal constantemente para tener un mayor aprovechamiento del sistema. También se debe realizar mantenimiento preventivo a los equipos para una prolongación de su vida útil.

En cuanto al sistema se tiene que considerar que no existe un sistema imperfecto sino uno mal utilizado.

Bibliografía

- Adams, E.E., y R.J. Ebert. *Administracion de la produccion y las operaciones*. 4° Ed. Colombia: Universidad de Missouri, 1994.
- Alvarez C, Alfredo, y Eduardo Huayta C. *Errores y medidas*. La Paz, 2007.
- Aranda de Duero. *Desarrollo de Aplicaciones Informaticas*. 2004-2005. http://www.manualesgratuitos.iespana.es/doc/intefaces_usuario.pdf (último acceso: 2014).
- Beer, et al. *Gestion de recursos humanos*. España: Ministerio del trabajo, 1989.
- Booch, Grady, James Rumbaugh, y Ivar Jacobson. *Lenguaje unificado de modelado*. 1° Ed. Madrid: Editorial: Addison-Wesley Iberoamericana, 1999.
- Braude, Erick J. *Ingenieria de software: Una perspectiva orientada a objetos*. 1° Ed. Mexico: Alfa&Omega, 2005.
- Buffa, E. S. «La adminstracion de sistemas productivos.» En *Adminstracion de operaciones*. Mexico: LUMUSA, 1981.
- Castro Ordoñez, Raul. «Preparacion y evaluacion de proyectos 1.» Apuntes, La Paz, 2010.
- Chase, R.B., y N.J. Aquilano. *Direccion de la administracion de la produccion y de las operaciones*. 6° Ed. Wimintong: Addinson-Wesley Iberoamericana, 1994.
- CHECKING. *Gestion de finanzas*. 2010. <http://educaciones.cubaeduca.cu/medias/pdf/2185.pdf> (último acceso: 2014).
- . *Gestion de proveedores*. 2005. http://itilv3.osiatis.es/disenio_servicios_TI/gestion_proveedores/introduccion_objetivos.php (último acceso: 2014).
- . *Metodologias de Desarrollo de Software*. 2004. http://www.infromatizate.net/metodologias_de_deasarrollo_de_software (último acceso: Octubre de 2013).
- . *Monitorizando la calidad de Software en Desarrollo*. 2008. <http://www.als-es.com> (último acceso: Enero de 2014).
- Chiavenato, Idalveto. «Part VIII Enfoque sistematico de la administración.» En *Introducción a la teoía general de la administiración*, de Idalveto Chiavenato, 367. Interamericana: McGrawHill, 2006.

- COMPANYS Pacual, Ramon. *Planificacion y programacion de la produccion*. Barcelona, España: Boixaren Marcombo, 1989.
- Fernandez, E. *Contabilidad comercial*. 9° Ed. La Paz: Gisbert&CIA, 1979.
- Gaither, Norman, y Greg Frazier. *Administracion de produccion y operaciones*. 8° Ed. Interamericana: Soluciones empresariales, 2008.
- Gitman, E. *Fundamentos de Administracion financiera*. 5° Ed. Vol. 1. Interamericana: EMPES, 2003.
- Gronroos, Cristian. *Marketing y gestion de los servicios*. 5° Ed. Madrid: Diaz de Santos, 1990.
- Gujarati, Damondar N., y Dawn C. Porter. «Autocorrelación.» En *Econometria*, 412-466. D.F.: McGrawHill, 2010.
- Hiller, F., y Lieberman G. *Introduccion a la investigacion de operaciones*. 3° Ed. Mexico: McGrawHill, 1991.
- Holt, y Winters. *Modelo de Holt-Winters*. s.f.
- Honorable Congreso Nacional de Bolivia. *Ley General del Trabajo N° 220*. 1942.
<http://www.fundempresa.org.bo/>. 2011. <http://www.fundempresa.org.bo/directorio/ver-mas.php?id=fd4284061106ce2ad39bf71299e859ed&seccion=&division=&clase=&rubro=la%20espa%F1ola&depto=00&page=&searchSW=1> (último acceso: 2014).
- Hunca Aliaga, H. B. *Elaboracion de Metricas para elaborar la seguridad del Software durante su desarrollo*. La Paz, Bolivia: Tesis de Grado Universidad Mayor de San Andres, 2011.
- Hurtado Carmona, Douglas. *Teoria General de Sistemas*. 2° Edicion. Interamerica: Lulu, 2011.
- Impuestos Nacionales de Bolivia. «Capitulo 2 Aspectos tecnicos de las facturas o notas fiscales.» En *Nuevo sistema de facturacion (NSF-07)*, 6-9. La Paz, 2007.
- Kinnear, Tomas C, y James R Taylor. *Investigacion de mercados*. Quinta. Mexico: McGrawHill, 1998.
- Kort, H., H. Silverschatz, y S. Sudarshan. *Fundamentos de Base de Datos*. 3° Ed. España: McGrawHill, 1998.
- Kotler, Paul. *Direccion de marketing*. Interamericana: Prentice Hall, 1995.

- Larman, C. «UML y Patrones.» En *Introduccion al analisis y diseño orientado a objetos*. Mexico: Pretince Hall, 1999.
- Levaggi, G. *Teoría General de Sistemas*. Buenos Aires: UGERMAN, 1999.
- MARTINEZ EDGAR UML. *Lenguaje modelado unificado*. 2003.
<http://www.usuarios.lycos.es/oopere/uml.htm> (último acceso: Abril de 2014).
- Mendoza Gutierrez, Julio L., entrevista de Victor Joaquin Vargas Arce y Dennis Marvin Blanco Aguilar. *Tesis* (2014).
- Microsoft Visual Studio. *Visual Studio 2012 Product Guide*. 2012.
- Niebel, Ben, y Andris Freidvalds. *Ingenieria Industrial Metodos, Estandares y Diseño del trabajo*. 11va. Mexico: Alfaomega, 2004.
- OBJECT MANAGMENT GROUP. *Model Driven Architecture*. 2006.
<http://www.gidis.ing.unlpam.edu.ar/download/pdfs/IntroduccionUML.PDFmodelado-de-sistemas.pdf> (último acceso: Abril de 2014).
- Organizacion Internacional del Trabajo. *Introduccion al Estudio del Trabajo*. Cuarta. Ginebra: Oficina Internacional del Trabajo, 1996.
- Ortega Castro, Alfonso. *Introduccion a las finanzas*. Interamericana: McGrawHill, 2010.
- Pando Viamont, Grover. «Gestion de la produccion y operaciones.» Apuntes, La Paz, 2013.
- Platini Velthuis, M., y F. J. Garcia Rubio. *Calidad en el desarrollo y mantenimiento del Software*. Interamericana, 2003.
- Pressman, Roger. *Ingenieria de Software: Un enfoque practico*. 5° Ed. McGrawHill, 2002.
- Stevenson, Richard A. *Fundamentos de finanzas*. Interamericana: McGrawHill, 2010.
- Taha, H. *Investigacion de operaciones*. 6° Ed. Mexico: Pratince Hall, 1998.
- Vargas Arce, Victor Joaquin, y Dennis Marvin Blanco Aguilar. *SIS-K*. La Paz, Abril de 2014.
- Video2Brain. *SQL Server 2012*. Interamericana, 2013.
- Wayne L. , Winston. *Operations Research Applications and Algoritms*. Cuarta. Mexico: McGrawHill, 2003.
- WIKIPEDIA. *Wikipedia*. 2009. <http://www.wikipedia.org> (último acceso: 2014).

ANEXO





**ANEXO A:
NUMEROS
ALEATORIOS**

ANEXO 0-1: NUMEROS ALEATORIOS PARA LA SECCION DE MOLIDO

0,0404728	0,4611554	0,8440455	0,5991107	0,5399186	0,8299811	0,0931353	0,5806648
0,7453650	0,5184291	0,7866944	0,5949083	0,3243834	0,2435212	0,7490368	0,6325091
0,6285820	0,6604065	0,5564940	0,5089705	0,5897136	0,1709871	0,7356815	0,2734855
0,2191780	0,2575165	0,3338697	0,6744556	0,8118295	0,5130213	0,4212365	0,8847402
0,5712521	0,5594406	0,9962657	0,8942346	0,9137146	0,7833300	0,4033003	0,3438450
0,5912659	0,0984388	0,9725292	0,8977560	0,1674879	0,1529582	0,9481881	0,1263032
0,5726402	0,3889946	0,6571397	0,5592569	0,8497753	0,2376047	0,4933684	0,8942182
0,8413782	0,2468125	0,9483750	0,5590303	0,5461173	0,1875029	0,8311269	0,1307862
0,1776858	0,7265947	0,8339595	0,2907664	0,9972804	0,2686980	0,9756298	0,1908761
0,7583143	0,9062176	0,5091754	0,9189908	0,9999666	0,9945636	0,2798158	0,0226833
0,8691151	0,3640496	0,4391473	0,4354078	0,9384694	0,4257054	0,8140706	0,4201380
0,9344893	0,3629133	0,6677682	0,8696933	0,9889144	0,2484479	0,9613524	0,5790560
0,0006477	0,1773655	0,3234503	0,6253842	0,6804931	0,2467787	0,2220166	0,2652055
0,3467854	0,6318569	0,9939477	0,0547908	0,1550963	0,2806485	0,2735497	0,8575598
0,9524525	0,2205208	0,2316101	0,1762943	0,7536734	0,4014048	0,8810761	0,0871161
0,2758260	0,0773578	0,8143781	0,4595097	0,6152583	0,6063034	0,3022819	0,0178960
0,8403767	0,6151226	0,5256288	0,9577664	0,3566193	0,2211322	0,9149204	0,4162718
0,6618641	0,6227945	0,5771301	0,2435094	0,3807142	0,9557935	0,1495105	0,9188102
0,4432074	0,2540177	0,2384872	0,4985546	0,3667583	0,1673344	0,6563946	0,4907501
0,4626632	0,5789022	0,3039646	0,6258158	0,9851412	0,5077377	0,2513726	0,5743362
0,7767955	0,9678342	0,1500882	0,3768105	0,7295449	0,2922011	0,1409855	0,6340944
0,9036678	0,7979300	0,1530483	0,7648058	0,9325402	0,1508868	0,4769682	0,6760430
0,9525685	0,7726934	0,7373397	0,9208929	0,7543607	0,6543597	0,4650765	0,0672938
0,4662375	0,6282505	0,0848140	0,9593662	0,6058386	0,0691129	0,0858757	0,9764913
0,7443619	0,4156234	0,9727910					

ANEXO 0-2: NUMEROS ALEATORIOS PARA LA SECCION DE MEZCLADO

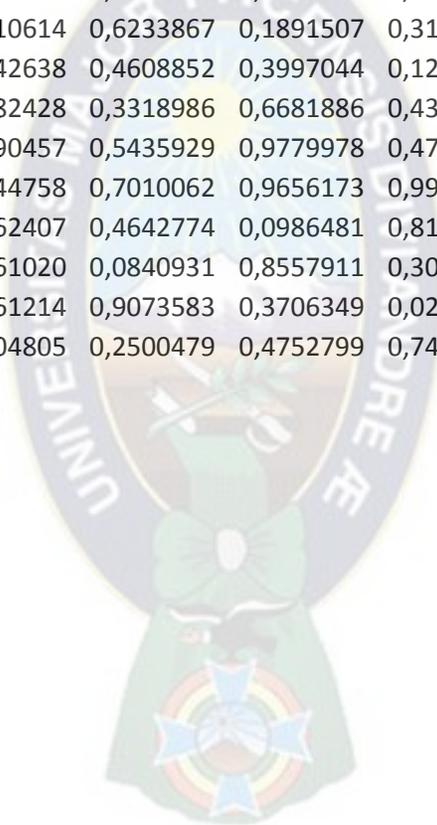
0,1498279	0,6068824	0,0960796	0,4670457	0,7644779	0,7079843	0,6947431	0,2295526
0,7618210	0,4040372	0,1146408	0,8575536	0,2229639	0,4367547	0,4333615	0,7355229
0,4458964	0,9242192	0,5288321	0,9722531	0,5075031	0,7635640	0,1762843	0,8016355
0,5758492	0,1231849	0,8084709	0,4199693	0,7468484	0,7484130	0,7799892	0,5256536
0,2319371	0,6994688	0,1790969	0,6513942	0,8323857	0,6397626	0,1765738	0,0444517
0,5229017	0,7342173	0,6504783	0,5725336	0,7631701	0,1834613	0,1772638	0,2681711
0,0230090	0,4469668	0,7764881	0,1066660	0,4638184	0,2284262	0,1801790	0,2498788
0,1549178	0,9363197	0,0211273	0,0299161	0,0092591	0,3553366	0,1782234	0,7611602
0,6582220	0,6078156	0,3047477	0,5848739	0,7633791	0,1802459	0,7243373	0,1624743
0,9952649	0,7655747	0,3154152	0,1527623	0,1473199	0,6631204	0,4324350	0,9342657
0,8417764	0,7890472	0,8583626	0,5856168	0,2132060	0,0464317	0,7327767	0,7427256
0,7038854	0,2534097	0,4756518	0,6944967	0,7235098	0,4230635	0,4153030	0,7679808
0,8476543	0,2175245	0,8372680	0,3708525	0,7869880	0,6110441	0,1681450	0,4428070
0,0468661	0,3496354	0,6891696	0,9910736	0,6915174	0,7652925	0,7343048	0,6145534
0,9009731	0,3492520	0,1025178	0,5214906	0,1388785	0,9701239	0,2975800	0,1589978
0,2055945	0,0358156	0,9912145	0,4252170	0,2920392	0,5226492	0,6159625	0,5145226
0,3739484	0,7930977	0,1658516	0,9413183	0,9819785	0,9333913	0,0310471	0,8925191
0,9937827	0,5964361	0,3888005	0,2380682	0,2406653	0,6767149	0,4692502	0,4682808
0,1497069	0,8327435	0,8835034	0,1902797	0,5364919	0,0879413	0,2464314	0,1371605
0,9872142	0,3203731	0,2817447	0,5894505	0,7571965	0,2740257	0,7740069	0,7917783
0,7081244	0,6118438	0,8652045	0,2774457	0,3458485	0,7846107	0,5380987	0,4642858
0,4882560	0,1853954	0,5170922	0,4451798	0,8449352	0,8810608	0,1994879	0,6739440
0,3416217	0,7689278	0,5219495	0,0592306	0,7577806	0,1876339	0,3395545	0,2871089
0,8687912	0,8020038	0,7990303	0,9497475	0,1330685	0,0956507	0,4149095	0,9193086
0,4502455	0,6035072	0,3714734	0,2270100	0,3718163	0,1941098	0,3534281	0,0060678
0,5850095	0,8607476	0,1108138	0,9257274	0,6808760	0,4628567	0,3245776	0,6230754
0,7857524	0,2245924	0,2544049	0,5051539	0,1235631	0,6222869	0,6717547	0,6440605
0,8241980	0,2008651	0,5879291	0,7671316	0,5326274	0,4705790	0,4431437	0,2529493
0,9927381	0,1671906	0,7227463	0,9734511	0,5793796	0,2638436	0,9154077	0,0204489
0,4570632	0,8852304	0,8811542	0,0138873	0,8594233	0,0807922	0,9824541	0,6489885
0,2549512	0,9951434	0,0077516	0,8061731	0,3116748	0,8764647	0,3180740	0,6449242
0,7113500	0,6443301	0,4515192	0,1087340	0,3927956	0,4741059	0,9894036	0,5681899
0,5321986	0,1958278	0,7179597	0,3972157	0,4709979	0,1178391	0,5273666	0,7129856
0,7887996	0,1784081	0,4479570	0,3322227	0,8816511	0,6680011	0,1327154	0,4305352
0,2108751	0,9373066	0,8431497	0,3454469	0,2942276	0,9606732	0,3702057	0,6167807
0,2962601	0,5283744	0,1226442	0,8496611	0,3729935	0,7552945	0,6447859	0,5202747
0,0751800	0,4888261	0,2565081	0,8135078	0,6214355	0,6093134	0,4455547	0,4108481
0,2405379	0,5082233	0,3993915	0,5018983	0,5273657	0,2783814	0,6229977	0,6263667
0,4016663	0,2991829	0,4612260	0,7651194	0,6313314	0,7881183	0,3006909	0,0424778
0,9150476	0,3975535	0,0876918	0,3973077	0,5043761	0,5044194	0,9125616	0,0525856
0,3369063	0,6330233	0,8582995	0,5956145	0,9167424	0,2135304	0,6773455	0,1414915
0,3236100	0,0023258	0,2036919	0,2829370	0,7733403	0,2752521	0,4467374	0,6650581
0,8384305	0,9676747	0,2503409	0,4148528	0,7710542	0,3206708	0,5401870	0,0819718
0,2748495	0,3642350	0,2971792					

ANEXO 0-3: NUMEROS ALEATORIOS PARA LA SECCION DE EMBUTIDO

0,5119241	0,6718440	0,0585201	0,5301030	0,4969906	0,6924763	0,3847497	0,3561766
0,6187659	0,0924206	0,4080324	0,0881955	0,0133634	0,0636621	0,2184568	0,4868870
0,4318660	0,5415471	0,2596585	0,9869651	0,7451354	0,8633437	0,8140832	0,4135705
0,9560104	0,7339400	0,0416654	0,9548041	0,5675316	0,6743052	0,0683554	0,8932376
0,2842566	0,1169536	0,4219758	0,2911201	0,0237845	0,9465762	0,0043673	0,3433215
0,0588649	0,6018626	0,6883012	0,4353693	0,8363127	0,1073663	0,8350047	0,6106793
0,4734419	0,1913750	0,1210723	0,7600081	0,1124726	0,0122748	0,7784678	0,7297356
0,2984824	0,2478841	0,2533518	0,6227619	0,1048165	0,0680328	0,6496574	0,3603523
0,0546101	0,5025884	0,8400419	0,2962592	0,5143497	0,3620675	0,9415289	0,9608909
0,9739975	0,9800713	0,0623060	0,7738910	0,1372013	0,6045827	0,2552148	0,8852749
0,5754628	0,2280897	0,2740131	0,4072956	0,0383880	0,2398654	0,7743780	0,6984224
0,5769668	0,4754291	0,7863566	0,0606670	0,9013328	0,6919904	0,6184724	0,5162547
0,5589170	0,5326734	0,2285640	0,6123795	0,4559795	0,6023151	0,6814307	0,6238300
0,3891853	0,7022431	0,1243061	0,1934678	0,8407844	0,3277755	0,3991263	0,7138569
0,3619498	0,7784412	0,5725409	0,4683473	0,5590192	0,9183906	0,1915687	0,9302658
0,9141000	0,1231737	0,5406153	0,7247885	0,0068925	0,3878096	0,0488979	0,3658007
0,1674409	0,3593724	0,5890893	0,3302961	0,7656333	0,7099452	0,1765228	0,0507262
0,9054012	0,2060989	0,2161774	0,1505357	0,9160767	0,4390815	0,6595827	0,1201676
0,0803334	0,3736885	0,9350531	0,8798172	0,4349716	0,2234163	0,6020852	0,7582070
0,7533010	0,5230541	0,1400574	0,1573826	0,3708637	0,0088690	0,0603207	0,5231872
0,6361281	0,9082564	0,5058233	0,1496508	0,4463530	0,9135857	0,0901734	0,2148943
0,1961920	0,2412249	0,3628228	0,8429589	0,9768288	0,0480601	0,0377688	0,5578003
0,2881572	0,8390429	0,1592226	0,2891982	0,8885048	0,9731533	0,8369998	0,0733881
0,8281564	0,0928458	0,4930894	0,8110149	0,0739169	0,4251169	0,8728759	0,2501600
0,2180716	0,4968580	0,3260705	0,8995412	0,1039598	0,9012985	0,3264201	0,5913117
0,6663363	0,9698732	0,9565941	0,3748214	0,9534116	0,3253535	0,8084099	0,3897428
0,7448077	0,3979244	0,8474995	0,4996064	0,0878429	0,5005341	0,1100283	0,1423488
0,5612485	0,8337065	0,6372546	0,5129179	0,8041754	0,1417521	0,8284547	0,9745997
0,0062470	0,1255710	0,7515212	0,2440000	0,0159195	0,9195194	0,3746900	0,2055922
0,3334299	0,2568317	0,6107779	0,5457293	0,5320054	0,3727403	0,9955328	0,6039272
0,7323597	0,5213721	0,4947662	0,8437554	0,8883483	0,7267824	0,4994112	0,0965597
0,4535753	0,1727788	0,5517887	0,9886529	0,3098865	0,4974785	0,2097447	0,7588358
0,7826779	0,1766290	0,0065518	0,5768458	0,1314609	0,8552026	0,4034072	0,1371836
0,3090127	0,5258102	0,3778475	0,8177845	0,5228091	0,5847198	0,1163290	0,3300086
0,6873454	0,5463074	0,6490187	0,3158363	0,5052082	0,3099190	0,8146622	0,2100138
0,9285585	0,2766051	0,3293153	0,3506021	0,1233043	0,3445593	0,1967973	0,8949126
0,8717991	0,8607216	0,6403976	0,3198667	0,1794094	0,2700453	0,2101706	0,1200175
0,1264352	0,2971705	0,9645606	0,5539339	0,6027999	0,6393993	0,0454181	0,2544475
0,1293116	0,0899785	0,9422308	0,2707145	0,7215642	0,8633095	0,1071385	0,4303915
0,6609902	0,8831813	0,0833287	0,6348251	0,1736363	0,1318307	0,3708237	0,0742496
0,9489034	0,5816448	0,6856711	0,6128053	0,2374146	0,3721433	0,8919337	0,9325274
0,8845949	0,1471724	0,5351441	0,9795946	0,8983479	0,3814853	0,1408641	0,2631799
0,5643896	0,4638814	0,4931543	0,4044988	0,8550589	0,2647712	0,4319112	0,0455273
0,9261390	0,1988831	0,5344300	0,8063988	0,4089694	0,4969070	0,5159040	0,8101069
0,7326456	0,0354640	0,5362415	0,3440356	0,4685856	0,4142443	0,6213121	0,2099763

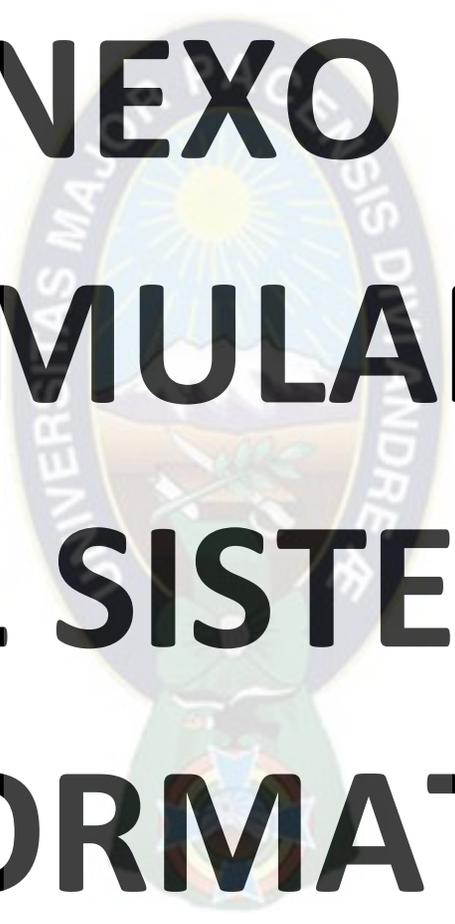
(Continuación anexo A-3)

0,9071384	0,8028356	0,2442324	0,6326341	0,3434278	0,5117931	0,3856108	0,3309137
0,6187281	0,8059866	0,6896643	0,5395111	0,9050613	0,9479952	0,5679820	0,4264733
0,0818370	0,7975375	0,5943613	0,7754415	0,0714146	0,5103128	0,9490859	0,7023998
0,7448556	0,1118513	0,9641218	0,5235094	0,5180246	0,9042078	0,8824668	0,7161858
0,3013076	0,7221699	0,2906910	0,3161643	0,0576528	0,3966425	0,4054884	0,6014527
0,4187741	0,0309690	0,2417923	0,4614463	0,1408447	0,3533087	0,3628644	0,7264661
0,1092222	0,5137841	0,6679288	0,6928376	0,7826788	0,9063532	0,0913927	0,7090874
0,2979907	0,0199680	0,6991633	0,8891488	0,9522058	0,1430621	0,2372136	0,9764013
0,8384658	0,1453197	0,1112314	0,9903873	0,5898939	0,8909209	0,8870759	0,4367835
0,2850641	0,9870164	0,7355413	0,4541703	0,1152339	0,6533312	0,4396277	0,6049451
0,5673985	0,9482143	0,2009044	0,7368089	0,9929371	0,9706044	0,0476204	0,1278866
0,0071104	0,6162610	0,0710614	0,6233867	0,1891507	0,3159827	0,0031932	0,6458432
0,3051078	0,2609441	0,6142638	0,4608852	0,3997044	0,1240763	0,0001055	0,9075689
0,4275104	0,5549226	0,0882428	0,3318986	0,6681886	0,4372055	0,7244302	0,4291649
0,3405427	0,4641629	0,6790457	0,5435929	0,9779978	0,4768525	0,5397651	0,0273423
0,7857153	0,5230396	0,0944758	0,7010062	0,9656173	0,9963412	0,4667544	0,1438409
0,0194989	0,9016902	0,8862407	0,4642774	0,0986481	0,8197338	0,1835885	0,3027741
0,3721607	0,5378676	0,1661020	0,0840931	0,8557911	0,3021565	0,9750312	0,1655715
0,6609620	0,8276039	0,7561214	0,9073583	0,3706349	0,0297297	0,2531693	0,5783375
0,9472549	0,7215777	0,5804805	0,2500479	0,4752799	0,7485133	0,4078771	0,7458528



ANEXO 0-4: NUMEROS ALEATORIOS PARA LA SECCION DE COCINADO

0,1650544	0,0636261	0,2605528	0,5024408	0,2489661	0,0950755	0,3570049	0,4725555
0,4654017	0,1850211	0,4210564	0,2349736	0,5544551	0,1081654	0,6781582	0,3148562
0,9647247	0,4185791	0,3948387	0,6203003	0,9245592	0,2107367	0,9553337	0,7025559
0,6796952	0,0128585	0,7798940	0,2669978	0,2625311	0,2308567	0,0133802	0,7660746
0,2640975	0,0068327	0,8844320	0,4165144	0,1166512	0,7831259	0,1727454	0,7088616
0,1210926	0,0855058	0,2274528	0,5762218	0,8499901	0,2000052	0,6143342	0,6029893
0,8838984	0,5721228	0,2566524	0,1320708	0,5001883	0,3360550	0,0806164	0,5612677
0,5806008	0,5809305	0,8305957	0,6623163	0,3070925	0,0637843	0,9748710	0,2526174
0,1693738	0,8153380	0,8178854	0,1320412	0,0630801	0,0904979	0,2253313	0,2125576
0,7640778	0,2512143	0,6914945	0,5352587	0,3477689	0,1680864	0,6670075	0,1365730
0,6913697	0,5386210	0,1911320	0,2123978	0,8866153	0,4317289	0,7910418	0,6069890
0,6756756	0,2103471	0,5197497	0,3000148	0,8514174	0,8098685	0,7032743	0,3136512
0,1951966	0,4486287	0,8066161	0,8239157	0,4292731	0,9522594	0,2042473	0,1689968
0,4698711	0,9258033	0,5045029	0,6990216	0,5713580	0,1673275	0,5011990	0,5728104
0,5485843	0,4394162	0,0965563	0,2942562	0,4841288	0,6301127	0,2916628	0,3998274
0,7947441	0,6100830	0,3882899	0,9492156	0,7419438	0,0018457	0,6008852	0,0519630
0,2646473	0,9521869	0,0866101	0,1010478	0,9527699	0,9647483	0,8782835	0,3309718
0,4465862	0,1873635	0,9121921	0,1402555	0,0453716	0,2381845	0,2793552	0,0382356
0,7805250	0,9547273	0,9931092	0,3941106	0,7398861	0,0122694	0,9399201	0,5933615
0,3065138	0,7671851	0,3093897	0,0375350	0,5725817	0,1179930	0,5318917	0,4559254
0,3156410	0,6193456	0,4330395	0,5179046	0,8345072	0,9910557	0,2474714	0,9017026
0,7845949	0,7103605	0,6213489	0,8532563	0,2518932	0,8004055	0,7970652	0,1676767
0,5961111	0,7426546	0,8158363	0,1107978	0,7609733	0,7931658	0,0433568	0,2740435
0,5846974	0,8798627	0,8308560	0,6845776	0,3025348	0,1077986	0,0162787	0,9594802
0,2066536	0,9319446	0,2559234	0,4796995	0,0454712	0,1839812	0,3835879	0,5083760
0,7718305	0,7111895	0,1141195	0,3159949	0,5274171	0,4965162	0,8421148	0,3777335
0,6261489	0,9015442	0,5208968	0,9570241	0,0160693	0,1054730	0,0894947	0,4160558
0,9111166	0,1905416	0,3088828	0,9964218	0,7425859	0,1585917	0,2458182	0,9958786
0,5587267	0,3300334	0,6939904	0,9658666	0,8133778	0,8588516	0,0941932	0,5492418
0,8602313	0,9687154	0,6635478	0,9376825	0,8912693	0,5614164	0,3079579	0,5599871
0,5615922	0,6942160	0,2533191	0,1599997	0,8640970	0,1516461	0,3451175	0,9531382
0,4414348	0,1698495	0,1593635	0,1241519	0,0426499	0,6219123	0,4490007	0,5186348
0,2776568	0,7949586	0,9735021	0,9731697	0,1622546	0,6519076	0,9269334	0,4980058
0,3379036	0,0347096	0,6892660	0,4575642	0,9560806	0,4623409	0,8456783	0,4471065
0,3550154	0,2515510	0,9917267	0,4603631	0,7936821			



**ANEXO B:
FORMULARIOS
DEL SISTEMA
INFORMÁTICO**



ANEXO C:

FORMULARIOS

DEL ESTUDIO

DE METODOS



**ANEXO D:
CODIGO
FUENTE**

ANEXO 0-1: CODIGO FUENTE, MODULO DE AUTENTICACION

```
Public Class frmAutenticacion
    Private Sub btnIngresar_Click(sender As Object, e As EventArgs) Handles
        btnIngresar.Click
            Dim adapterEmp As New ds_EmpleadosTableAdapters.CT_EmpleadosTableAdapter
            Dim odatasepEmp As New ds_Empleados
            adapterEmp.FillByAutenticacion(odatasepEmp.CT_Empleados, txtUsuario.Text,
            txtContraseña.Text, cboCargo.Text)
            If odatasepEmp.CT_Empleados.Count = 1 Then
                If cboCargo.Items(0) = cboCargo.Text Then
                    frmPrincipal.Show()
                    Me.Hide()
                End If
                If cboCargo.Items(1) = cboCargo.Text Then
                    frmPrincipal.AdministracionToolStripMenuItem.Enabled = False
                    frmPrincipal.AprovisionamientoToolStripMenuItem.Enabled = True
                    frmPrincipal.ProduccionToolStripMenuItem.Enabled = True
                    frmPrincipal.ComercializacionToolStripMenuItem.Enabled = True
                    frmPrincipal.ContabilidadToolStripMenuItem.Enabled = False
                    frmPrincipal.Show()
                    Me.Hide()
                End If
                If cboCargo.Items(2) = cboCargo.Text Then
                    frmPrincipal.AdministracionToolStripMenuItem.Enabled = False
                    frmPrincipal.AprovisionamientoToolStripMenuItem.Enabled = False
                    frmPrincipal.ProduccionToolStripMenuItem.Enabled = False
                    frmPrincipal.ComercializacionToolStripMenuItem.Enabled = False
                    frmPrincipal.ContabilidadToolStripMenuItem.Enabled = True
                    frmPrincipal.Show()
                    Me.Hide()
                End If
                If cboCargo.Items(3) = cboCargo.Text Then
                    frmPrincipal.AdministracionToolStripMenuItem.Enabled = False
                    frmPrincipal.AprovisionamientoToolStripMenuItem.Enabled = True
                    frmPrincipal.ProduccionToolStripMenuItem.Enabled = True
                    frmPrincipal.ComercializacionToolStripMenuItem.Enabled = False
                    frmPrincipal.ContabilidadToolStripMenuItem.Enabled = False
                    frmPrincipal.Show()
                    Me.Hide()
                End If
                If cboCargo.Items(4) = cboCargo.Text Then
                    frmPrincipal.AdministracionToolStripMenuItem.Enabled = False
                    frmPrincipal.AprovisionamientoToolStripMenuItem.Enabled = False
                    frmPrincipal.ProduccionToolStripMenuItem.Enabled = True
                    frmPrincipal.ComercializacionToolStripMenuItem.Enabled = True
                    frmPrincipal.ContabilidadToolStripMenuItem.Enabled = False
                    frmPrincipal.Show()
                    Me.Hide()
                End If
                If cboCargo.Items(5) = cboCargo.Text Then
                    frmPrincipal.AdministracionToolStripMenuItem.Enabled = False
                    frmPrincipal.AprovisionamientoToolStripMenuItem.Enabled = True
                    frmPrincipal.ProduccionToolStripMenuItem.Enabled = True
                    frmPrincipal.ComercializacionToolStripMenuItem.Enabled = True
                    frmPrincipal.ContabilidadToolStripMenuItem.Enabled = False
                    frmPrincipal.Show()
                    Me.Hide()
                End If
            Else
```

```

        MessageBox.Show("Datos no validos " & txtUsuario.Text & " no existe o la
contraseña es incorrecta" & vbCrLf & "Introduzca nuevamente los datos", "Error en la
autenticacion", MessageBoxButtons.OK, MessageBoxIcon.Error)
    End If
End Sub
Private Sub btnSalir_Click(sender As Object, e As EventArgs) Handles btnSalir.Click
    Application.Exit()
End Sub
Private Sub txtUsuario_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
txtUsuario.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub txtContraseña_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
txtContraseña.KeyPress
    If e.KeyChar = ChrW(Keys.Enter) Then
        btnIngresar.PerformClick()
    End If
End Sub
Private Sub frmAutenticacion_FormClosed(sender As Object, e As FormClosedEventArgs)
Handles MyBase.FormClosed
    Application.Exit()
End Sub
End Class

```



ANEXO 0-2: CODIGO FUENTE, GESTION DE PERSONAL, MODULO ADMINISTRACION

```
Public Class frmEmpleados
    Private Sub CT_EmpleadosBindingNavigatorSaveItem_Click(sender As Object, e As
EventArgs) Handles CT_EmpleadosBindingNavigatorSaveItem.Click
        Try
            Me.CT_EmpleadosBindingSource.EndEdit()
            Me.TableAdapterManager.UpdateAll(Me.Ds_Empleados)
            Actualizar()
            MessageBox.Show("Cambios guardados con exito", "Guardando datos",
MessageBoxButtons.OK, MessageBoxIcon.Information)
        Catch ex As Exception
            If MessageBox.Show("Ocurrio un error en la base de datos" & vbCrLf & "desea
continuar?", "Error al guardar", MessageBoxButtons.YesNo, MessageBoxIcon.Question) =
Windows.Forms.DialogResult.No Then
                Actualizar()
            End If
        End Try
    End Sub
    Private Sub frmEmpleados_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Actualizar()
    End Sub
    Private Sub Botones(ByVal x As Boolean)
        BindingNavigatorAddNewItem.Visible = x
        BindingNavigatorModificarItem.Visible = x
        BindingNavigatorDeleteItem.Visible = x
        CT_EmpleadosBindingNavigatorSaveItem.Visible = Not x
        BindingNavigatorCancelarItem.Visible = Not x
    End Sub
    Private Sub Bloquear(ByVal x As Boolean)
        Cod_empleadoMaskedTextBox.ReadOnly = x
        NombreMaskedTextBox.ReadOnly = x
        Direccion_emplMaskedTextBox.ReadOnly = x
        Telefono_emplMaskedTextBox.ReadOnly = x
        ContraseñaMaskedTextBox.ReadOnly = x
        Fecha_contratacionDateTimePicker.Enabled = Not x
        btnCargo.Visible = Not x
        SalarioMaskedTextBox.ReadOnly = x
        btnBuscaFoto.Visible = Not x
    End Sub
    Private Sub Actualizar()
        Me.CT_EmpleadosTableAdapter.Fill(Me.Ds_Empleados.CT_Empleados)
        Botones(True)
        Bloquear(True)
        Me.CT_EmpleadosDataGridView.Enabled = True
        Me.CT_EmpleadosDataGridView.Focus()
    End Sub
    Private Sub Cod_empleadoMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles Cod_empleadoMaskedTextBox.Validating
        If Cod_empleadoMaskedTextBox.MaskCompleted Then
            e.Cancel = False
            ErrorProvider1.SetError(Cod_empleadoMaskedTextBox, Nothing)
        Else
            e.Cancel = True
            ErrorProvider1.SetError(Cod_empleadoMaskedTextBox, "Dato no valido")
        End If
    End Sub
    Private Sub NombreMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles NombreMaskedTextBox.Validating
        If NombreMaskedTextBox.MaskCompleted Then
            e.Cancel = False
        End If
    End Sub
End Class
```

```

        ErrorProvider1.SetError(NombreMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(NombreMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub CIMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles CIMaskedTextBox.Validating
    If CIMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(CIMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(CIMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub Direccion_emplMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles Direccion_emplMaskedTextBox.Validating
    If Direccion_emplMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(Direccion_emplMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(Direccion_emplMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub Telefono_emplMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles Telefono_emplMaskedTextBox.Validating
    If Telefono_emplMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(Telefono_emplMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(Telefono_emplMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub SalarioMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles SalarioMaskedTextBox.Validating
    If SalarioMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(SalarioMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(SalarioMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub Cod_empleadoMaskedTextBox_KeyPress(sender As Object, e As
KeyPressEventArgs) Handles Cod_empleadoMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub NombreMaskedTextBox_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles NombreMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub

```

```

Private Sub CIMaskedTextBox_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles CIMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub Direccion_emplMaskedTextBox_KeyPress(sender As Object, e As
KeyPressEventArgs) Handles Direccion_emplMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub Telefono_emplMaskedTextBox_KeyPress(sender As Object, e As
KeyPressEventArgs) Handles Telefono_emplMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles btnCargo.Click
    frmManualfunc.Show()
End Sub
Private Sub SalarioMaskedTextBox_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles SalarioMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub ContraseñaMaskedTextBox_KeyPress(sender As Object, e As
KeyPressEventArgs) Handles ContraseñaMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub btnBuscar_Click(sender As Object, e As EventArgs) Handles btnBuscar.Click
    Dim dato As String = txtDato.Text
    Try
        If cboModo.Items(1) = cboModo.Text Then
            Me.CT_EmpleadosTableAdapter.FillByCodigoemp(Me.Ds_Empleados.CT_Empleados,
dato)
        End If
        If cboModo.Items(2) = cboModo.Text Then
            Me.CT_EmpleadosTableAdapter.FillBynombre(Me.Ds_Empleados.CT_Empleados,
dato & "%")
        End If
    Catch ex As Exception
    End Try
End Sub
Private Sub cboModo_SelectedValueChanged(sender As Object, e As EventArgs) Handles
cboModo.SelectedValueChanged
    If cboModo.Items(0) = cboModo.Text Then
        Me.CT_EmpleadosTableAdapter.Fill(Me.Ds_Empleados.CT_Empleados)
        txtDato.Clear()
    Else
        txtDato.Clear()
        txtDato.Focus()
    End If
End Sub

```

```

End Sub
Private Sub txtDato_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
txtDato.KeyPress
    If e.KeyChar = ChrW(Keys.Enter) Then
        btnBuscar.PerformClick()
    End If
End Sub
Private Sub BindingNavigatorAddNewItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorAddNewItem.Click
    Botones(False)
    Bloquear(False)
    Me.CT_EmpleadosDataGridView.Enabled = False
    ErrorProvider1.Dispose()
    Cod_empleadoMaskedTextBox.Focus()
End Sub
Private Sub BindingNavigatorModificarItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorModificarItem.Click
    Botones(False)
    Bloquear(False)
    Me.CT_EmpleadosDataGridView.Enabled = False
    ErrorProvider1.Dispose()
    Cod_empleadoMaskedTextBox.Focus()
End Sub
Private Sub BindingNavigatorDeleteItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorDeleteItem.Click
    Botones(False)
    Me.CT_EmpleadosDataGridView.Enabled = False
End Sub
Private Sub BindingNavigatorCancelarItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorCancelarItem.Click
    Actualizar()
End Sub
Private Sub btnBuscaFoto_Click(sender As Object, e As EventArgs) Handles
btnBuscaFoto.Click
    Try
        Dim foto As String
        OpenFileDialog1.Filter = "Imágenes *.jpg|*.jpg|Imágenes *.png|*.png|Todos los
archivos *.*|*.*"
        OpenFileDialog1.ShowDialog()
        foto = OpenFileDialog1.FileName
        FotoPictureBox.Image = System.Drawing.Image.FromFile(foto)
    Catch ex As Exception
        MessageBox.Show("Imagen no valida", "Error de tipo de dato",
MessageBoxButtons.OK, MessageBoxIcon.Information)
    End Try
End Sub
End Class

```

ANEXO 0-3: CODIGO FUENTE, MANUAL DE FUNCIONES, MODULO ADMINISTRACION

```
Public Class frmOrganigrama
    Private Sub btnGerenteG_Click(sender As Object, e As EventArgs) Handles
btnGerenteG.Click
        Dim loPSI As New ProcessStartInfo
        Dim loProceso As New Process
        loPSI.FileName = Application.StartupPath & "\mf_ggeneral.pdf"
        Try
            loProceso = Process.Start(loPSI)
        Catch Exp As Exception
            MessageBox.Show(Exp.Message, "Carga de archivo", MessageBoxButtons.OK,
MessageBoxIcon.Information)
        End Try
    End Sub
    Private Sub btnGerenteprod_Click(sender As Object, e As EventArgs) Handles
btnGerenteprod.Click
        Dim loPSI As New ProcessStartInfo
        Dim loProceso As New Process
        loPSI.FileName = Application.StartupPath & "\mf_gproduccion.pdf"
        Try
            loProceso = Process.Start(loPSI)
        Catch Exp As Exception
            MessageBox.Show(Exp.Message, "Carga de archivo", MessageBoxButtons.OK,
MessageBoxIcon.Information)
        End Try
    End Sub
    Private Sub btnContador_Click(sender As Object, e As EventArgs) Handles
btnContador.Click
        Dim loPSI As New ProcessStartInfo
        Dim loProceso As New Process
        loPSI.FileName = Application.StartupPath & "\mf_contador.pdf"
        Try
            loProceso = Process.Start(loPSI)
        Catch Exp As Exception
            MessageBox.Show(Exp.Message, "Carga de archivo", MessageBoxButtons.OK,
MessageBoxIcon.Information)
        End Try
    End Sub
    Private Sub btnSecretaria_Click(sender As Object, e As EventArgs) Handles
btnSecretaria.Click
        Dim loPSI As New ProcessStartInfo
        Dim loProceso As New Process
        loPSI.FileName = Application.StartupPath & "\mf_secretaria.pdf"
        Try
            loProceso = Process.Start(loPSI)
        Catch Exp As Exception
            MessageBox.Show(Exp.Message, "Carga de archivo", MessageBoxButtons.OK,
MessageBoxIcon.Information)
        End Try
    End Sub
    Private Sub btnAsGeProd_Click(sender As Object, e As EventArgs) Handles
btnAsGeProd.Click
        Dim loPSI As New ProcessStartInfo
        Dim loProceso As New Process
        loPSI.FileName = Application.StartupPath & "\mf_asistenteg.pdf"
        Try
            loProceso = Process.Start(loPSI)
        Catch Exp As Exception
            MessageBox.Show(Exp.Message, "Carga de archivo", MessageBoxButtons.OK,
MessageBoxIcon.Information)
        End Try
    End Sub
End Class
```

```

        End Try
    End Sub
    Private Sub btnJefeProd_Click(sender As Object, e As EventArgs) Handles
btnJefeProd.Click
        Dim loPSI As New ProcessStartInfo
        Dim loProceso As New Process
        loPSI.FileName = Application.StartupPath & "\mf_jdeplanta.pdf"
        Try
            loProceso = Process.Start(loPSI)
        Catch Exp As Exception
            MessageBox.Show(Exp.Message, "Carga de archivo", MessageBoxButtons.OK,
MessageBoxIcon.Information)
        End Try
    End Sub
    Private Sub btnEncAlmIns_Click(sender As Object, e As EventArgs) Handles
btnEncAlmIns.Click
        Dim loPSI As New ProcessStartInfo
        Dim loProceso As New Process
        loPSI.FileName = Application.StartupPath & "\mf_almaceni.pdf"
        Try
            loProceso = Process.Start(loPSI)
        Catch Exp As Exception
            MessageBox.Show(Exp.Message, "Carga de archivo", MessageBoxButtons.OK,
MessageBoxIcon.Information)
        End Try
    End Sub
    Private Sub btnEncAlmProdTerm_Click(sender As Object, e As EventArgs) Handles
btnEncAlmProdTerm.Click
        Dim loPSI As New ProcessStartInfo
        Dim loProceso As New Process
        loPSI.FileName = Application.StartupPath & "\mf_almacenpt.pdf"
        Try
            loProceso = Process.Start(loPSI)
        Catch Exp As Exception
            MessageBox.Show(Exp.Message, "Carga de archivo", MessageBoxButtons.OK,
MessageBoxIcon.Information)
        End Try
    End Sub
    Private Sub btnAyuAlmIns_Click(sender As Object, e As EventArgs) Handles
btnAyuAlmIns.Click
        Dim loPSI As New ProcessStartInfo
        Dim loProceso As New Process
        loPSI.FileName = Application.StartupPath & "\mf_aalmaceni.pdf"
        Try
            loProceso = Process.Start(loPSI)
        Catch Exp As Exception
            MessageBox.Show(Exp.Message, "Carga de archivo", MessageBoxButtons.OK,
MessageBoxIcon.Information)
        End Try
    End Sub
    Private Sub btnAyuAlmProdTern_Click(sender As Object, e As EventArgs) Handles
btnAyuAlmProdTern.Click
        Dim loPSI As New ProcessStartInfo
        Dim loProceso As New Process
        loPSI.FileName = Application.StartupPath & "\mf_aalmacenpt.pdf"
        Try
            loProceso = Process.Start(loPSI)
        Catch Exp As Exception
            MessageBox.Show(Exp.Message, "Carga de archivo", MessageBoxButtons.OK,
MessageBoxIcon.Information)
        End Try
    End Sub

```

```

End Sub
Private Sub btnCortador_Click(sender As Object, e As EventArgs) Handles
btnCortador.Click
    Dim loPSI As New ProcessStartInfo
    Dim loProceso As New Process
    loPSI.FileName = Application.StartupPath & "\\mf_smolido.pdf"
    Try
        loProceso = Process.Start(loPSI)
    Catch Exp As Exception
        MessageBox.Show(Exp.Message, "Carga de archivo", MessageBoxButtons.OK,
MessageBoxIcon.Information)
    End Try
End Sub
Private Sub btnOpCutter_Click(sender As Object, e As EventArgs) Handles
btnOpCutter.Click
    Dim loPSI As New ProcessStartInfo
    Dim loProceso As New Process
    loPSI.FileName = Application.StartupPath & "\\mf_smezclado.pdf"
    Try
        loProceso = Process.Start(loPSI)
    Catch Exp As Exception
        MessageBox.Show(Exp.Message, "Carga de archivo", MessageBoxButtons.OK,
MessageBoxIcon.Information)
    End Try
End Sub
Private Sub btnAyuOpCutter_Click(sender As Object, e As EventArgs) Handles
btnAyuOpCutter.Click
    Dim loPSI As New ProcessStartInfo
    Dim loProceso As New Process
    loPSI.FileName = Application.StartupPath & "\\mf_asmezclado.pdf"
    Try
        loProceso = Process.Start(loPSI)
    Catch Exp As Exception
        MessageBox.Show(Exp.Message, "Carga de archivo", MessageBoxButtons.OK,
MessageBoxIcon.Information)
    End Try
End Sub
Private Sub btnEmbutidor_Click(sender As Object, e As EventArgs) Handles
btnEmbutidor.Click
    Dim loPSI As New ProcessStartInfo
    Dim loProceso As New Process
    loPSI.FileName = Application.StartupPath & "\\mf_sembutido.pdf"
    Try
        loProceso = Process.Start(loPSI)
    Catch Exp As Exception
        MessageBox.Show(Exp.Message, "Carga de archivo", MessageBoxButtons.OK,
MessageBoxIcon.Information)
    End Try
End Sub
Private Sub btnAyuEmbutidor_Click(sender As Object, e As EventArgs) Handles
btnAyuEmbutidor.Click
    Dim loPSI As New ProcessStartInfo
    Dim loProceso As New Process
    loPSI.FileName = Application.StartupPath & "\\mf_asebutido.pdf"
    Try
        loProceso = Process.Start(loPSI)
    Catch Exp As Exception
        MessageBox.Show(Exp.Message, "Carga de archivo", MessageBoxButtons.OK,
MessageBoxIcon.Information)
    End Try
End Sub

```

```
Private Sub btnEncHorno_Click(sender As Object, e As EventArgs) Handles  
btnEncHorno.Click  
Dim loPSI As New ProcessStartInfo  
Dim loProceso As New Process  
loPSI.FileName = Application.StartupPath & "\mf_shornos.pdf"  
Try  
    loProceso = Process.Start(loPSI)  
Catch Exp As Exception  
    MessageBox.Show(Exp.Message, "Carga de archivo", MessageBoxButtons.OK,  
MessageBoxIcon.Information)  
End Try  
End Sub  
Private Sub btnEncCoci_Click(sender As Object, e As EventArgs) Handles  
btnEncCoci.Click  
Dim loPSI As New ProcessStartInfo  
Dim loProceso As New Process  
loPSI.FileName = Application.StartupPath & "\mf_shornos.pdf"  
Try  
    loProceso = Process.Start(loPSI)  
Catch Exp As Exception  
    MessageBox.Show(Exp.Message, "Carga de archivo", MessageBoxButtons.OK,  
MessageBoxIcon.Information)  
End Try  
End Sub  
End Class
```



ANEXO 0-4: CODIGO FUENTE, BOLETAS DE PAGO, MODULO ADMINISTRACION

```
Public Class frmPlanillaSueldos
    Dim sueldobase, bp, he, che, ba, dom, ob, afp, rciva, od, tg, td, lp As Decimal
    Dim gestion, codsuel, carnet, dt, dd As Integer
    Dim mes, apenom, carg As String
    Dim fcont, hoy, diferencia As Date
    Private Sub Imprimir()
        PrintDialog1.ShowDialog()
        PrintForm1.PrinterSettings = PrintDialog1.PrinterSettings
        PrintForm1.Print()
    Private Sub Button4_Click(sender As Object, e As EventArgs) Handles Button4.Click
        frmBuscaEmpleado.ShowDialog()
    End Sub
    Private Sub Button2_Click(sender As Object, e As EventArgs)
        Close()
    End Sub
    Private Sub txtSuelBase_TextChanged(sender As Object, e As EventArgs) Handles
txtSuelBase.TextChanged
        Try
            sueldobase = txtSuelBase.Text
            txtTG.Text = Format(sueldobase + bp + che + ba + dom + ob, "0.00")
        Catch ex As Exception
        End Try

    End Sub
    Private Sub txtBP_TextChanged(sender As Object, e As EventArgs) Handles
txtBP.TextChanged
        Try
            bp = txtBP.Text
            txtTG.Text = Format(sueldobase + bp + che + ba + dom + ob, "0.00")
        Catch ex As Exception
        End Try

    End Sub
    Private Sub txtHE_TextChanged(sender As Object, e As EventArgs) Handles
txtHE.TextChanged
        Try
            he = txtHE.Text
            che = (sueldobase / 207.84) * 0.3 * he
            txtCHE.Text = che
            txtTG.Text = Format(sueldobase + bp + che + ba + dom + ob, "0.00")
        Catch ex As Exception
        End Try

    End Sub
    Private Sub txtBA_TextChanged(sender As Object, e As EventArgs) Handles
txtBA.TextChanged
        Try
            ba = txtBA.Text
            txtTG.Text = Format(sueldobase + bp + che + ba + dom + ob, "0.00")
        Catch ex As Exception
        End Try

    End Sub
    Private Sub TextBox1_TextChanged(sender As Object, e As EventArgs) Handles
txtdom.TextChanged
        Try
            dom = txtdom.Text
            txtTG.Text = Format(sueldobase + bp + che + ba + dom + ob, "0.00")
        Catch ex As Exception
        End Try

    End Sub
End Class
```

```

Private Sub txtob_TextChanged(sender As Object, e As EventArgs) Handles
txtob.TextChanged
    Try
        ob = txtob.Text
        txtTG.Text = Format(sueldobase + bp + che + ba + dom + ob, "0.00")
    Catch ex As Exception
    End Try
End Sub
Private Sub txtrciva_TextChanged(sender As Object, e As EventArgs) Handles
txtrciva.TextChanged
    Try
        rciva = txtrciva.Text
        txtTD.Text = Format(afp + rciva + od, "0.00")
    Catch ex As Exception
    End Try
End Sub
Private Sub txtOtros_TextChanged(sender As Object, e As EventArgs) Handles
txtOtros.TextChanged
    Try
        od = txtOtros.Text
        txtTD.Text = Format(afp + rciva + od, "0.00")
    Catch ex As Exception
    End Try
End Sub
Private Sub txtTG_TextChanged(sender As Object, e As EventArgs) Handles
txtTG.TextChanged
    Try
        tg = txtTG.Text
        afp = tg * 0.1221
        txtAFP.Text = Format(afp, "0.00")
        rciva = (tg - afp - 2400) * 0.13
        If rciva > 0 Then
            txtrciva.Text = Format(rciva, "0.00")
        Else
            txtrciva.Text = 0
        End If
        txtLP.Text = Format(tg - td, "0.00")
    Catch ex As Exception
    End Try
End Sub
Private Sub txtAFP_TextChanged(sender As Object, e As EventArgs) Handles
txtAFP.TextChanged
    Try
        txtTD.Text = Format(afp + rciva + od, "0.00")
    Catch ex As Exception
    End Try
End Sub
Private Sub txtTD_TextChanged(sender As Object, e As EventArgs) Handles
txtTD.TextChanged
    Try
        td = txtTD.Text
        txtLP.Text = Format(tg - td, "0.00")
    Catch ex As Exception
    End Try
End Sub
Private Sub ComboBox1_SelectedValueChanged(sender As Object, e As EventArgs) Handles
cboMes.SelectedValueChanged
    If cboMes.Items(0) = cboMes.Text Then
        mes = cboMes.Text
    End If
    If cboMes.Items(1) = cboMes.Text Then

```

```

        mes = cboMes.Text
    End If
    If cboMes.Items(2) = cboMes.Text Then
        mes = cboMes.Text
    End If
    If cboMes.Items(3) = cboMes.Text Then
        mes = cboMes.Text
    End If
    If cboMes.Items(4) = cboMes.Text Then
        mes = cboMes.Text
    End If
    If cboMes.Items(5) = cboMes.Text Then
        mes = cboMes.Text
    End If
    If cboMes.Items(6) = cboMes.Text Then
        mes = cboMes.Text
    End If
    If cboMes.Items(7) = cboMes.Text Then
        mes = cboMes.Text
    End If
    If cboMes.Items(8) = cboMes.Text Then
        mes = cboMes.Text
    End If
    If cboMes.Items(9) = cboMes.Text Then
        mes = cboMes.Text
    End If
    If cboMes.Items(10) = cboMes.Text Then
        mes = cboMes.Text
    End If
    If cboMes.Items(11) = cboMes.Text Then
        mes = cboMes.Text
    End If
End Sub
Private Sub cboGestion_SelectedValueChanged(sender As Object, e As EventArgs) Handles
cboGestion.SelectedValueChanged
    If cboGestion.Items(0) = cboGestion.Text Then
        gestion = cboGestion.Text
    End If
    If cboGestion.Items(1) = cboGestion.Text Then
        gestion = cboGestion.Text
    End If
    If cboGestion.Items(2) = cboGestion.Text Then
        gestion = cboGestion.Text
    End If
    If cboGestion.Items(3) = cboGestion.Text Then
        gestion = cboGestion.Text
    End If
    If cboGestion.Items(4) = cboGestion.Text Then
        gestion = cboGestion.Text
    End If
    If cboGestion.Items(5) = cboGestion.Text Then
        gestion = cboGestion.Text
    End If
    If cboGestion.Items(6) = cboGestion.Text Then
        gestion = cboGestion.Text
    End If
    If cboGestion.Items(7) = cboGestion.Text Then
        gestion = cboGestion.Text
    End If
    If cboGestion.Items(8) = cboGestion.Text Then

```

```

        gestion = cboGestion.Text
    End If
    If cboGestion.Items(9) = cboGestion.Text Then
        gestion = cboGestion.Text
    End If
End Sub
Private Sub txtApNo_TextChanged(sender As Object, e As EventArgs) Handles
txtApNo.TextChanged
    apenom = txtApNo.Text
End Sub
Private Sub txtCI_TextChanged(sender As Object, e As EventArgs) Handles
txtCI.TextChanged
    carnet = txtCI.Text
End Sub
Private Sub txtFechcont_TextChanged(sender As Object, e As EventArgs) Handles
txtFechcont.TextChanged
    fcont = txtFechcont.Text
End Sub
Private Sub txtCar_TextChanged(sender As Object, e As EventArgs) Handles
txtCar.TextChanged
    carg = txtCar.Text
End Sub
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim odaptersueld As New dsSueldosTableAdapters.CT_SueldosTableAdapter
    codsuel = odaptersueld.ScalarQueryCuentaSuel
    odaptersueld.InsertQuerySueldo(codsuel, carnet, apenom, carg, fcont, mes,
gestion, txtDiatrab.Text, salario, ba, he, che, bp, dom, ob, tg, afp, rciva, od, td, lp)
    Imprimir()
End Sub
Private Sub txtDiasDom_TextChanged(sender As Object, e As EventArgs)
End Sub
Private Sub txtDiatrab_TextChanged(sender As Object, e As EventArgs) Handles
txtDiatrab.TextChanged
    dt = txtDiatrab.Text
End Sub
Private Sub Button2_Click_1(sender As Object, e As EventArgs) Handles Button2.Click
    Dim ant As Decimal
    Dim bpdia, bpmes, bpgest, bddia, bdmes, bdgest As Integer
    Dim odaptercont As New ds_EmpleadosTableAdapters.CT_EmpleadosTableAdapter
    bddia = odaptercont.ScalarQueryDiaCont(txtcodemp.Text)
    bdmes = odaptercont.ScalarQueryMesCont(txtcodemp.Text)
    bdgest = odaptercont.ScalarQueryGestCont(txtcodemp.Text)
    bpgest = DateTimePicker1.Value.Year
    bpmes = DateTimePicker1.Value.Month
    bpdia = DateTimePicker1.Value.Day
    ant = (bpgest - bdgest) + (bpmes - bdmes) / 12 + (bpdia - bddia) / 31
    Select Case ant
        Case Is < 2
            txtBA.Text = 0
        Case Is < 5
            ba = sueldobase * 0.05
            txtBA.Text = ba
        Case Is < 8
            ba = sueldobase * 0.11
            txtBA.Text = ba
        Case Is < 11
            ba = sueldobase * 0.18
            txtBA.Text = ba
        Case Is < 15
            ba = sueldobase * 0.26
            txtBA.Text = ba
    End Select
End Sub

```

```
Case Is < 20
    ba = sueldobase * 0.34
    txtBA.Text = ba
Case Is < 25
    ba = sueldobase * 0.42
    txtBA.Text = ba
Case Is >= 25
    ba = sueldobase * 0.5
    txtBA.Text = ba
Case Else
End Select
End Sub
Private Sub txtLP_TextChanged(sender As Object, e As EventArgs) Handles
txtLP.TextChanged
    lp = txtLP.Text
End Sub
Private Sub CheckBox1_CheckedChanged(sender As Object, e As EventArgs) Handles
CheckBox1.CheckedChanged
    txtdom.Text = Format((sueldobase / dt) * 4, "0.00")
End Sub
End Class
```



ANEXO 0-5: CODIGO FUENTE, PLANILLA DE SUELDOS PATRONALES, MODULO ADMINISTRACION

```
Public Class frmPlanSalPat
    Private Sub CT_SueldosBindingNavigatorSaveItem_Click(sender As Object, e As
EventArgs)
        Me.Validate()
        Me.CT_SueldosBindingSource.EndEdit()
        Me.TableAdapterManager.UpdateAll(Me.DsSueldos)
    End Sub
    Private Sub frmPlanSalPat_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Me.CT_SueldosTableAdapter.Fill(Me.DsSueldos.CT_Sueldos)
    End Sub
    Private Sub Imprimir()
        PrintDialog1.ShowDialog()
        PrintForm1.PrinterSettings = PrintDialog1.PrinterSettings
        PrintForm1.Print()
    End Sub
    Private Sub btnFiltro_Click(sender As Object, e As EventArgs) Handles btnFiltro.Click
        Me.CT_SueldosTableAdapter.FillByMesGestion(Me.DsSueldos.CT_Sueldos, cboMes.Text,
cboGestion.Text)
    End Sub
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        Imprimir()
    End Sub
End Class
```



ANEXO 0-6: CODIGO FUENTE, GESTION DE PROVEEDORES, MODULO PROVISION

```
Public Class frmProveedor
    Private Sub CT_ProveedoresBindingNavigatorSaveItem_Click(sender As Object, e As
EventArgs) Handles CT_ProveedoresBindingNavigatorSaveItem.Click
        Try
            Me.CT_ProveedoresBindingSource.EndEdit()
            Me.TableAdapterManager.UpdateAll(Me.DsProveedor)
            Actualizar()
            MessageBox.Show("Cambios guardados con éxito", "Guardando datos",
MessageBoxButtons.OK, MessageBoxIcon.Information)
        Catch ex As Exception
            If MessageBox.Show("Ocurrió un error en la base de datos" & vbCrLf & "desea
continuar?", "Error al guardar", MessageBoxButtons.YesNo, MessageBoxIcon.Question) =
Windows.Forms.DialogResult.No Then
                Actualizar()
            End If
        End Try
    End Sub
    Private Sub frmProveedor_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Actualizar()
    End Sub
    Private Sub Botones(ByVal x As Boolean)
        BindingNavigatorAddNewItem.Visible = x
        BindingNavigatorModificarItem.Visible = x
        BindingNavigatorDeleteItem.Visible = x
        CT_ProveedoresBindingNavigatorSaveItem.Visible = Not x
        BindingNavigatorCancelarItem.Visible = Not x
    End Sub
    Private Sub Bloquear(ByVal x As Boolean)
        Cod_ProveedorMaskedTextBox.ReadOnly = x
        Empresa_ProvMaskedTextBox.ReadOnly = x
        Direccion_ProvMaskedTextBox.ReadOnly = x
        Telefono_ProvMaskedTextBox.ReadOnly = x
        Saldo_cppMaskedTextBox.ReadOnly = x
        CI_NITMaskedTextBox.ReadOnly = x
    End Sub
    Private Sub Actualizar()
        Me.CT_ProveedoresTableAdapter.Fill(Me.DsProveedor.CT_Proveedores)
        Botones(True)
        Bloquear(True)
        Me.CT_ProveedoresDataGridView.Enabled = True
        Me.CT_ProveedoresDataGridView.Focus()
    End Sub
    Private Sub Cod_ProvMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles Cod_ProveedorMaskedTextBox.Validating
        If Cod_ProveedorMaskedTextBox.MaskCompleted Then
            e.Cancel = False
            ErrorProvider1.SetError(Cod_ProveedorMaskedTextBox, Nothing)
        Else
            e.Cancel = True
            ErrorProvider1.SetError(Cod_ProveedorMaskedTextBox, "Dato no valido")
        End If
    End Sub
    Private Sub EmpresaMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles Empresa_ProvMaskedTextBox.Validating
        If Empresa_ProvMaskedTextBox.MaskCompleted Then
            e.Cancel = False
            ErrorProvider1.SetError(Empresa_ProvMaskedTextBox, Nothing)
        Else
            e.Cancel = True
        End If
    End Sub
End Class
```

```

        ErrorProvider1.SetError(Empresa_ProvMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub DireccionMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles Direccion_ProvMaskedTextBox.Validating
    If Direccion_ProvMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(Direccion_ProvMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(Direccion_ProvMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub TelefonoMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles Telefono_ProvMaskedTextBox.Validating
    If Telefono_ProvMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(Telefono_ProvMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(Telefono_ProvMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub CI_NITMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles CI_NITMaskedTextBox.Validating
    If CI_NITMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(CI_NITMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(CI_NITMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub Saldo_provMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles Saldo_cppMaskedTextBox.Validating
    If Saldo_cppMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(Saldo_cppMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(Saldo_cppMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub Cod_ProvMaskedTextBox_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles Cod_ProveedorMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub EmpresaMaskedTextBox_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles Empresa_ProvMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub DireccionMaskedTextBox_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles Direccion_ProvMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
    End If
End Sub

```

```

        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub TelefonoMaskedTextBox_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles Telefono_ProvMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub CI_NITMaskedTextBox_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles CI_NITMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub btnBuscar_Click(sender As Object, e As EventArgs) Handles btnBuscar.Click
    Dim dato As String = txtDato.Text
    Try
        If cboModo.Items(1) = cboModo.Text Then
            Me.CT_ProveedoresTableAdapter.FillByCodigo(Me.DsProveedor.CT_Proveedores,
dato)
        End If
        If cboModo.Items(2) = cboModo.Text Then
            Me.CT_ProveedoresTableAdapter.FillByempresa(Me.DsProveedor.CT_Proveedores, dato & "%")
        End If
    Catch ex As Exception
    End Try
End Sub
Private Sub cboModo_SelectedValueChanged(sender As Object, e As EventArgs) Handles
cboModo.SelectedValueChanged
    If cboModo.Items(0) = cboModo.Text Then
        Me.CT_ProveedoresTableAdapter.Fill(Me.DsProveedor.CT_Proveedores)
        txtDato.Clear()
    Else
        txtDato.Clear()
        txtDato.Focus()
    End If
End Sub
Private Sub txtDato_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
txtDato.KeyPress
    If e.KeyChar = ChrW(Keys.Enter) Then
        btnBuscar.PerformClick()
    End If
End Sub
Private Sub BindingNavigatorAddNewItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorAddNewItem.Click
    Botones(False)
    Bloquear(False)
    Me.CT_ProveedoresDataGridView.Enabled = False
    ErrorProvider1.Dispose()
    Cod_ProveedorMaskedTextBox.Focus()
End Sub
Private Sub BindingNavigatorModificarItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorModificarItem.Click
    Botones(False)
    Bloquear(False)
    Me.CT_ProveedoresDataGridView.Enabled = False
    ErrorProvider1.Dispose()

```

```
        Cod_ProveedorMaskedTextBox.Focus()
    End Sub
    Private Sub BindingNavigatorDeleteItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorDeleteItem.Click
        Botones(False)
        Me.CT_ProveedoresDataGridView.Enabled = False
    End Sub
    Private Sub BindingNavigatorCancelarItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorCancelarItem.Click
        Actualizar()
    End Sub
End Class
```



ANEXO 0-7: CODIGO FUENTE, COMPRAS, MODULO PROVISION

```
Public Class frmCompra
    Private Sub frmCompra_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Dim odaptercomp As New dsComprasTableAdapters.CT_ComprasTableAdapter
        contcompras = odaptercomp.ScalarQueryCuentaCompras
        txtCodComp.Text = contcompras + 1
    End Sub
    Private Sub Imprimir()
        PrintDialog1.ShowDialog()
        PrintForm1.PrinterSettings = PrintDialog1.PrinterSettings
        PrintForm1.Print()
    End Sub
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        Dim odaptercomp As New dsComprasTableAdapters.CT_ComprasTableAdapter
        odaptercomp.InsertQueryCompras(txtCodComp.Text, txtProvee.Text, txtCantidad.Text,
        txtDescIn.Text, txtImporte.Text)
        Dim odapterinsum As New dsInsumosTableAdapters.CT_InsumosTableAdapter
        odapterinsum.UpdateQueryInsumoscant(cantact + cantidadinsumo, codinsumo)
        MessageBox.Show("Registro guardado con éxito", "Guardando datos",
        MessageBoxButtons.OK, MessageBoxIcon.Information)
        Imprimir()
        contcompras = odaptercomp.ScalarQueryCuentaCompras
        txtCodComp.Text = contcompras + 1
        txtDescIn.Clear()
        txtProvee.Clear()
        txtCantidad.Clear()
        txtImporte.Clear()
        txtProvee.Focus()
    End Sub
    Private Sub txtDescIn_KeyPress(sender As Object, e As KeyEventArgs) Handles
    txtDescIn.KeyPress
        If e.KeyChar = Convert.ToChar(Keys.Enter) Then
            e.Handled = True
            SendKeys.Send("{TAB}")
        End If
    End Sub
    Private Sub txtCantidad_KeyPress(sender As Object, e As KeyEventArgs) Handles
    txtCantidad.KeyPress
        If e.KeyChar = Convert.ToChar(Keys.Enter) Then
            e.Handled = True
            SendKeys.Send("{TAB}")
        End If
    End Sub
    Private Sub txtImporte_KeyPress(sender As Object, e As KeyEventArgs) Handles
    txtImporte.KeyPress
        If e.KeyChar = ChrW(Keys.Enter) Then
            Button1.PerformClick()
        End If
    End Sub
    Private Sub btnBuscaProv_Click(sender As Object, e As EventArgs) Handles
    btnBuscaProv.Click
        frmBuscaProveedor.ShowDialog()
    End Sub
    Private Sub btnBuscaInsumo_Click(sender As Object, e As EventArgs) Handles
    btnBuscaInsumo.Click
        frmBuscaInsumo.ShowDialog()
    End Sub
    Dim cantidadinsumo As Integer
    Private Sub txtCodInsumo_TextChanged(sender As Object, e As EventArgs) Handles
    txtCodInsumo.TextChanged
```

```
        codinsum = txtCodInsumo.Text
    End Sub
    Private Sub txtCantidad_TextChanged(sender As Object, e As EventArgs) Handles
txtCantidad.TextChanged
        Try
            cantidadinsumo = txtCantidad.Text
        Catch ex As Exception
        End Try
    End Sub
End Class
```



ANEXO 0-8: CODIGO FUENTE, PRODUCTOS TERMINADOS, MODULO PRODUCCION

```
Public Class frmProductos
    Private Sub CT_ProductosBindingNavigatorSaveItem_Click(sender As Object, e As
EventArgs) Handles CT_ProductosBindingNavigatorSaveItem.Click
        Try
            Me.CT_ProductosBindingSource.EndEdit()
            Me.TableAdapterManager.UpdateAll(Me.DsProductos)
            Actualizar()
            MessageBox.Show("Cambios guardados con exito", "Guardando datos",
MessageBoxButtons.OK, MessageBoxIcon.Information)
        Catch ex As Exception
            If MessageBox.Show("Ocurrio un error en la base de datos" & vbCrLf & "desea
continuar?", "Error al guardar", MessageBoxButtons.YesNo, MessageBoxIcon.Question) =
Windows.Forms.DialogResult.No Then
                Actualizar()
            End If
        End Try
    End Sub
    Private Sub frmInventario_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Actualizar()
    End Sub
    Private Sub Botones(ByVal x As Boolean)
        BindingNavigatorAddNewItem.Visible = x
        BindingNavigatorModificarItem.Visible = x
        BindingNavigatorDeleteItem.Visible = x
        CT_ProductosBindingNavigatorSaveItem.Visible = Not x
        BindingNavigatorCancelarItem.Visible = Not x
    End Sub
    Private Sub Bloquear(ByVal x As Boolean)
        Cod_ProductoMaskedTextBox.ReadOnly = x
        DescripcionMaskedTextBox.ReadOnly = x
        PrecioMaskedTextBox.ReadOnly = x
        ExistenciasMaskedTextBox.ReadOnly = x
    End Sub
    Private Sub Actualizar()
        Me.CT_ProductosTableAdapter.Fill(Me.DsProductos.CT_Productos)
        Botones(True)
        Bloquear(True)
        Me.CT_ProductosDataGridView.Enabled = True
        Me.CT_ProductosDataGridView.Focus()
    End Sub
    Private Sub Cod_productoMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles Cod_productoMaskedTextBox.Validating
        If Cod_productoMaskedTextBox.MaskCompleted Then
            e.Cancel = False
            ErrorProvider1.SetError(Cod_productoMaskedTextBox, Nothing)
        Else
            e.Cancel = True
            ErrorProvider1.SetError(Cod_productoMaskedTextBox, "Dato no valido")
        End If
    End Sub
    Private Sub DescripcionMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles DescripcionMaskedTextBox.Validating
        If DescripcionMaskedTextBox.MaskCompleted Then
            e.Cancel = False
            ErrorProvider1.SetError(DescripcionMaskedTextBox, Nothing)
        Else
            e.Cancel = True
            ErrorProvider1.SetError(DescripcionMaskedTextBox, "Dato no valido")
        End If
    End Sub
End Class
```

```

End Sub
Private Sub PrecioMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles PrecioMaskedTextBox.Validating
    If PrecioMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(PrecioMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(PrecioMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub Existencia_fisicaMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles ExistenciasMaskedTextBox.Validating
    If ExistenciasMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(ExistenciasMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(ExistenciasMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub Saldo_invpMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles Saldo_invpMaskedTextBox.Validating
    If Saldo_invpMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(Saldo_invpMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(Saldo_invpMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub btnBuscar_Click(sender As Object, e As EventArgs) Handles btnBuscar.Click
    Dim dato As String = txtDato.Text
    Try
        If cboModo.Items(1) = cboModo.Text Then
            Me.CT_ProductosTableAdapter.FillByProducto(Me.DsProductos.CT_Productos,
dato)
        End If
        If cboModo.Items(2) = cboModo.Text Then
            Me.CT_ProductosTableAdapter.FillByProducto(Me.DsProductos.CT_Productos,
dato & "%")
        End If
    Catch ex As Exception
    End Try
End Sub
Private Sub cboModo_SelectedValueChanged(sender As Object, e As EventArgs) Handles
cboModo.SelectedValueChanged
    If cboModo.Items(0) = cboModo.Text Then
        Me.CT_ProductosTableAdapter.Fill(Me.DsProductos.CT_Productos)
        txtDato.Clear()
    Else
        txtDato.Clear()
        txtDato.Focus()
    End If
End Sub
Private Sub txtDato_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
txtDato.KeyPress
    If e.KeyChar = ChrW(Keys.Enter) Then
        btnBuscar.PerformClick()
    End If
End Sub

```

```

    Private Sub BindingNavigatorAddNewItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorAddNewItem.Click
        Botones(False)
        Bloquear(False)
        Me.CT_ProductosDataGridView.Enabled = False
        ErrorProvider1.Dispose()
        Cod_productoMaskedTextBox.Focus()
    End Sub
    Private Sub BindingNavigatorModificarItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorModificarItem.Click
        Botones(False)
        Bloquear(False)
        Me.CT_ProductosDataGridView.Enabled = False
        ErrorProvider1.Dispose()
        Cod_productoMaskedTextBox.Focus()
    End Sub
    Private Sub BindingNavigatorDeleteItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorDeleteItem.Click
        Botones(False)
        Me.CT_ProductosDataGridView.Enabled = False
    End Sub
    Private Sub BindingNavigatorCancelarItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorCancelarItem.Click
        Actualizar()
    End Sub
    Private Sub Cod_productoMaskedTextBox_KeyPress(sender As Object, e As
KeyPressEventArgs) Handles Cod_productoMaskedTextBox.KeyPress
        If e.KeyChar = Convert.ToChar(Keys.Enter) Then
            e.Handled = True
            SendKeys.Send("{TAB}")
        End If
    End Sub
    Private Sub DescripcionMaskedTextBox_KeyPress(sender As Object, e As
KeyPressEventArgs) Handles DescripcionMaskedTextBox.KeyPress
        If e.KeyChar = Convert.ToChar(Keys.Enter) Then
            e.Handled = True
            SendKeys.Send("{TAB}")
        End If
    End Sub
    Private Sub PrecioMaskedTextBox_KeyPress(sender As Object, e As KeyPressEvent)
Handles PrecioMaskedTextBox.KeyPress
        If e.KeyChar = Convert.ToChar(Keys.Enter) Then
            e.Handled = True
            SendKeys.Send("{TAB}")
        End If
    End Sub
    Private Sub ExistenciasMaskedTextBox_KeyPress(sender As Object, e As
KeyPressEventArgs) Handles ExistenciasMaskedTextBox.KeyPress
        If e.KeyChar = Convert.ToChar(Keys.Enter) Then
            e.Handled = True
            SendKeys.Send("{TAB}")
        End If
    End Sub
End Class

```

ANEXO 0-9: CODIGO FUENTE, INSUMOS, MODULO PRODUCCION

```
Public Class frmInsumos
    Private Sub CT_InsumosBindingNavigatorSaveItem_Click(sender As Object, e As
EventArgs) Handles CT_InsumosBindingNavigatorSaveItem.Click
        Try
            Me.CT_InsumosBindingSource.EndEdit()
            Me.TableAdapterManager.UpdateAll(Me.DsInsumos)
            Actualizar()
            MessageBox.Show("Cambios guardados con exito", "Guardando datos",
MessageBoxButtons.OK, MessageBoxIcon.Information)
        Catch ex As Exception
            If MessageBox.Show("Ocurrio un error en la base de datos" & vbCrLf & "desea
continuar?", "Error al guardar", MessageBoxButtons.YesNo, MessageBoxIcon.Question) =
Windows.Forms.DialogResult.No Then
                Actualizar()
            End If
        End Try
    End Sub
    Private Sub frmInsumos_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Actualizar()
    End Sub
    Private Sub Botones(ByVal x As Boolean)
        BindingNavigatorAddNewItem.Visible = x
        BindingNavigatorModificarItem.Visible = x
        BindingNavigatorDeleteItem.Visible = x
        CT_InsumosBindingNavigatorSaveItem.Visible = Not x
        BindingNavigatorCancelarItem.Visible = Not x
    End Sub
    Private Sub Bloquear(ByVal x As Boolean)
        Cod_insumoMaskedTextBox.ReadOnly = x
        DescripcionMaskedTextBox.ReadOnly = x
        Existencia_insMaskedTextBox.ReadOnly = x
        Saldo_inventarioMaskedTextBox.ReadOnly = x
    End Sub
    Private Sub Actualizar()
        Me.CT_InsumosTableAdapter.Fill(Me.DsInsumos.CT_Insumos)
        Botones(True)
        Bloquear(True)
        Me.CT_InsumosDataGridView.Enabled = True
        Me.CT_InsumosDataGridView.Focus()
    End Sub
    Private Sub Cod_insumoMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles Cod_insumoMaskedTextBox.Validating
        If Cod_insumoMaskedTextBox.MaskCompleted Then
            e.Cancel = False
            ErrorProvider1.SetError(Cod_insumoMaskedTextBox, Nothing)
        Else
            e.Cancel = True
            ErrorProvider1.SetError(Cod_insumoMaskedTextBox, "Dato no valido")
        End If
    End Sub
    Private Sub DescripcionMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles DescripcionMaskedTextBox.Validating
        If DescripcionMaskedTextBox.MaskCompleted Then
            e.Cancel = False
            ErrorProvider1.SetError(DescripcionMaskedTextBox, Nothing)
        Else
            e.Cancel = True
            ErrorProvider1.SetError(DescripcionMaskedTextBox, "Dato no valido")
        End If
    End Sub
End Class
```

```

End Sub
Private Sub PrecioMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles PrecioMaskedTextBox.Validating
    If PrecioMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(PrecioMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(PrecioMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub Ficha_tecnicaMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles Ficha_tecnicaMaskedTextBox.Validating
    If Ficha_tecnicaMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(Ficha_tecnicaMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(Ficha_tecnicaMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub Existencia_fisicaMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles Existencia_insMaskedTextBox.Validating
    If Existencia_insMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(Existencia_insMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(Existencia_insMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub Saldo_inviMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles Saldo_inventarioMaskedTextBox.Validating
    If Saldo_inventarioMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(Saldo_inventarioMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(Saldo_inventarioMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub Cod_insumoMaskedTextBox_KeyPress(sender As Object, e As
KeyPressEventArgs) Handles Cod_insumoMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub DescripcionMaskedTextBox_KeyPress(sender As Object, e As
KeyPressEventArgs) Handles DescripcionMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub PrecioMaskedTextBox_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles PrecioMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If

```

```

End Sub
Private Sub Ficha_tecnicaMaskedTextBox_KeyPress(sender As Object, e As
KeyPressEventArgs) Handles Ficha_tecnicaMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub Existencia_fisicaMaskedTextBox_KeyPress(sender As Object, e As
KeyPressEventArgs) Handles Existencia_insMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub btnBuscar_Click(sender As Object, e As EventArgs) Handles btnBuscar.Click
    Dim dato As String = txtDato.Text
    Try
        If cboModo.Items(1) = cboModo.Text Then
            Me.CT_InsumosTableAdapter.FillByCodigo(Me.DsInsumos.CT_Insumos, dato)
        End If
        If cboModo.Items(2) = cboModo.Text Then
            Me.CT_InsumosTableAdapter.FillByInsumo(Me.DsInsumos.CT_Insumos, dato &
"%")
        End If
    Catch ex As Exception
    End Try
End Sub
Private Sub cboModo_SelectedValueChanged(sender As Object, e As EventArgs) Handles
cboModo.SelectedValueChanged
    If cboModo.Items(0) = cboModo.Text Then
        Me.CT_InsumosTableAdapter.Fill(Me.DsInsumos.CT_Insumos)
        txtDato.Clear()
    Else
        txtDato.Clear()
        txtDato.Focus()
    End If
End Sub
Private Sub txtDato_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
txtDato.KeyPress
    If e.KeyChar = ChrW(Keys.Enter) Then
        btnBuscar.PerformClick()
    End If
End Sub
Private Sub BindingNavigatorAddNewItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorAddNewItem.Click
    Botones(False)
    Bloquear(False)
    Me.CT_InsumosDataGridView.Enabled = False
    ErrorProvider1.Dispose()
    Cod_insumoMaskedTextBox.Focus()
End Sub
Private Sub BindingNavigatorModificarItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorModificarItem.Click
    Botones(False)
    Bloquear(False)
    Me.CT_InsumosDataGridView.Enabled = False
    ErrorProvider1.Dispose()
    Cod_insumoMaskedTextBox.Focus()
End Sub

```

```
Private Sub BindingNavigatorDeleteItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorDeleteItem.Click
    Botones(False)
    Me.CT_InsumosDataGridView.Enabled = False
End Sub
Private Sub BindingNavigatorCancelarItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorCancelarItem.Click
    Actualizar()
End Sub
End Class
```



ANEXO 0-10: CODIGO FUENTE, PROGRAMA MAESTRO DE PRODUCCION, MODULO PRODUCCION

```
Public Class frmSimplex
    Dim prex1 As Decimal = 38
    Dim prex2 As Decimal = 41
    Dim prex3 As Decimal = 35
    Dim prex4 As Decimal = 36
    Dim prex5 As Decimal = 47
    Dim chod1 As Decimal = 21
    Dim chod2 As Decimal = 21
    Dim chod3 As Decimal = 21
    Dim chod4 As Decimal = 33
    Dim chod5 As Decimal = 31
    Dim chon1 As Decimal = chod1 * 1.3
    Dim chon2 As Decimal = chod2 * 1.3
    Dim chon3 As Decimal = chod3 * 1.3
    Dim chon4 As Decimal = chod4 * 1.3
    Dim chon5 As Decimal = chod5 * 1.3
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        txtsold1.Text = Format(prex1 - chod1, "0,00")
        txtsold2.Text = Format(prex2 - chod2, "0,00")
        txtsold3.Text = Format(prex3 - chod3, "0,00")
        txtsold4.Text = Format(prex4 - chod4, "0,00")
        txtsold5.Text = Format(prex5 - chod5, "0,00")
        txtsoln1.Text = Format(prex1 - chon1, "0,00")
        txtsoln2.Text = Format(prex2 - chon2, "0,00")
        txtsoln3.Text = Format(prex3 - chon3, "0,00")
        txtsoln4.Text = Format(prex4 - chon4, "0,00")
        txtsoln5.Text = Format(prex5 - chon5, "0,00")
        Dim objExcel As Excel.Application
        Dim xLibro As Excel.Workbook
        objExcel = New Excel.Application
        xLibro = objExcel.Workbooks.Open("C:\Users\Victor\Desktop\Libro1.xlsm")
        xLibro.Application.Cells(6, 15) = txttdh1.Text * 80
        xLibro.Application.Cells(7, 15) = txttdh2.Text * 80
        xLibro.Application.Cells(8, 15) = txttdh3.Text * 80
        xLibro.Application.Cells(9, 15) = txttdh4.Text * 80
        xLibro.Application.Cells(10, 15) = txttdh5.Text * 80
        xLibro.Application.Cells(11, 15) = txttdh6.Text * 80
        xLibro.Application.Cells(12, 15) = txttdh7.Text * 80
        xLibro.Application.Cells(13, 15) = txttdh8.Text * 80
        xLibro.Application.Cells(14, 15) = txttdh9.Text * 80
        xLibro.Application.Cells(15, 15) = txttdh10.Text * 80
        xLibro.Application.Cells(16, 15) = txttdh11.Text * 80
        xLibro.Application.Cells(17, 15) = txttdh12.Text * 80
        xLibro.Application.Cells(2, 3) = txtsold1.Text
        xLibro.Application.Cells(2, 4) = txtsold2.Text
        xLibro.Application.Cells(2, 5) = txtsold3.Text
        xLibro.Application.Cells(2, 6) = txtsold4.Text
        xLibro.Application.Cells(2, 7) = txtsold5.Text
        xLibro.Application.Cells(2, 8) = txtsoln1.Text
        xLibro.Application.Cells(2, 9) = txtsoln2.Text
        xLibro.Application.Cells(2, 10) = txtsoln3.Text
        xLibro.Application.Cells(2, 11) = txtsoln4.Text
        xLibro.Application.Cells(2, 12) = txtsoln5.Text
        xLibro.Close(SaveChanges:=True)
        xLibro = Nothing
        objExcel.Quit()
        objExcel = Nothing
        Dim ds As New DataSet()
```

```

    Dim connectionString2 As String = "Provider=Microsoft.ACE.OLEDB.12.0;" & "Data
Source=C:\Users\Victor\Desktop\Libro1.xlsm;" & "Extended Properties=Excel 12.0"
    Dim excelData As New OleDbDataAdapter("SELECT Descripcion,Valores FROM [Hoja2$]
", connectionString2)
    excelData.TableMappings.Add("Table", "ExcelSheet")
    excelData.Fill(ds)
    Me.DataGridView1.DataSource = ds.Tables(0)
    Me.Refresh()
End Sub
Private Sub txtchod1_TextChanged(sender As Object, e As EventArgs)
    chod1 = txtchod1.Text
End Sub
Private Sub txtpre1_TextChanged(sender As Object, e As EventArgs)
    prex1 = txtpre1.Text
End Sub
Private Sub txtchon1_TextChanged(sender As Object, e As EventArgs)
    chon1 = txtchon1.Text
End Sub
Private Sub txtchod2_TextChanged(sender As Object, e As EventArgs)
    chod2 = txtchod2.Text
End Sub
Private Sub txtchod3_TextChanged(sender As Object, e As EventArgs)
    chod3 = txtchod3.Text
End Sub
Private Sub txtchod4_TextChanged(sender As Object, e As EventArgs)
    chod4 = txtchod4.Text
End Sub
Private Sub txtchod5_TextChanged(sender As Object, e As EventArgs)
    chod5 = txtchod5.Text
End Sub
Private Sub txtchon2_TextChanged(sender As Object, e As EventArgs)
    chon2 = txtchon2.Text
End Sub
Private Sub txtchon3_TextChanged(sender As Object, e As EventArgs)
    chon3 = txtchon3.Text
End Sub
Private Sub txtchon4_TextChanged(sender As Object, e As EventArgs)
    chon4 = txtchon4.Text
End Sub
Private Sub txtchon5_TextChanged(sender As Object, e As EventArgs)
    chon5 = txtchon5.Text
End Sub
Private Sub txtpre2_TextChanged(sender As Object, e As EventArgs)
    prex2 = txtpre2.Text
End Sub
Private Sub txtpre3_TextChanged(sender As Object, e As EventArgs)
    prex3 = txtpre3.Text
End Sub
Private Sub txtpre4_TextChanged(sender As Object, e As EventArgs)
    prex4 = txtpre4.Text
End Sub
Private Sub txtpre5_TextChanged(sender As Object, e As EventArgs)
    prex5 = txtpre5.Text
End Sub
Private Sub frmSimplex_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    txtdh1.Text = 22
    txtdh2.Text = 19
    txtdh3.Text = 21
    txtdh4.Text = 21
    txtdh5.Text = 22
    txtdh6.Text = 20

```

```

txtdh7.Text = 21
txtdh8.Text = 22
txtdh9.Text = 23
txtdh10.Text = 22
txtdh11.Text = 22
txtdh12.Text = 23
txtchod1.Text = chod1
txtchod2.Text = chod2
txtchod3.Text = chod3
txtchod4.Text = chod4
txtchod5.Text = chod5
txtchon1.Text = chon1
txtchon2.Text = chon2
txtchon3.Text = chon3
txtchon4.Text = chon4
txtchon5.Text = chon5
txtpre1.Text = prex1
txtpre2.Text = prex2
txtpre3.Text = prex3
txtpre4.Text = prex4
txtpre5.Text = prex5
Dim ds1 As New DataSet()
Dim connectionString As String = "Provider=Microsoft.ACE.OLEDB.12.0;" & "Data
Source=C:\Users\Victor\Desktop\pronostico.xlsx;" & "Extended Properties=Excel 12.0"
Dim excelData1 As New OleDbDataAdapter("SELECT * FROM [RESUMEN$] ",
connectionString)
excelData1.TableMappings.Add("Table", "ExcelSheet")
excelData1.Fill(ds1)
Me.DataGridView3.DataSource = ds1.Tables(0)
Me.Refresh()
Me.DataGridView3.ClearSelection()
Me.DataGridView3.CurrentCell = Me.DataGridView3.Rows(Me.DataGridView3.RowCount -
(14 - Me.DateTimePicker1.Value.Month)).Cells(0)
Dim ds2 As New DataSet()
Dim excelData2 As New OleDbDataAdapter("SELECT * FROM [Datos$] ",
connectionString)
excelData2.TableMappings.Add("Table", "ExcelSheet")
excelData2.Fill(ds2)
Me.DataGridView2.DataSource = ds2.Tables(0)
Me.Refresh()
Dim ds3 As New DataSet()
Dim exceldata3 As New OleDbDataAdapter("SELECT * FROM [Hoja3$] ",
connectionString)
exceldata3.TableMappings.Add("Table", "ExcelSheet")
exceldata3.Fill(ds3)
Me.DataGridView4.DataSource = ds3.Tables(0)
Me.Refresh()
End Sub
End Class

```

ANEXO 0-11: CODIGO FUENTE, GESTION DE CLIENTES, MODULO VENTAS

```
Public Class frmClientes
    Private Sub CT_ClientesBindingNavigatorSaveItem_Click(sender As Object, e As
EventArgs) Handles CT_ClientesBindingNavigatorSaveItem.Click
        Try
            Me.CT_ClientesBindingSource.EndEdit()
            Me.TableAdapterManager.UpdateAll(Me.Ds_Clientes)
            Actualizar()
            MessageBox.Show("Cambios guardados con exito", "Guardando datos",
MessageBoxButtons.OK, MessageBoxIcon.Information)
        Catch ex As Exception
            If MessageBox.Show("Ocurrio un error en la base de datos" & vbCrLf & "desea
continuar?", "Error al guardar", MessageBoxButtons.YesNo, MessageBoxIcon.Question) =
Windows.Forms.DialogResult.No Then
                Actualizar()
            End If
        End Try
    End Sub
    Private Sub frmClientes_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Actualizar()
    End Sub
    Private Sub Botones(ByVal x As Boolean)
        BindingNavigatorAddNewItem.Visible = x
        BindingNavigatorModificarItem.Visible = x
        BindingNavigatorDeleteItem.Visible = x
        CT_ClientesBindingNavigatorSaveItem.Visible = Not x
        BindingNavigatorCancelarItem.Visible = Not x
    End Sub
    Private Sub Bloquear(ByVal x As Boolean)
        Cod_clienteMaskedTextBox.ReadOnly = x
        EmpresaMaskedTextBox.ReadOnly = x
        DireccionMaskedTextBox.ReadOnly = x
        TelefonoMaskedTextBox.ReadOnly = x
        CI_NITMaskedTextBox.ReadOnly = x
        Saldo_cpcMaskedTextBox.ReadOnly = x
    End Sub
    Private Sub Actualizar()
        Me.CT_ClientesTableAdapter.Fill(Me.Ds_Clientes.CT_Clientes)
        Botones(True)
        Bloquear(True)
        Me.CT_ClientesDataGridView.Enabled = True
        Me.CT_ClientesDataGridView.Focus()
    End Sub
    Private Sub Cod_clienteMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles Cod_clienteMaskedTextBox.Validating
        If Cod_clienteMaskedTextBox.MaskCompleted Then
            e.Cancel = False
            ErrorProvider1.SetError(Cod_clienteMaskedTextBox, Nothing)
        Else
            e.Cancel = True
            ErrorProvider1.SetError(Cod_clienteMaskedTextBox, "Dato no valido")
        End If
    End Sub
    Private Sub EmpresaMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles EmpresaMaskedTextBox.Validating
        If EmpresaMaskedTextBox.MaskCompleted Then
            e.Cancel = False
            ErrorProvider1.SetError(EmpresaMaskedTextBox, Nothing)
        Else
            e.Cancel = True
        End If
    End Sub
End Class
```

```

        ErrorProvider1.SetError(EmpresaMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub DireccionMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles DireccionMaskedTextBox.Validating
    If DireccionMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(DireccionMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(DireccionMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub TelefonoMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles TelefonoMaskedTextBox.Validating
    If TelefonoMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(TelefonoMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(TelefonoMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub CI_NITMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles CI_NITMaskedTextBox.Validating
    If CI_NITMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(CI_NITMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(CI_NITMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub Saldo_cpcMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles Saldo_cpcMaskedTextBox.Validating
    If Saldo_cpcMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(Saldo_cpcMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(Saldo_cpcMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub Cod_clienteMaskedTextBox_KeyPress(sender As Object, e As
KeyPressEventArgs) Handles Cod_clienteMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub EmpresaMaskedTextBox_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles EmpresaMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub DireccionMaskedTextBox_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles DireccionMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
    End If
End Sub

```

```

        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub TelefonoMaskedTextBox_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles TelefonoMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub CI_NITMaskedTextBox_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles CI_NITMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub btnBuscar_Click(sender As Object, e As EventArgs) Handles btnBuscar.Click
    Dim dato As String = txtDato.Text
    Try
        If cboModo.Items(1) = cboModo.Text Then
            Me.CT_ClientesTableAdapter.FillByCodigo(Me.Ds_Clientes.CT_Clientes, dato)
        End If
        If cboModo.Items(2) = cboModo.Text Then
            Me.CT_ClientesTableAdapter.FillByEmpresa(Me.Ds_Clientes.CT_Clientes, dato
& "%")
        End If
        If cboModo.Items(3) = cboModo.Text Then
            Me.CT_ClientesTableAdapter.FillByCI_NIT(Me.Ds_Clientes.CT_Clientes, dato)
        End If
    Catch ex As Exception
    End Try
End Sub
Private Sub cboModo_SelectedValueChanged(sender As Object, e As EventArgs) Handles
cboModo.SelectedValueChanged
    If cboModo.Items(0) = cboModo.Text Then
        Me.CT_ClientesTableAdapter.Fill(Me.Ds_Clientes.CT_Clientes)
        txtDato.Clear()
    Else
        txtDato.Clear()
        txtDato.Focus()
    End If
End Sub
Private Sub txtDato_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
txtDato.KeyPress
    If e.KeyChar = ChrW(Keys.Enter) Then
        btnBuscar.PerformClick()
    End If
End Sub
Private Sub BindingNavigatorAddNewItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorAddNewItem.Click
    Botones(False)
    Bloquear(False)
    Me.CT_ClientesDataGridView.Enabled = False
    ErrorProvider1.Dispose()
    Cod_clienteMaskedTextBox.Focus()
End Sub
Private Sub BindingNavigatorModificarItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorModificarItem.Click
    Botones(False)
    Bloquear(False)

```

```
Me.CT_ClientesDataGridView.Enabled = False
ErrorProvider1.Dispose()
Cod_clienteMaskedTextBox.Focus()
End Sub
Private Sub BindingNavigatorDeleteItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorDeleteItem.Click
Botones(False)
Me.CT_ClientesDataGridView.Enabled = False
End Sub
Private Sub BindingNavigatorCancelarItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorCancelarItem.Click
Actualizar()
End Sub
End Class
```



ANEXO 0-12: CODIGO FUENTE, VENTAS, MODULO VENTAS

```
Public Class frmVenta
    Private Sub frmVenta_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        textoconta = 0
        Dim oadapterfac As New dsFacturaTableAdapters.CT_FacturasTableAdapter
        confactura = oadapterfac.ScalarQueryContFac()
        txtNoFac.Text = confactura + 1
    End Sub
    Private Sub txtPre1_TextChanged(sender As Object, e As EventArgs) Handles
txtPre1.TextChanged
        Try
            pr1 = txtPre1.Text
            txtTot1.Text = cant1 * pr1
            f1 = txtTot1.Text
            txtTotalPag.Text = f1 + f2 + f3 + f4 + f5 + f6
        Catch ex As Exception
        End Try
    End Sub
    Private Sub txtPre2_TextChanged(sender As Object, e As EventArgs) Handles
txtPre2.TextChanged
        Try
            pr2 = txtPre2.Text
            txtTot2.Text = cant2 * pr2
            f2 = txtTot2.Text
            txtTotalPag.Text = f1 + f2 + f3 + f4 + f5 + f6
        Catch ex As Exception
        End Try
    End Sub
    Private Sub txtPre3_TextChanged(sender As Object, e As EventArgs) Handles
txtPre3.TextChanged
        Try
            pr3 = txtPre3.Text
            txtTot3.Text = cant3 * pr3
            f3 = txtTot3.Text
            txtTotalPag.Text = f1 + f2 + f3 + f4 + f5 + f6
        Catch ex As Exception
        End Try
    End Sub
    Private Sub txtPre4_TextChanged(sender As Object, e As EventArgs) Handles
txtPre4.TextChanged
        Try
            pr4 = txtPre4.Text
            txtTot4.Text = cant4 * pr4
            f4 = txtTot4.Text
            txtTotalPag.Text = f1 + f2 + f3 + f4 + f5 + f6
        Catch ex As Exception
        End Try
    End Sub
    Private Sub txtPre5_TextChanged(sender As Object, e As EventArgs) Handles
txtPre5.TextChanged
        Try
            pr5 = txtPre5.Text
            txtTot5.Text = cant5 * pr5
            f5 = txtTot5.Text
            txtTotalPag.Text = f1 + f2 + f3 + f4 + f5 + f6
        Catch ex As Exception
        End Try
    End Sub
End Class
```

```

Private Sub txtPre6_TextChanged(sender As Object, e As EventArgs) Handles
txtPre6.TextChanged
    Try
        pr6 = txtPre6.Text
        txtTot6.Text = cant6 * pr6
        f6 = txtTot6.Text
        txtTotalPag.Text = f1 + f2 + f3 + f4 + f5 + f6
    Catch ex As Exception
    End Try
End Sub
Private Sub txtEfectivo_TextChanged(sender As Object, e As EventArgs) Handles
txtEfectivo.TextChanged
    Try
        totpag = txtTotalPag.Text
        efect = txtEfectivo.Text
        txtCambio.Text = efect - totpag
    Catch ex As Exception
    End Try
End Sub
Private Sub btnAceptar_Click(sender As Object, e As EventArgs) Handles
btnAceptar.Click
    Dim odaptersalpro As New dsProductosTableAdapters.CT_ProductosTableAdapter
    If txtex1.Text = "" Then
        MessageBox.Show("Introduzca los datos necesarios para proceder", "Datos
Insuficientes", MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
    Else
        If txtex2.Text = "" Then
            odaptersalpro.UpdateQueryModExistencias(dif1, txtcod1.Text)
        Else
            If txtex3.Text = "" Then
                odaptersalpro.UpdateQueryModExistencias(dif1, txtcod1.Text)
                odaptersalpro.UpdateQueryModExistencias(dif2, txtcod2.Text)
            Else
                If txtex4.Text = "" Then
                    odaptersalpro.UpdateQueryModExistencias(dif1, txtcod1.Text)
                    odaptersalpro.UpdateQueryModExistencias(dif2, txtcod2.Text)
                    odaptersalpro.UpdateQueryModExistencias(dif3, txtcod3.Text)
                Else
                    If txtex5.Text = "" Then
                        odaptersalpro.UpdateQueryModExistencias(dif1, txtcod1.Text)
                        odaptersalpro.UpdateQueryModExistencias(dif2, txtcod2.Text)
                        odaptersalpro.UpdateQueryModExistencias(dif3, txtcod3.Text)
                        odaptersalpro.UpdateQueryModExistencias(dif4, txtcod4.Text)
                    Else
                        If txtex6.Text = "" Then
                            odaptersalpro.UpdateQueryModExistencias(dif1,
txtcod1.Text)
                                odaptersalpro.UpdateQueryModExistencias(dif2,
txtcod2.Text)
                                odaptersalpro.UpdateQueryModExistencias(dif3,
txtcod3.Text)
                                odaptersalpro.UpdateQueryModExistencias(dif4,
txtcod4.Text)
                                odaptersalpro.UpdateQueryModExistencias(dif5,
txtcod5.Text)
                            Else
                                odaptersalpro.UpdateQueryModExistencias(dif1,
txtcod1.Text)
                                    odaptersalpro.UpdateQueryModExistencias(dif2,
txtcod2.Text)

```

```

txtcod3.Text)                odaptersalpro.UpdateQueryModExistencias(dif3,
txtcod4.Text)                odaptersalpro.UpdateQueryModExistencias(dif4,
txtcod5.Text)                odaptersalpro.UpdateQueryModExistencias(dif5,
txtcod6.Text)                odaptersalpro.UpdateQueryModExistencias(dif6,

                                End If
                                End If
                                End If
                                End If
                                End If
frmFactura.txtNombre.Text = txtCliente.Text
frmFactura.txtNIT_CI.Text = txtCI_NIT.Text
frmFactura.txtCant1.Text = txtCant1.Text
frmFactura.txtCant2.Text = txtCant2.Text
frmFactura.txtCant3.Text = txtCant3.Text
frmFactura.txtCant4.Text = txtCant4.Text
frmFactura.txtCant5.Text = txtCant5.Text
frmFactura.txtCant6.Text = txtCant6.Text
frmFactura.txtCon1.Text = txtCon1.Text
frmFactura.txtCon2.Text = txtCon2.Text
frmFactura.txtCon3.Text = txtCon3.Text
frmFactura.txtCon4.Text = txtCon4.Text
frmFactura.txtCon5.Text = txtCon5.Text
frmFactura.txtCon6.Text = txtCon6.Text
frmFactura.txtPre1.Text = txtPre1.Text
frmFactura.txtPre2.Text = txtPre2.Text
frmFactura.txtPre3.Text = txtPre3.Text
frmFactura.txtPre4.Text = txtPre4.Text
frmFactura.txtPre5.Text = txtPre5.Text
frmFactura.txtPre6.Text = txtPre6.Text
frmFactura.txtTot1.Text = txtTot1.Text
frmFactura.txtTot2.Text = txtTot2.Text
frmFactura.txtTot3.Text = txtTot3.Text
frmFactura.txtTot4.Text = txtTot4.Text
frmFactura.txtTot5.Text = txtTot5.Text
frmFactura.txtTot6.Text = txtTot6.Text
frmFactura.txtTotalPag.Text = txtTotalPag.Text
frmFactura.txtEfectivo.Text = txtEfectivo.Text
frmFactura.txtCambio.Text = txtCambio.Text
frmFactura.txtNoFactura.Text = txtNoFac.Text
frmFactura.txtFecha.Text = DateTimePicker1.Value
Dim adapterfac As New dsFacturaTableAdapters.CT_FacturasTableAdapter
adapterfac.InsertQueryFactura(txtNoFac.Text, txtCliente.Text, txtCI_NIT.Text,
txtTotalPag.Text, DateTimePicker1.Value, "")
txtCliente.Text = ""
txtCI_NIT.Text = ""
txtCant1.Text = ""
txtCant2.Text = ""
txtCant3.Text = ""
txtCant4.Text = ""
txtCant5.Text = ""
txtCant6.Text = ""
txtCon1.Text = ""
txtCon2.Text = ""
txtCon3.Text = ""
txtCon4.Text = ""
txtCon5.Text = ""

```

```

        txtCon6.Text = ""
        txtPre1.Text = ""
        txtTot1.Text = ""
        txtTotalPag.Text = ""
        txtEfectivo.Text = ""
        txtCambio.Text = ""
        frmFactura.ShowDialog()
    End Sub
    Private Sub txtCon1_DoubleClick(sender As Object, e As EventArgs) Handles
txtCon1.DoubleClick
        textoconta = 1
        frmBuscaProducto.ShowDialog()
    End Sub
    Private Sub txtCon2_DoubleClick(sender As Object, e As EventArgs) Handles
txtCon2.DoubleClick
        textoconta = 2
        frmBuscaProducto.ShowDialog()
    End Sub
    Private Sub txtCon3_DoubleClick(sender As Object, e As EventArgs) Handles
txtCon3.DoubleClick
        textoconta = 3
        frmBuscaProducto.ShowDialog()
    End Sub
    Private Sub txtCon4_DoubleClick(sender As Object, e As EventArgs) Handles
txtCon4.DoubleClick
        textoconta = 4
        frmBuscaProducto.ShowDialog()
    End Sub
    Private Sub txtCon5_DoubleClick(sender As Object, e As EventArgs) Handles
txtCon5.DoubleClick
        textoconta = 5
        frmBuscaProducto.ShowDialog()
    End Sub
    Private Sub txtCon6_DoubleClick(sender As Object, e As EventArgs) Handles
txtCon6.DoubleClick
        textoconta = 6
        frmBuscaProducto.ShowDialog()
    End Sub
    Private Sub txtCliente_KeyPress(sender As Object, e As KeyEventArgs) Handles
txtCliente.KeyPress
        If e.KeyChar = Convert.ToChar(Keys.Enter) Then
            e.Handled = True
            SendKeys.Send("{TAB}")
        End If
    End Sub
    Private Sub txtCI_NIT_KeyPress(sender As Object, e As KeyEventArgs) Handles
txtCI_NIT.KeyPress
        If e.KeyChar = Convert.ToChar(Keys.Enter) Then
            e.Handled = True
            SendKeys.Send("{TAB}")
        End If
    End Sub
    Private Sub txtCant1_KeyPress(sender As Object, e As KeyEventArgs) Handles
txtCant1.KeyPress
        If e.KeyChar = Convert.ToChar(Keys.Enter) Then
            e.Handled = True
            SendKeys.Send("{TAB}")
        End If
    End Sub
    Private Sub txtCant2_KeyPress(sender As Object, e As KeyEventArgs) Handles
txtCant2.KeyPress

```

```

        If e.KeyChar = Convert.ToChar(Keys.Enter) Then
            e.Handled = True
            SendKeys.Send("{TAB}")
        End If
    End Sub
    Private Sub txtCant3_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
txtCant3.KeyPress
        If e.KeyChar = Convert.ToChar(Keys.Enter) Then
            e.Handled = True
            SendKeys.Send("{TAB}")
        End If
    End Sub
    Private Sub txtCant4_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
txtCant4.KeyPress
        If e.KeyChar = Convert.ToChar(Keys.Enter) Then
            e.Handled = True
            SendKeys.Send("{TAB}")
        End If
    End Sub
    Private Sub txtCant5_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
txtCant5.KeyPress
        If e.KeyChar = Convert.ToChar(Keys.Enter) Then
            e.Handled = True
            SendKeys.Send("{TAB}")
        End If
    End Sub
    Private Sub txtCant6_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
txtCant6.KeyPress
        If e.KeyChar = Convert.ToChar(Keys.Enter) Then
            e.Handled = True
            SendKeys.Send("{TAB}")
        End If
    End Sub
    Private Sub txtEfectivo_KeyPress(sender As Object, e As KeyPressEventArgs) Handles
txtEfectivo.KeyPress
        If e.KeyChar = Convert.ToChar(Keys.Enter) Then
            e.Handled = True
            SendKeys.Send("{TAB}")
        End If
    End Sub
    Dim cant1, cant2, cant3, cant4, cant5, cant6, exi1, exi2, exi3, exi4, exi5, exi6 As
Integer
    Dim dif1, dif2, dif3, dif4, dif5, dif6 As Integer
    Private Sub txtCant1_TextChanged(sender As Object, e As EventArgs) Handles
txtCant1.TextChanged
        Try
            cant1 = txtCant1.Text
        Catch ex As Exception
        End Try
    End Sub
    Private Sub txtCant2_TextChanged(sender As Object, e As EventArgs) Handles
txtCant2.TextChanged
        Try
            cant2 = txtCant2.Text
        Catch ex As Exception
        End Try
    End Sub
    Private Sub txtCant3_TextChanged(sender As Object, e As EventArgs) Handles
txtCant3.TextChanged
        Try
            cant3 = txtCant3.Text

```

```

        Catch ex As Exception
        End Try
    End Sub
    Private Sub txtCant4_TextChanged(sender As Object, e As EventArgs) Handles
txtCant4.TextChanged
        Try
            cant4 = txtCant4.Text
        Catch ex As Exception
        End Try
    End Sub
    Private Sub txtCant5_TextChanged(sender As Object, e As EventArgs) Handles
txtCant5.TextChanged
        Try
            cant5 = txtCant5.Text
        Catch ex As Exception
        End Try
    End Sub
    Private Sub txtCant6_TextChanged(sender As Object, e As EventArgs) Handles
txtCant6.TextChanged
        Try
            cant6 = txtCant6.Text
        Catch ex As Exception
        End Try
    End Sub
    Private Sub txtex1_TextChanged(sender As Object, e As EventArgs) Handles
txtex1.TextChanged
        Try
            exi1 = txtex1.Text
            If cant1 > exi1 Then
                MessageBox.Show("No hay suficiente cantidad del producto en inventario",
                "Inventario Insuficiente", MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
                txtCant1.Text = ""
                txtCon1.Text = ""
                txtPre1.Text = ""
                txtTot1.Text = ""
            Else
                dif1 = exi1 - cant1
            End If
        Catch ex As Exception
        End Try
    End Sub
    Private Sub txtex2_TextChanged(sender As Object, e As EventArgs) Handles
txtex2.TextChanged
        Try
            exi2 = txtex2.Text
            If cant2 > exi2 Then
                MessageBox.Show("No hay suficiente cantidad del producto en inventario",
                "Inventario Insuficiente", MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
                txtCant2.Text = ""
                txtCon2.Text = ""
                txtPre2.Text = ""
                txtTot2.Text = ""
            Else
                dif2 = exi2 - cant2
            End If
        Catch ex As Exception
        End Try
    End Sub
    Private Sub txtex3_TextChanged(sender As Object, e As EventArgs) Handles
txtex3.TextChanged
        Try

```

```

        exi3 = txtex3.Text
        If cant3 > exi3 Then
            MessageBox.Show("No hay suficiente cantidad del producto en inventario",
                "Inventario Insuficiente", MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            txtCant3.Text = ""
            txtCon3.Text = ""
            txtPre3.Text = ""
            txtTot3.Text = ""
        Else
            dif3 = exi3 - cant3
        End If
    Catch ex As Exception
    End Try
End Sub
Private Sub txtex4_TextChanged(sender As Object, e As EventArgs) Handles
txtex4.TextChanged
    Try
        exi4 = txtex4.Text
        If cant4 > exi4 Then
            MessageBox.Show("No hay suficiente cantidad del producto en inventario",
                "Inventario Insuficiente", MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            txtCant4.Text = ""
            txtCon4.Text = ""
            txtPre4.Text = ""
            txtTot4.Text = ""
        Else
            dif4 = exi4 - cant4
        End If
    Catch ex As Exception
    End Try
End Sub
Private Sub txtex5_TextChanged(sender As Object, e As EventArgs) Handles
txtex5.TextChanged
    Try
        exi5 = txtex5.Text
        If cant5 > exi5 Then
            MessageBox.Show("No hay suficiente cantidad del producto en inventario",
                "Inventario Insuficiente", MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            txtCant5.Text = ""
            txtCon5.Text = ""
            txtPre5.Text = ""
            txtTot5.Text = ""
        Else
            dif5 = exi5 - cant5
        End If
    Catch ex As Exception
    End Try
End Sub
Private Sub txtex6_TextChanged(sender As Object, e As EventArgs) Handles
txtex6.TextChanged
    Try
        exi6 = txtex6.Text
        If cant6 > exi6 Then
            MessageBox.Show("No hay suficiente cantidad del producto en inventario",
                "Inventario Insuficiente", MessageBoxButtons.OK, MessageBoxIcon.Exclamation)
            txtCant6.Text = ""
            txtCon6.Text = ""
            txtPre6.Text = ""
            txtTot6.Text = ""
        Else
            dif6 = exi6 - cant6
        End If
    Catch ex As Exception
    End Try
End Sub

```

```
End If
Catch ex As Exception
End Try
End Sub
End Class
```



ANEXO 0-13: CODIGO FUENTE, REGISTROS CONTABLES, MODULO CONTABILIDAD

```
Public Class frmContabilidad
    Private Sub frmContabilidad_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Me.CT_TransaccionesTableAdapter.Fill(Me.DsTransacciones.CT_Transacciones)
        textoconta = 0
        idcuenta = 0
        contador = 0
        Dim oadaptertran As New dsTransaccionesTableAdapters.CT_TransaccionesTableAdapter
        contador = oadaptertran.ScalarQueryContador()
        txtcodtran.Text = contador
        t = TextBox1.Text
        txtdescripcion.Focus()
    End Sub
    Private Sub TextB1_DoubleClick(sender As Object, e As EventArgs) Handles TextB1.DoubleClick
        textoconta = 1
        idcuenta = 1
        frmBuscaCuenta.ShowDialog()
    End Sub
    Private Sub TextB2_DoubleClick(sender As Object, e As EventArgs) Handles TextB2.DoubleClick
        textoconta = 2
        idcuenta = 2
        frmBuscaCuenta.ShowDialog()
    End Sub
    Private Sub TextB3_DoubleClick(sender As Object, e As EventArgs) Handles TextB3.DoubleClick
        textoconta = 3
        idcuenta = 3
        frmBuscaCuenta.ShowDialog()
    End Sub
    Private Sub TextB4_DoubleClick(sender As Object, e As EventArgs) Handles TextB4.DoubleClick
        textoconta = 4
        idcuenta = 4
        frmBuscaCuenta.ShowDialog()
    End Sub
    Private Sub TextB5_DoubleClick(sender As Object, e As EventArgs) Handles TextB5.DoubleClick
        textoconta = 5
        idcuenta = 5
        frmBuscaCuenta.ShowDialog()
    End Sub
    Private Sub TextB6_DoubleClick(sender As Object, e As EventArgs) Handles TextB6.DoubleClick
        textoconta = 6
        idcuenta = 6
        frmBuscaCuenta.ShowDialog()
    End Sub
    Private Sub TextB7_DoubleClick(sender As Object, e As EventArgs) Handles TextB7.DoubleClick
        textoconta = 7
        idcuenta = 7
        frmBuscaCuenta.ShowDialog()
    End Sub
    Private Sub TextB8_DoubleClick(sender As Object, e As EventArgs) Handles TextB8.DoubleClick
        textoconta = 8
        idcuenta = 8
    End Sub
End Class
```

```

        frmBuscaCuenta.ShowDialog()
    End Sub
    Private Sub TextB11_DoubleClick(sender As Object, e As EventArgs) Handles
TextB11.DoubleClick
        textoconta = 11
        idcuenta = 1
        frmBuscaCuenta.ShowDialog()
    End Sub
    Private Sub TextB12_DoubleClick(sender As Object, e As EventArgs) Handles
TextB12.DoubleClick
        textoconta = 12
        idcuenta = 2
        frmBuscaCuenta.ShowDialog()
    End Sub
    Private Sub TextB13_DoubleClick(sender As Object, e As EventArgs) Handles
TextB13.DoubleClick
        textoconta = 13
        idcuenta = 3
        frmBuscaCuenta.ShowDialog()
    End Sub
    Private Sub TextB14_DoubleClick(sender As Object, e As EventArgs) Handles
TextB14.DoubleClick
        textoconta = 14
        idcuenta = 4
        frmBuscaCuenta.ShowDialog()
    End Sub
    Private Sub TextB15_DoubleClick(sender As Object, e As EventArgs) Handles
TextB15.DoubleClick
        textoconta = 15
        idcuenta = 5
        frmBuscaCuenta.ShowDialog()
    End Sub
    Private Sub TextB16_DoubleClick(sender As Object, e As EventArgs) Handles
TextB16.DoubleClick
        textoconta = 16
        idcuenta = 6
        frmBuscaCuenta.ShowDialog()
    End Sub
    Private Sub TextB17_DoubleClick(sender As Object, e As EventArgs) Handles
TextB17.DoubleClick
        textoconta = 17
        idcuenta = 7
        frmBuscaCuenta.ShowDialog()
    End Sub
    Private Sub TextB18_DoubleClick(sender As Object, e As EventArgs) Handles
TextB18.DoubleClick
        textoconta = 18
        idcuenta = 8
        frmBuscaCuenta.ShowDialog()
    End Sub
    Private Sub CT_TransaccionesBindingNavigatorSaveItem_Click(sender As Object, e As
EventArgs)
        Me.Validate()
        Me.CT_TransaccionesBindingSource.EndEdit()
        Me.TableAdapterManager.UpdateAll(Me.DsTransacciones)
    End Sub
    Private Sub btnGuardar_Click(sender As Object, e As EventArgs) Handles
btnGuardar.Click
        Try
            If TextB1.Text = "" Then
                cuen1 = TextB11.Text

```

```

Else
    cuen1 = TextB1.Text
End If
If TextB2.Text = "" Then
    cuen2 = TextB12.Text
Else
    cuen2 = TextB2.Text
End If
If TextB3.Text = "" Then
    cuen3 = TextB13.Text
Else
    cuen3 = TextB3.Text
End If
If TextB4.Text = "" Then
    cuen4 = TextB14.Text
Else
    cuen4 = TextB4.Text
End If
If TextB5.Text = "" Then
    cuen5 = TextB15.Text
Else
    cuen5 = TextB5.Text
End If
If TextB6.Text = "" Then
    cuen6 = TextB16.Text
Else
    cuen6 = TextB6.Text
End If
If TextB7.Text = "" Then
    cuen7 = TextB17.Text
Else
    cuen7 = TextB7.Text
End If
If TextB8.Text = "" Then
    cuen8 = TextB18.Text
Else
    cuen8 = TextB8.Text
End If
Catch ex As Exception
    MessageBox.Show("Error en el registro de cuentas", "Error al Guardar",
    MessageBoxButtons.OK, MessageBoxIcon.Error)
End Try
If TextB21.Text = "" Then
    If TextTC1.Text = 1 Or 5 Then
        sal1 = -(TextB31.Text)
    Else
        sal1 = TextB31.Text
    End If
Else
    If TextTC1.Text = 1 Or 5 Then
        sal1 = TextB21.Text
    Else
        sal1 = -(TextB21.Text)
    End If
End If
If TextB22.Text = "" Then
    If TextTC2.Text = 1 Or 5 Then
        sal2 = -(TextB32.Text)
    Else
        sal2 = TextB32.Text
    End If

```

```

Else
  If TextTC2.Text = 1 Then
    sal2 = TextB22.Text
  Else
    sal2 = -(TextB22.Text)
  End If
End If
Try
  If TextB23.Text = "" Then
    If TextTC3.Text = 1 Or 5 Then
      sal3 = -(TextB33.Text)
    Else
      sal3 = TextB33.Text
    End If
  Else
    If TextTC3.Text = 1 Then
      sal3 = TextB23.Text
    Else
      sal3 = -(TextB23.Text)
    End If
  End If
Catch ex As Exception
End Try
Try
  If TextB24.Text = "" Then
    If TextTC4.Text = 1 Or 5 Then
      sal4 = -(TextB34.Text)
    Else
      sal4 = TextB34.Text
    End If
  Else
    If TextTC4.Text = 1 Then
      sal4 = TextB24.Text
    Else
      sal4 = -(TextB24.Text)
    End If
  End If
Catch ex As Exception
End Try
Try
  If TextB25.Text = "" Then
    If TextTC5.Text = 1 Or 5 Then
      sal5 = -(TextB35.Text)
    Else
      sal5 = TextB35.Text
    End If
  Else
    If TextTC5.Text = 1 Then
      sal5 = TextB25.Text
    Else
      sal5 = -(TextB25.Text)
    End If
  End If
Catch ex As Exception
End Try
Try
  If TextB26.Text = "" Then
    If TextTC6.Text = 1 Or 5 Then
      sal6 = -(TextB36.Text)
    Else
      sal6 = TextB36.Text
    End If
  End If

```

```

        End If
    Else
        If TextTC6.Text = 1 Or 5 Then
            sal6 = TextB26.Text
        Else
            sal6 = -(TextB26.Text)
        End If
    End If
Catch ex As Exception
End Try
Try
    If TextB27.Text = "" Then
        If TextTC7.Text = 1 Or 5 Then
            sal7 = -(TextB37.Text)
        Else
            sal7 = TextB37.Text
        End If
    Else
        If TextTC7.Text = 1 Or 5 Then
            sal7 = TextB27.Text
        Else
            sal7 = -(TextB27.Text)
        End If
    End If
Catch ex As Exception
End Try
Try
    If TextB28.Text = "" Then
        If TextTC8.Text = 1 Or 5 Then
            sal8 = -(TextB38.Text)
        Else
            sal8 = TextB38.Text
        End If
    Else
        If TextTC8.Text = 1 Or 5 Then
            sal8 = TextB28.Text
        Else
            sal8 = -(TextB28.Text)
        End If
    End If
Catch ex As Exception
End Try
Dim oadaptertran As New dsTransaccionesTableAdapters.CT_TransaccionesTableAdapter
Dim odataseptran As New dsTransacciones
Try
    If TextB29.Text = TextB39.Text Then
        desc = txtdescripcion.Text
        fechattran = DateTimePicker1.Text
        codtran = txtcodtran.Text
        oadaptertran.InsertQueryAgregarFila(codtran, cuen1, cuen2, cuen3, cuen4,
cuen5, cuen6, cuen7, cuen8, sal1, sal2, sal3, sal4, sal5, sal6, sal7, sal8, desc,
fechatran)
        MessageBox.Show("Registro guardado con exito", "Guardando datos",
MessageBoxButtons.OK, MessageBoxIcon.Information)
    Else
        MessageBox.Show("No se puede guardar porque se lleno mal la planilla" &
vbCrLf & "corrija los errores", "Error al guardar datos", MessageBoxButtons.OK,
MessageBoxIcon.Error)
    End If
Catch ex As Exception
End Try

```

```

End Sub
Private Sub TextB21_TextChanged(sender As Object, e As EventArgs) Handles
TextB21.TextChanged
Try
a = TextB21.Text
TextB29.Text = a + b + c + d + f + g + h + i
TextB41.Text = Format(a / t, "0.00")
TextB49.Text = Format(a / t + b / t + c / t + d / t + f / t + g / t + h / t +
i / t, "0.00")
Catch ex As Exception
End Try
End Sub
Private Sub TextB22_TextChanged(sender As Object, e As EventArgs) Handles
TextB22.TextChanged
Try
b = TextB22.Text
TextB29.Text = a + b + c + d + f + g + h + i
TextB42.Text = Format(b / t, "0.00")
TextB49.Text = Format(a / t + b / t + c / t + d / t + f / t + g / t + h / t +
i / t, "0.00")
Catch ex As Exception
End Try
End Sub
Private Sub TextB23_TextChanged(sender As Object, e As EventArgs) Handles
TextB23.TextChanged
Try
c = TextB23.Text
TextB29.Text = a + b + c + d + f + g + h + i
TextB43.Text = Format(c / t, "0.00")
TextB49.Text = Format(a / t + b / t + c / t + d / t + f / t + g / t + h / t +
i / t, "0.00")
Catch ex As Exception
End Try
End Sub
Private Sub TextB24_TextChanged(sender As Object, e As EventArgs) Handles
TextB24.TextChanged
Try
d = TextB24.Text
TextB29.Text = a + b + c + d + f + g + h + i
TextB44.Text = Format(d / t, "0.00")
TextB49.Text = Format(a / t + b / t + c / t + d / t + f / t + g / t + h / t +
i / t, "0.00")
Catch ex As Exception
End Try
End Sub
Private Sub TextB25_TextChanged(sender As Object, e As EventArgs) Handles
TextB25.TextChanged
Try
f = TextB25.Text
TextB29.Text = a + b + c + d + f + g + h + i
TextB45.Text = Format(f / t, "0.00")
TextB49.Text = Format(a / t + b / t + c / t + d / t + f / t + g / t + h / t +
i / t, "0.00")
Catch ex As Exception
End Try
End Sub
Private Sub TextB26_TextChanged(sender As Object, e As EventArgs) Handles
TextB26.TextChanged
Try
g = TextB26.Text
TextB29.Text = a + b + c + d + f + g + h + i

```

```

        TextB46.Text = Format(g / t, "0.00")
        TextB49.Text = Format(a / t + b / t + c / t + d / t + f / t + g / t + h / t +
i / t, "0.00")
    Catch ex As Exception
    End Try
End Sub
Private Sub TextB27_TextChanged(sender As Object, e As EventArgs) Handles
TextB27.TextChanged
    Try
        h = TextB27.Text
        TextB29.Text = a + b + c + d + f + g + h + i
        TextB47.Text = Format(h / t, "0.00")
        TextB49.Text = Format(a / t + b / t + c / t + d / t + f / t + g / t + h / t +
i / t, "0.00")
    Catch ex As Exception
    End Try
End Sub
Private Sub TextB28_TextChanged(sender As Object, e As EventArgs) Handles
TextB28.TextChanged
    Try
        i = TextB28.Text
        TextB29.Text = a + b + c + d + f + g + h + i
        TextB48.Text = Format(i / t, "0.00")
        TextB49.Text = Format(a / t + b / t + c / t + d / t + f / t + g / t + h / t +
i / t, "0.00")
    Catch ex As Exception
    End Try
End Sub
Private Sub TextB31_TextChanged(sender As Object, e As EventArgs) Handles
TextB31.TextChanged
    Try
        j = TextB31.Text
        TextB39.Text = j + k + l + m + n + o + p + q
        TextB51.Text = Format(j / t, "0.00")
        TextB59.Text = Format(j / t + k / t + l / t + m / t + n / t + o / t + p / t +
q / t, "0.00")
    Catch ex As Exception
    End Try
End Sub
Private Sub TextB32_TextChanged(sender As Object, e As EventArgs) Handles
TextB32.TextChanged
    Try
        k = TextB32.Text
        TextB39.Text = j + k + l + m + n + o + p + q
        TextB52.Text = Format(k / t, "0.00")
        TextB59.Text = Format(j / t + k / t + l / t + m / t + n / t + o / t + p / t +
q / t, "0.00")
    Catch ex As Exception
    End Try
End Sub
Private Sub TextB33_TextChanged(sender As Object, e As EventArgs) Handles
TextB33.TextChanged
    Try
        l = TextB33.Text
        TextB39.Text = j + k + l + m + n + o + p + q
        TextB53.Text = Format(l / t, "0.00")
        TextB59.Text = Format(j / t + k / t + l / t + m / t + n / t + o / t + p / t +
q / t, "0.00")
    Catch ex As Exception
    End Try
End Sub

```

```

Private Sub TextB34_TextChanged(sender As Object, e As EventArgs) Handles
TextB34.TextChanged
    Try
        m = TextB34.Text
        TextB39.Text = j + k + l + m + n + o + p + q
        TextB54.Text = Format(m / t, "0.00")
        TextB59.Text = Format(j / t + k / t + l / t + m / t + n / t + o / t + p / t +
q / t, "0.00")
    Catch ex As Exception
    End Try
End Sub
Private Sub TextB35_TextChanged(sender As Object, e As EventArgs) Handles
TextB35.TextChanged
    Try
        n = TextB35.Text
        TextB39.Text = j + k + l + m + n + o + p + q
        TextB55.Text = Format(n / t, "0.00")
        TextB59.Text = Format(j / t + k / t + l / t + m / t + n / t + o / t + p / t +
q / t, "0.00")
    Catch ex As Exception
    End Try
End Sub
Private Sub TextB36_TextChanged(sender As Object, e As EventArgs) Handles
TextB36.TextChanged
    Try
        o = TextB36.Text
        TextB39.Text = j + k + l + m + n + o + p + q
        TextB56.Text = Format(o / t, "0.00")
        TextB59.Text = Format(j / t + k / t + l / t + m / t + n / t + o / t + p / t +
q / t, "0.00")
    Catch ex As Exception
    End Try
End Sub
Private Sub TextB37_TextChanged(sender As Object, e As EventArgs) Handles
TextB37.TextChanged
    Try
        p = TextB37.Text
        TextB39.Text = j + k + l + m + n + o + p + q
        TextB57.Text = Format(p / t, "0.00")
        TextB59.Text = Format(j / t + k / t + l / t + m / t + n / t + o / t + p / t +
q / t, "0.00")
    Catch ex As Exception
    End Try
End Sub
Private Sub TextB38_TextChanged(sender As Object, e As EventArgs) Handles
TextB38.TextChanged
    Try
        q = TextB38.Text
        TextB39.Text = j + k + l + m + n + o + p + q
        TextB58.Text = Format(q / t, "0.00")
        TextB59.Text = Format(j / t + k / t + l / t + m / t + n / t + o / t + p / t +
q / t, "0.00")
    Catch ex As Exception
    End Try
End Sub
Private Sub btnCancel_Click(sender As Object, e As EventArgs) Handles
btnCancelar.Click

    Close()
End Sub
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click

```

```
txtdescripcion.Clear()  
TextTC1.Clear()  
TextTC2.Clear()  
TextTC3.Clear()  
TextTC4.Clear()  
TextTC5.Clear()  
TextTC6.Clear()  
TextTC7.Clear()  
TextTC8.Clear()  
TextB1.Clear()  
TextB2.Clear()  
TextB3.Clear()  
TextB4.Clear()  
TextB5.Clear()  
TextB6.Clear()  
TextB7.Clear()  
TextB8.Clear()  
TextB11.Clear()  
TextB12.Clear()  
TextB13.Clear()  
TextB14.Clear()  
TextB15.Clear()  
TextB16.Clear()  
TextB17.Clear()  
TextB18.Clear()  
TextB21.Clear()  
TextB22.Clear()  
TextB23.Clear()  
TextB24.Clear()  
TextB25.Clear()  
TextB26.Clear()  
TextB27.Clear()  
TextB28.Clear()  
TextB29.Clear()  
TextB31.Clear()  
TextB32.Clear()  
TextB33.Clear()  
TextB34.Clear()  
TextB35.Clear()  
TextB36.Clear()  
TextB37.Clear()  
TextB38.Clear()  
TextB39.Clear()  
TextB41.Clear()  
TextB42.Clear()  
TextB43.Clear()  
TextB44.Clear()  
TextB45.Clear()  
TextB46.Clear()  
TextB47.Clear()  
TextB48.Clear()  
TextB49.Clear()  
TextB51.Clear()  
TextB52.Clear()  
TextB53.Clear()  
TextB54.Clear()  
TextB55.Clear()  
TextB56.Clear()  
TextB57.Clear()  
TextB58.Clear()  
TextB59.Clear()
```



```
a = 0
b = 0
c = 0
d = 0
f = 0
g = 0
h = 0
i = 0
j = 0
k = 0
l = 0
m = 0
n = 0
o = 0
p = 0
q = 0
Dim oadaptertran As New dsTransaccionesTableAdapters.CT_TransaccionesTableAdapter
contador = oadaptertran.ScalarQueryContador()
txtcodtran.Text = contador
End Sub
End Class
```



ANEXO 0-14: CODIGO FUENTE, CUENTAS, MODULO CONTABILIDAD

```
Public Class frmCuentas
    Private Sub CT_CuentasBindingNavigatorSaveItem_Click(sender As Object, e As
EventArgs) Handles CT_CuentasBindingNavigatorSaveItem.Click
        Try
            Me.CT_CuentasBindingSource.EndEdit()
            Me.TableAdapterManager.UpdateAll(Me.DsCuentas)
            Actualizar()
            MessageBox.Show("Cambios guardados con exito", "Guardando datos",
MessageBoxButtons.OK, MessageBoxIcon.Information)
        Catch ex As Exception
            If MessageBox.Show("Ocurrio un error en la base de datos" & vbCrLf & "desea
continuar?", "Error al guardar", MessageBoxButtons.YesNo, MessageBoxIcon.Question) =
Windows.Forms.DialogResult.No Then
                Actualizar()
            End If
        End Try
    End Sub
    Private Sub frmCuentas_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Actualizar()
    End Sub
    Private Sub Botones(ByVal x As Boolean)
        BindingNavigatorAddNewItem.Visible = x
        BindingNavigatorModificarItem.Visible = x
        BindingNavigatorDeleteItem.Visible = x
        CT_CuentasBindingNavigatorSaveItem.Visible = Not x
        BindingNavigatorCancelarItem.Visible = Not x
    End Sub
    Private Sub Bloquear(ByVal x As Boolean)
        Cod_CuentaMaskedTextBox.ReadOnly = x
        CuentaMaskedTextBox.ReadOnly = x
        SaldoMaskedTextBox.ReadOnly = x
        RubroMaskedTextBox.ReadOnly = x
        GrupoMaskedTextBox.ReadOnly = x
        TituloMaskedTextBox.ReadOnly = x
    End Sub
    Private Sub Actualizar()
        Me.CT_CuentasTableAdapter.Fill(Me.DsCuentas.CT_Cuentas)
        Botones(True)
        Bloquear(True)
        Me.CT_CuentasDataGridView.Enabled = True
        Me.CT_CuentasDataGridView.Focus()
    End Sub
    Private Sub btnBuscar_Click(sender As Object, e As EventArgs) Handles btnBuscar.Click
        Dim dato As String = txtDato.Text
        Try
            If cboModo.Items(1) = cboModo.Text Then
                Me.CT_CuentasTableAdapter.FillByCuenta(Me.DsCuentas.CT_Cuentas, dato)
            End If
        Catch ex As Exception
        End Try
    End Sub
    Private Sub cboModo_SelectedValueChanged(sender As Object, e As EventArgs) Handles
cboModo.SelectedValueChanged
        Try
            If cboModo.Items(0) = cboModo.Text Then
                Me.CT_CuentasTableAdapter.Fill(Me.DsCuentas.CT_Cuentas)
                txtDato.Clear()
            Else
                txtDato.Clear()
            End If
        End Try
    End Sub
End Class
```

```

        txtDato.Focus()
    End If
    If cboModo.Items(2) = cboModo.Text Then
        Me.CT_CuentasTableAdapter.FillByActivo(Me.DsCuentas.CT_Cuentas)
        txtDato.Clear()
    Else
        txtDato.Clear()
        txtDato.Focus()
    End If
    If cboModo.Items(3) = cboModo.Text Then
        Me.CT_CuentasTableAdapter.FillByPasivo(Me.DsCuentas.CT_Cuentas)
        txtDato.Clear()
    Else
        txtDato.Clear()
        txtDato.Focus()
    End If
    If cboModo.Items(4) = cboModo.Text Then
        Me.CT_CuentasTableAdapter.FillByCapital(Me.DsCuentas.CT_Cuentas)
        txtDato.Clear()
    Else
        txtDato.Clear()
        txtDato.Focus()
    End If
    If cboModo.Items(5) = cboModo.Text Then
        Me.CT_CuentasTableAdapter.FillByIngresos(Me.DsCuentas.CT_Cuentas)
        txtDato.Clear()
    Else
        txtDato.Clear()
        txtDato.Focus()
    End If
    If cboModo.Items(6) = cboModo.Text Then
        Me.CT_CuentasTableAdapter.FillByGastos(Me.DsCuentas.CT_Cuentas)
        txtDato.Clear()
    Else
        txtDato.Clear()
        txtDato.Focus()
    End If
Catch ex As Exception
End Try
End Sub
Private Sub Cod_CuentaMaskedTextBox_KeyPress(sender As Object, e As
KeyPressEventArgs) Handles Cod_CuentaMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub CuentaMaskedTextBox_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles CuentaMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub SaldoMaskedTextBox_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles SaldoMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub

```

```

Private Sub RubroMaskedTextBox_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles RubroMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub GrupoMaskedTextBox_KeyPress(sender As Object, e As KeyPressEventArgs)
Handles GrupoMaskedTextBox.KeyPress
    If e.KeyChar = Convert.ToChar(Keys.Enter) Then
        e.Handled = True
        SendKeys.Send("{TAB}")
    End If
End Sub
Private Sub Cod_CuentaMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles Cod_CuentaMaskedTextBox.Validating
    If Cod_CuentaMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(Cod_CuentaMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(Cod_CuentaMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub CuentaMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles CuentaMaskedTextBox.Validating
    If CuentaMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(CuentaMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(CuentaMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub SaldoMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles SaldoMaskedTextBox.Validating
    If SaldoMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(SaldoMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(SaldoMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub RubroMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles RubroMaskedTextBox.Validating
    If RubroMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(RubroMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(RubroMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub GrupoMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles GrupoMaskedTextBox.Validating
    If GrupoMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(GrupoMaskedTextBox, Nothing)
    Else
        e.Cancel = True

```

```

        ErrorProvider1.SetError(GrupoMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub TituloMaskedTextBox_Validating(sender As Object, e As
System.ComponentModel.CancelEventArgs) Handles TituloMaskedTextBox.Validating
    If TituloMaskedTextBox.MaskCompleted Then
        e.Cancel = False
        ErrorProvider1.SetError(TituloMaskedTextBox, Nothing)
    Else
        e.Cancel = True
        ErrorProvider1.SetError(TituloMaskedTextBox, "Dato no valido")
    End If
End Sub
Private Sub BindingNavigatorAddNewItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorAddNewItem.Click
    Botones(False)
    Bloquear(False)
    Me.CT_CuentasDataGridView.Enabled = False
    ErrorProvider1.Dispose()
    Cod_CuentaMaskedTextBox.Focus()
End Sub
Private Sub BindingNavigatorModificarItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorModificarItem.Click
    Botones(False)
    Bloquear(False)
    Me.CT_CuentasDataGridView.Enabled = False
    ErrorProvider1.Dispose()
    Cod_CuentaMaskedTextBox.Focus()
End Sub
Private Sub BindingNavigatorDeleteItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorDeleteItem.Click
    Botones(False)
    Me.CT_CuentasDataGridView.Enabled = False
End Sub
Private Sub BindingNavigatorCancelarItem_Click(sender As Object, e As EventArgs)
Handles BindingNavigatorCancelarItem.Click
    Actualizar()
End Sub
End Class

```

ANEXO 0-15: CODIGO FUENTE, INDICADORES FINANCIEROS, MODULO CONTABILIDAD

```
Public Class frmIndicesFin
    Private Sub frmIndicesFin_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Dim oadapterCuent As New dsCuentasTableAdapters.CT_CuentasTableAdapter
        Dim odatasepCuent As New dsCuentas
        activo = oadapterCuent.ScalarQueryActivo
        pasivo = oadapterCuent.ScalarQueryPasivo
        capital = oadapterCuent.ScalarQueryCapital
        actcorr = oadapterCuent.ScalarQueryActivoCorriente
        pascorr = oadapterCuent.ScalarQueryPasivoCorriente
        cmv = oadapterCuent.ScalarQueryCMV
        cmvsal = oadapterCuent.ScalarQueryCMVySalario
        cxp = oadapterCuent.ScalarQueryCuantaPagar
        cxc = oadapterCuent.ScalarQueryCuantasCobrar
        dlp = oadapterCuent.ScalarQueryDeudaLargoPla
        gastos = oadapterCuent.ScalarQueryGastos
        ingresos = oadapterCuent.ScalarQueryIngresos
        interes = oadapterCuent.ScalarQueryInteres
        inventario = oadapterCuent.ScalarQueryInventario
        ventaneta = oadapterCuent.ScalarQueryVentaNeta
        txtCaptrab.Text = Format(actcorr + pascorr, "0.00")
        txtInCorr.Text = Format(1 / (actcorr / pascorr), "0.00")
        txtPruAcida.Text = Format(1 / (actcorr - inventario) / pascorr, "0.00")
        txtEndeu.Text = Format((pasivo / activo) * 100, "0.00")
        txtEndeuLP.Text = Format(dlp / capital, "0.00")
        txtInApala.Text = Format(pasivo / capital, "0.00")
        txtCapPagInt.Text = Format((ingresos - gastos) / interes, "0.00")
        txtRotcxc.Text = Format((cxc / ventaneta) * 360, "0.00")
        txtRotcxp.Text = Format((cxp / cmv) * 360, "0.00")
        txtRotInv.Text = Format((inventario / cmv) * 360, "0.00")
        txtRotAct.Text = Format((ventaneta / activo) * 100, "0.00")
        txtMB.Text = Format(((ventaneta - cmv) / ventaneta) * 100, "0.00")
        txtMN.Text = Format(((ventaneta - cmvsal) / ventaneta) * 100, "0.00")
        txtRentAct.Text = Format((ventaneta - cmvsal) / activo, "0.00")
        txtRentPat.Text = Format((ventaneta - cmvsal) / capital, "0.00")
    End Sub
    Private Sub btnAtras_Click(sender As Object, e As EventArgs)
        Me.Close()
    End Sub
    Private Sub Imprimir()
        PrintDialog1.ShowDialog()
        PrintForm1.PrinterSettings = PrintDialog1.PrinterSettings
        PrintForm1.Print()
    End Sub
    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        Imprimir()
    End Sub
End Class
```

ANEXO 0-16: CODIGO FUENTE, ESTADO DE RESULTADOS, MODULO CONTABILIDAD

```

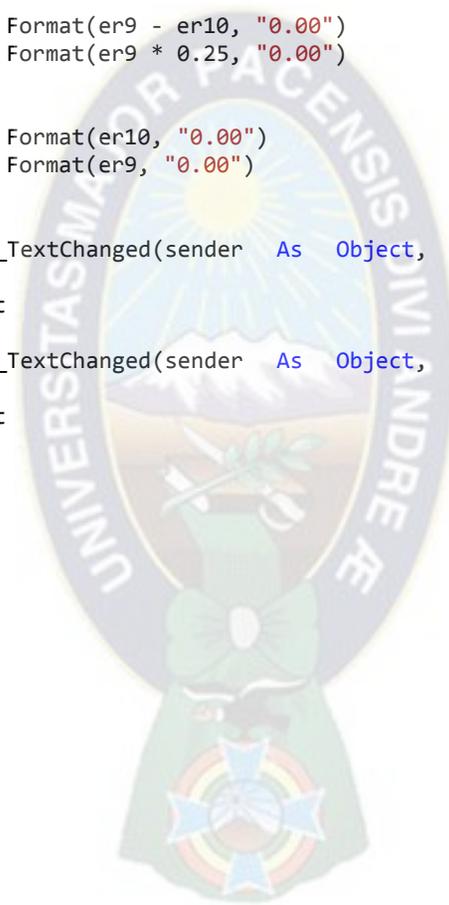
Public Class frmEstadoResultados
    Private Sub frmEstadoResultados_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Dim meses As String = ""
        If DateTimePicker1.Value.Month = 1 Then meses = "Enero"
        If DateTimePicker1.Value.Month = 2 Then meses = "Febrero"
        If DateTimePicker1.Value.Month = 3 Then meses = "Marzo"
        If DateTimePicker1.Value.Month = 4 Then meses = "Abril"
        If DateTimePicker1.Value.Month = 5 Then meses = "Mayo"
        If DateTimePicker1.Value.Month = 6 Then meses = "Junio"
        If DateTimePicker1.Value.Month = 7 Then meses = "Julio"
        If DateTimePicker1.Value.Month = 8 Then meses = "Agosto"
        If DateTimePicker1.Value.Month = 9 Then meses = "Septiembre"
        If DateTimePicker1.Value.Month = 10 Then meses = "Octubre"
        If DateTimePicker1.Value.Month = 11 Then meses = "Noviembre"
        If DateTimePicker1.Value.Month = 12 Then meses = "Diciembre"
        txtfecha.Text = "Del 1 de Enero de " & DateTimePicker1.Value.Year & " al " &
        DateTimePicker1.Value.Day & " de " & meses & " de " & DateTimePicker1.Value.Year
        Dim dsCuentas As New dsCuentasTableAdapters.CT_CuentasTableAdapter
        saldo1.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB1.Text), "0.00")
        saldo4.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB4.Text), "0.00")
        Saldo7.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB7.Text), "0.00")
        Saldo9.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB9.Text), "0.00")
        Saldo6.Text = Format(odaptercuentas.ScalarQueryGastosAdm, "0.00")
        Saldo10.Text = Format(odaptercuentas.ScalarQueryGastesExtra, "0.00")
    End Sub
    Private Sub Imprimir()
        PrintDialog1.ShowDialog()
        PrintForm1.PrinterSettings = PrintDialog1.PrinterSettings
        PrintForm1.Print()
    End Sub
    Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
        Imprimir()
    End Sub
    Dim er1, er2, er3, er4, er5, er6, er7, er8, er9, er10, er11 As Decimal
    Private Sub saldo1_TextChanged(sender As Object, e As EventArgs) Handles saldo1.TextChanged
        er1 = saldo1.Text
    End Sub
    Private Sub saldo4_TextChanged(sender As Object, e As EventArgs) Handles saldo4.TextChanged
        er2 = saldo4.Text
        saldo5.Text = Format(er1 - er2, "0.00")
    End Sub
    Private Sub saldo5_TextChanged(sender As Object, e As EventArgs) Handles saldo5.TextChanged
        er3 = saldo5.Text
    End Sub
    Private Sub Saldo6_TextChanged(sender As Object, e As EventArgs) Handles Saldo6.TextChanged
        er4 = Saldo6.Text
        Saldo8.Text = Format(er3 - er4 - er5, "0.00")
    End Sub
    Private Sub Saldo7_TextChanged(sender As Object, e As EventArgs) Handles Saldo7.TextChanged
        er5 = Saldo7.Text
    End Sub
    Private Sub Saldo8_TextChanged(sender As Object, e As EventArgs) Handles Saldo8.TextChanged

```

```

        er6 = Saldo8.Text
    End Sub
    Private Sub Saldo9_TextChanged(sender As Object, e As EventArgs) Handles
Saldo9.TextChanged
        er7 = Saldo9.Text
    End Sub
    Private Sub Saldo10_TextChanged(sender As Object, e As EventArgs) Handles
Saldo10.TextChanged
        er8 = Saldo10.Text
        Saldo11.Text = Format(er6 + er7 - er8, "0.00")
    End Sub
    Private Sub Saldo11_TextChanged(sender As Object, e As EventArgs) Handles
Saldo11.TextChanged
        er9 = Saldo11.Text
        If er9 > 0 Then
            Saldo13.Text = Format(er9 - er10, "0.00")
            Saldo12.Text = Format(er9 * 0.25, "0.00")
        Else
            er10 = 0
            Saldo12.Text = Format(er10, "0.00")
            Saldo13.Text = Format(er9, "0.00")
        End If
    End Sub
    Private Sub Saldo12_TextChanged(sender As Object, e As EventArgs) Handles
Saldo12.TextChanged
        er10 = Saldo12.Text
    End Sub
    Private Sub Saldo13_TextChanged(sender As Object, e As EventArgs) Handles
Saldo13.TextChanged
        er11 = Saldo13.Text
    End Sub
End Class

```



ANEXO 0-17: CODIGO FUENTE, BALANCE GENERAL, MODULO CONTABILIDAD

```
Public Class frmBalanceGeneral
    Dim bgcuenta As String
    Dim bgsaldo As Decimal
    Private Sub frmBalanceGeneral_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        Dim odaptercuentas As New dsCuentasTableAdapters.CT_CuentasTableAdapter
        saldo1.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB1.Text), "0.00")
        saldo2.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB2.Text), "0.00")
        saldo3.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB3.Text), "0.00")
        saldo4.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB4.Text), "0.00")
        saldo5.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB5.Text), "0.00")
        saldo6.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB6.Text), "0.00")
        saldo7.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB7.Text), "0.00")
        saldo8.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB8.Text), "0.00")
        saldo9.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB9.Text), "0.00")
        saldo10.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB10.Text), "0.00")
        saldo11.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB11.Text), "0.00")
        saldo12.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB12.Text), "0.00")
        saldo13.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB13.Text), "0.00")
        saldo14.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB14.Text), "0.00")
        saldo15.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB15.Text), "0.00")
        saldo16.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB16.Text), "0.00")
        saldo17.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB17.Text), "0.00")
        saldo18.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB18.Text), "0.00")
        saldo19.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB19.Text), "0.00")
        saldo20.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB20.Text), "0.00")
        saldo21.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB21.Text), "0.00")
        saldo22.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB22.Text), "0.00")
        saldo31.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB31.Text), "0.00")
        saldo32.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB32.Text), "0.00")
        saldo33.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB33.Text), "0.00")
        saldo34.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB34.Text), "0.00")
        saldo35.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB35.Text), "0.00")
        saldo36.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB36.Text), "0.00")
        saldo37.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB37.Text), "0.00")
        saldo38.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB38.Text), "0.00")
        saldo39.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB39.Text), "0.00")
        saldo40.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB40.Text), "0.00")
        saldo41.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB41.Text), "0.00")
        saldo42.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB42.Text), "0.00")
        saldo43.Text = Format(odaptercuentas.ScalarQueryoCuenta(TB43.Text), "0.00")
        Dim meses As String = ""
        If DateTimePicker1.Value.Month = 1 Then meses = "Enero"
        If DateTimePicker1.Value.Month = 2 Then meses = "Febrero"
        If DateTimePicker1.Value.Month = 3 Then meses = "Marzo"
        If DateTimePicker1.Value.Month = 4 Then meses = "Abril"
        If DateTimePicker1.Value.Month = 5 Then meses = "Mayo"
        If DateTimePicker1.Value.Month = 6 Then meses = "Junio"
        If DateTimePicker1.Value.Month = 7 Then meses = "Julio"
        If DateTimePicker1.Value.Month = 8 Then meses = "Agosto"
        If DateTimePicker1.Value.Month = 9 Then meses = "Septiembre"
        If DateTimePicker1.Value.Month = 10 Then meses = "Octubre"
        If DateTimePicker1.Value.Month = 11 Then meses = "Noviembre"
        If DateTimePicker1.Value.Month = 12 Then meses = "Diciembre"
        txtfecha.Text = "al " & DateTimePicker1.Value.Day & " de " & meses & " de " &
        DateTimePicker1.Value.Year
    End Sub
    Dim ac1, ac2, ac3, ac4, ac5, ac6, in1, in2, af1, af2, af3, af4, af5, af6, af7, af8,
    af9, af10, af11, af12, af13, ad As Decimal
```

```

    Dim to1, to2, to3, to4, to5, to6, to7, to8, to9 As Decimal
    Dim pc1, pc2, pc3, pc4, pc5, pc6, pnc1, pnc2, pnc3, pnc4, ca1, ca2, ca3 As Decimal
    Private Sub saldo1_TextChanged(sender As Object, e As EventArgs) Handles
saldo1.TextChanged
        ac1 = saldo1.Text
    End Sub
    Private Sub saldo2_TextChanged(sender As Object, e As EventArgs) Handles
saldo2.TextChanged
        ac2 = saldo2.Text
    End Sub
    Private Sub saldo3_TextChanged(sender As Object, e As EventArgs) Handles
saldo3.TextChanged
        ac3 = saldo3.Text
    End Sub
    Private Sub saldo4_TextChanged(sender As Object, e As EventArgs) Handles
saldo4.TextChanged
        ac4 = saldo4.Text
    End Sub
    Private Sub saldo5_TextChanged(sender As Object, e As EventArgs) Handles
saldo5.TextChanged
        ac5 = saldo5.Text
    End Sub
    Private Sub saldo6_TextChanged(sender As Object, e As EventArgs) Handles
saldo6.TextChanged
        ac6 = saldo6.Text
        tot1.Text = ac1 + ac2 + ac3 + ac4 + ac5 + ac6
    End Sub
    Private Sub saldo7_TextChanged(sender As Object, e As EventArgs) Handles
saldo7.TextChanged
        in1 = saldo7.Text
    End Sub
    Private Sub saldo8_TextChanged(sender As Object, e As EventArgs) Handles
saldo8.TextChanged
        in2 = saldo8.Text
        tot2.Text = in1 + in2
    End Sub
    Private Sub saldo9_TextChanged(sender As Object, e As EventArgs) Handles
saldo9.TextChanged
        af1 = saldo9.Text
    End Sub
    Private Sub saldo10_TextChanged(sender As Object, e As EventArgs) Handles
saldo10.TextChanged
        af2 = saldo10.Text
    End Sub
    Private Sub saldo11_TextChanged(sender As Object, e As EventArgs) Handles
saldo11.TextChanged
        af3 = saldo11.Text
    End Sub
    Private Sub saldo12_TextChanged(sender As Object, e As EventArgs) Handles
saldo12.TextChanged
        af4 = saldo12.Text
    End Sub
    Private Sub saldo13_TextChanged(sender As Object, e As EventArgs) Handles
saldo13.TextChanged
        af5 = saldo13.Text
    End Sub
    Private Sub saldo14_TextChanged(sender As Object, e As EventArgs) Handles
saldo14.TextChanged
        af6 = saldo14.Text
    End Sub

```

```

    Private Sub saldo15_TextChanged(sender As Object, e As EventArgs) Handles
saldo15.TextChanged
        af7 = saldo15.Text
    End Sub
    Private Sub saldo16_TextChanged(sender As Object, e As EventArgs) Handles
saldo16.TextChanged
        af8 = saldo16.Text
    End Sub
    Private Sub saldo17_TextChanged(sender As Object, e As EventArgs) Handles
saldo17.TextChanged
        af9 = saldo17.Text
    End Sub
    Private Sub saldo18_TextChanged(sender As Object, e As EventArgs) Handles
saldo18.TextChanged
        af10 = saldo18.Text
    End Sub
    Private Sub saldo19_TextChanged(sender As Object, e As EventArgs) Handles
saldo19.TextChanged
        af11 = saldo19.Text
    End Sub
    Private Sub saldo20_TextChanged(sender As Object, e As EventArgs) Handles
saldo20.TextChanged
        af12 = saldo20.Text
    End Sub
    Private Sub saldo21_TextChanged(sender As Object, e As EventArgs) Handles
saldo21.TextChanged
        af13 = saldo21.Text
        tot3.Text = af1 + af2 + af3 + af4 + af5 + af6 + af7 + af8 + af9 + af10 + af11 +
af12 + af13
    End Sub
    Private Sub saldo22_TextChanged(sender As Object, e As EventArgs) Handles
saldo22.TextChanged
        ad = saldo22.Text
        tot4.Text = ad
    End Sub
    Private Sub tot4_TextChanged(sender As Object, e As EventArgs) Handles
tot4.TextChanged
        to4 = tot4.Text
        tot5.Text = to1 + to2 + to3 + to4 + to5
    End Sub
    Private Sub tot3_TextChanged(sender As Object, e As EventArgs) Handles
tot3.TextChanged
        to3 = tot3.Text
    End Sub
    Private Sub tot2_TextChanged(sender As Object, e As EventArgs) Handles
tot2.TextChanged
        to2 = tot2.Text
    End Sub
    Private Sub tot1_TextChanged(sender As Object, e As EventArgs) Handles
tot1.TextChanged
        to1 = tot1.Text
    End Sub
    Private Sub saldo31_TextChanged(sender As Object, e As EventArgs) Handles
saldo31.TextChanged
        pc1 = saldo31.Text
    End Sub
    Private Sub saldo32_TextChanged(sender As Object, e As EventArgs) Handles
saldo32.TextChanged
        pc2 = saldo32.Text
    End Sub

```

```

Private Sub saldo33_TextChanged(sender As Object, e As EventArgs) Handles
saldo33.TextChanged
    pc3 = saldo33.Text
End Sub
Private Sub saldo34_TextChanged(sender As Object, e As EventArgs) Handles
saldo34.TextChanged
    pc4 = saldo34.Text
End Sub
Private Sub saldo35_TextChanged(sender As Object, e As EventArgs) Handles
saldo35.TextChanged
    pc5 = saldo35.Text
End Sub
Private Sub saldo36_TextChanged(sender As Object, e As EventArgs) Handles
saldo36.TextChanged
    pc6 = saldo36.Text
    tot6.Text = pc1 + pc2 + pc3 + pc4 + pc5 + pc6
End Sub
Private Sub saldo37_TextChanged(sender As Object, e As EventArgs) Handles
saldo37.TextChanged
    pnc1 = saldo37.Text
End Sub
Private Sub saldo38_TextChanged(sender As Object, e As EventArgs) Handles
saldo38.TextChanged
    pnc2 = saldo38.Text
End Sub
Private Sub saldo39_TextChanged(sender As Object, e As EventArgs) Handles
saldo39.TextChanged
    pnc3 = saldo39.Text
End Sub
Private Sub saldo40_TextChanged(sender As Object, e As EventArgs) Handles
saldo40.TextChanged
    pnc4 = saldo40.Text
    tot7.Text = pnc1 + pnc2 + pnc3 + pnc4
End Sub
Private Sub saldo41_TextChanged(sender As Object, e As EventArgs) Handles
saldo41.TextChanged
    ca1 = saldo41.Text
End Sub
Private Sub saldo42_TextChanged(sender As Object, e As EventArgs) Handles
saldo42.TextChanged
    ca2 = saldo42.Text
End Sub
Private Sub saldo43_TextChanged(sender As Object, e As EventArgs) Handles
saldo43.TextChanged
    ca3 = saldo43.Text
    tot9.Text = ca1 + ca2 + ca3
End Sub
Private Sub tot6_TextChanged(sender As Object, e As EventArgs) Handles
tot6.TextChanged
    to6 = tot6.Text
End Sub
Private Sub tot7_TextChanged(sender As Object, e As EventArgs) Handles
tot7.TextChanged
    to7 = tot7.Text
    tot8.Text = to6 + to7
End Sub
Private Sub tot8_TextChanged(sender As Object, e As EventArgs) Handles
tot8.TextChanged
    to8 = tot8.Text
End Sub

```

```
Private Sub tot9_TextChanged(sender As Object, e As EventArgs) Handles
tot9.TextChanged
    to9 = tot9.Text
    tot10.Text = to8 + to9
End Sub
Private Sub Imprimir()
    PrintDialog1.ShowDialog()
    PrintForm1.PrinterSettings = PrintDialog1.PrinterSettings
    PrintForm1.Print()
End Sub
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Imprimir()
End Sub
End Class
```



The background features a large, faded watermark of the University of the Pacific logo. The logo is an oval shape with a sunburst at the top, a book in the center, and a green flower at the bottom. The text "UNIVERSITAS MAJOR PACENSIS DIVI AUREAE" is written around the perimeter of the oval.

ANEXO E:

NORMAS

CAPÍTULO II ASPECTOS TÉCNICOS DE LAS FACTURAS O NOTAS FISCALES

Artículo 10.- (Formato General) I. Los sujetos pasivos o terceros responsables del IVA, en función a la(s) modalidad(es) de facturación por la(s) que optare(n), deberán aplicar en las facturas o notas fiscales los siguientes formatos e información mínima, según corresponda:

1. Datos Básicos del Sujeto Pasivo.- (parte superior izquierda):
 - a) Razón Social en el caso de Personas Jurídicas o Nombre en el caso de Personas Naturales.
 - b) Domicilio tributario (casa matriz), número(s) telefónico(s) y la Alcaldía a la que pertenece (o domicilio y Alcaldía declarada para el caso de espectáculos públicos eventuales).
 - c) Número de sucursal, domicilio, número(s) telefónico(s) y la Alcaldía a la que pertenece (excepto en espectáculos públicos eventuales); sólo en el caso que la nota fiscal sea emitida en una sucursal.
 - d) Tipo de Autoimpresor y Número del Autoimpresor asignado por el SIN, para las facturas o notas fiscales de las modalidades de facturación, Punto de Venta Da Vinci (PVD), Computarizada (SFC) y Máquina Registradora (MR).
2. Datos de Dosificación.- (parte superior derecha):
 - a) Número de Identificación Tributaria (NIT) del sujeto pasivo (o de la Gerencia Distrital correspondiente, para el caso de espectáculos públicos eventuales).
 - b) Número correlativo de factura, nota fiscal o documento equivalente.
 - c) Número de Autorización.
 - d) El término "ORIGINAL" o "COPIA", según corresponda.
3. Título y Subtítulo.- (parte superior central):
 - a) "FACTURA" o "FACTURA POR TERCEROS" o "FACTURA CONJUNTA" o "FACTURA COMERCIAL DE EXPORTACIÓN" o "RECIBO DE ALQUILER" o "NOTA DE CRÉDITO-DÉBITO" o "FACTURA DE VENTA EN ZONAS FRANCAS" o "FACTURA TURISTICA"; según corresponda.
 - b) Como subtítulo: "Sin Derecho a Crédito Fiscal" (según el Artículo 44 de la presente Resolución); o "Cantidad de Facturas: XX" (ejemplo: Cantidad de Facturas: 05) sólo para la Característica Especial Facturación Conjunta.

4. Datos de la Transacción Comercial.- (parte central):
- a) Lugar y Fecha de emisión, en el siguiente orden: LUGAR, DIA, MES, AÑO.
 - b) Nombre o Razón Social del comprador.
 - c) Número de Identificación Tributaria o número de Documento de Identificación del comprador.
 - d) Domicilio del comprador, sólo en el caso de servicios públicos (Electricidad, Agua, Gas domiciliario y Teléfono).
 - e) Detalle: Cantidad, Concepto, Precio Unitario y Total (en caso de corresponder Descuentos e ICE).
 - f) Tipo de cambio oficial de venta correspondiente a la fecha de la transacción, cuando la operación sea en moneda extranjera. Para el caso de entidades del sistema financiero nacional, estas podrán utilizar el tipo de cambio de moneda extranjera, de acuerdo a disposiciones emitidas por la Superintendencia de Bancos y Entidades Financieras.
 - g) Total General en Bolivianos (numeral y literal).
5. Datos Finales.- (parte inferior):
- a) Código de Control, cuando corresponda (izquierda).
 - b) Fecha límite de emisión (a la derecha y en negrillas).
 - c) En negrillas y legible la leyenda: "La reproducción total o parcial y/o el uso no autorizado de esta Nota Fiscal, constituye un delito a ser sancionado conforme a Ley" (Central).

II. Las facturas o notas fiscales con la Característica Especial Facturación por Terceros, deberán consignar la siguiente información (ver Anexo 2):

- 1. Tercero Emisor.- En la parte superior de la nota fiscal, la información del Tercero emisor dispuesta por el inciso a) del numeral 1 del párrafo I del presente Artículo.
- 2. Sujeto Pasivo Titular.- Toda la información dispuesta en los numerales 1al5 del párrafo I del presente Artículo.

III. Las facturas o notas fiscales con la Característica Especial Facturación Conjunta, deberán consignar la siguiente información (ver anexo 3):

1. Sujeto pasivo emisor: Primero debe consignarse la información detallada en el párrafo I del presente Artículo, respecto al Sujeto Pasivo Emisor, excepto el inciso c) del numeral 5 del mismo párrafo.
2. Sujeto(s) pasivo(s) incluido(s): Una vez consignada la información previamente citada, para cada sujeto pasivo incluido se debe consignar los Datos Básicos de los incisos a), b) y c) del numeral 1 del párrafo I del presente Artículo, así como los Datos de Dosificación excepto el inciso d), del numeral 2 del mismo párrafo. Luego la información de los Datos de la Transacción Comercial establecida en los incisos e), f), y g) del numeral 4 y los Datos Finales de los incisos a) y b) del numeral 5 del mismo párrafo.
3. Por única vez: Al final de la factura o nota fiscal, la suma total (literal y numeral), así como la leyenda establecida en el inciso c) del numeral 5 del párrafo I del presente Artículo.

IV. Las facturas o notas fiscales a ser emitidas bajo la modalidad de Facturación Prevalorada, deberán consignar la misma información establecida en el párrafo I del presente Artículo, con las siguientes particularidades (ver Anexo 4):

1. Datos de la Transacción Comercial: Las facturas o notas fiscales de esta modalidad de facturación no requieren ser emitidas con los datos del comprador dispuestos en los incisos b), c), d) y f) del numeral 4 del párrafo I del presente Artículo, debiendo cumplir con la demás información referida en el numeral señalado. Asimismo, en relación a la fecha de emisión se aplicará el siguiente tratamiento diferenciado:

- a. Fecha de emisión preimpresa: Para el caso de espectáculos públicos eventuales o permanentes, u otros similares con fecha predeterminada.
- b. Fecha de emisión consignada de forma manuscrita, con fechero u otro medio tecnológico: Para el caso de servicios de espectáculos públicos permanentes de realización continua, venta de GLP en garrafas, pago por uso de terminales terrestres o aéreas, u otros similares.
- c. Sin fecha de emisión: Para aquellas notas fiscales que por su naturaleza y forma de emisión no puedan consignar la fecha, específicamente para servicios de telefonía prepagados con tarjetas y venta de productos a través de máquinas dispensadoras.

V. Las facturas o notas fiscales con la Característica Especial Facturación Comercial de Exportación, deberán consignar la misma información establecida en el párrafo I del presente Artículo (excepto la del inciso c) del numeral 4 del mismo párrafo), pudiendo adecuar el formato, información a consignar, moneda e idioma de la información de los incisos e), f) y g) del numeral 4 del mismo párrafo, en función a las particularidades y necesidades de la actividad (ver Anexo 5).

VI. Las facturas o notas fiscales con la Característica Especial Notas de Crédito - Débito, deberán consignar la misma información establecida en el párrafo I del presente Artículo, con las siguientes particularidades (ver Anexo 6):

1. Datos de Dosificación.- en la parte superior derecha, la misma información dispuesta en el numeral 2 del párrafo I del presente Artículo, consignando en el inciso d) las leyendas "ORIGINAL CLIENTE - NOTA DE DÉBITO" o "COPIA EMISOR - NOTA DE CRÉDITO", según corresponda.

2. Datos de la Transacción Comercial.- La misma información dispuesta en los incisos a), b), c) y d) del numeral 4 del párrafo I del presente Artículo, considerando en el inciso e) lo siguiente:

2.1 Datos de la Factura o Notafiscal Original.-

- a) Número correlativo de factura o nota fiscal.
- b) Número de Autorización.
- c) Fecha de emisión.
- d) Detalle: Cantidad, Concepto, Precio Unitario, Descuentos y Total.
- e) Tipo de cambio oficial de venta correspondiente a la fecha de la transacción, cuando la operación sea en moneda extranjera.
- f) Total General en Bolivianos.

2.2 Datos de la Devolución o Rescisión.-

- a) Detalle: Cantidad, Concepto, Precio Unitario e Importe Pagado, en la factura original.
- b) Importe total devuelto (numeral y literal).
- c) Monto efectivo del Crédito - Débito.

VII. Las facturas o notas fiscales emitidas a partir de la modalidad Punto de Venta Da Vinci, aplicarán los formatos establecidos en el párrafo I del presente Artículo,

considerando que el Logo y el Nombre o Razón Social del Sujeto Pasivo, deberá ser preimpreso o impreso en la parte superior izquierda del documento, de acuerdo a los tamaños previstos en el párrafo II del Artículo 13 de la presente Resolución.

Artículo 11.- (Formatos para Tickets y Rollos) I. Los sujetos pasivos o terceros responsables que opten por la modalidad de Facturación Máquinas Registradoras y aquellos que emitan Facturas o Notas Fiscales en Rollo a partir de las modalidades de Facturación Computarizada y Punto de Venta Da Vinci, deberán aplicar los siguientes formatos (ver Anexo 7 y 8):

1. Máquinas Registradoras:

1.1 Anverso:

- a) Título: "FACTURA".
- b) Número de Identificación Tributaria (NIT) del sujeto pasivo. c) Número de la Casa Matriz o Sucursal (Casa Matriz = 0; Sucursal = número asignado por el SIN) y Número del Autoimpresor (MR) asignado por el SIN.
- d) Número de Autorización.
- e) Número correlativo de factura o nota fiscal.
- f) Fecha de emisión, en el siguiente formato: DIA, MES, AÑO (DD/MM/AAAA).
- g) Número de Identificación Tributaria o número de Documento de Identificación del comprador.
- h) Detalle: Cantidad, Concepto, Precio Unitario y Total (en caso de corresponder descuentos).
- i) Total General en Bolivianos (numeral). j) Fecha límite de emisión.

1.2 Reverso:

- a) Razón Social en el caso de Personas Jurídicas o Nombre en el caso de Personas Naturales.
- b) Número de Identificación Tributaria (NIT) del sujeto pasivo.
- c) Domicilio tributario (casa matriz), número(s) telefónico(s) y la Alcaldía a la que pertenece.
- d) Número de sucursal, domicilio, número(s) telefónico(s) y la Alcaldía a la que pertenece; sólo en el caso que la nota fiscal sea emitida en una sucursal, sin perjuicio de incluir la misma información para el resto de sus sucursales.

- e) En negrillas y legible la leyenda: "La reproducción total o parcial y/o el uso no autorizado de esta Nota Fiscal, constituye un delito a ser sancionado conforme a Ley".

2. Facturación Computarizada y Da Vinci, en rollos:

2.1 Anverso:

- a) Razón Social en el caso de Personas Jurídicas o Nombre en el caso de Personas Naturales.
- b) Número de la Casa Matriz o Sucursal y Número del Autoimpresor (SFC o PVD) asignado por el SIN.
- c) Domicilio, número(s) telefónico(s) y la Alcaldía a la que pertenece la Casa Matriz o Sucursal de emisión (Casa Matriz = 0; Sucursal = número asignado por el SIN).
- d) Título y Subtítulo, según corresponda.
- e) Número de Identificación Tributaria (NIT) del sujeto pasivo.
- f) Número correlativo de factura o nota fiscal.
- g) Número de Autorización.
- h) Fecha de emisión, en el siguiente formato: DIA, MES, AÑO (DD/MM/AAAA).
- i) Nombre o Razón Social del comprador.
- j) Número de Identificación Tributaria o número de

Documento de Identificación del comprador. k) Detalle: Cantidad, Concepto, Precio Unitario y Total (en caso de corresponder descuentos e ICE). l) Tipo de cambio oficial de venta correspondiente a la fecha de la transacción, cuando la operación sea en moneda extranjera. Para el caso de entidades del sistema financiero nacional, estas podrán utilizar el tipo de cambio de moneda extranjera, de acuerdo a disposiciones emitidas por la Superintendencia de Bancos y Entidades Financieras. m) Total General en Bolivianos (numeral y literal). n) Código de Control. o) En negrillas la fecha límite de emisión. p) En negrillas y legible la leyenda: "La reproducción total o parcial y/o el uso no autorizado de esta Nota Fiscal, constituye un delito a ser sancionado conforme a Ley".

2.2. Reverso:

- a) Razón Social en el caso de Personas Jurídicas o Nombre en el caso de Personas Naturales.

- b) Número de Identificación Tributaria (NIT) del sujeto pasivo.
- c) Domicilio tributario (Casa Matriz), número(s) telefónico(s) y la Alcaldía a la que pertenece.
- d) Número de sucursal, domicilio, número(s) telefónico(s) y la Alcaldía a la que pertenece; sólo en el caso que la nota fiscal sea emitida en una sucursal, sin perjuicio de incluir la misma información para el resto de sus sucursales.

Artículo 12.- (Consistencia de la Información) La información del sujeto pasivo prevista en los Artículos precedentes, necesariamente debe coincidir con la registrada en el Padrón Nacional de Contribuyentes. Asimismo, la información de dosificación consignada en dichos documentos, debe coincidir con la otorgada por los Sistemas del SIN.

Artículo 13.- (Tamaños) I. Las facturas o notas fiscales del Sistema de Facturación, tanto originales como copias, deberán aplicar los límites mínimos y máximos según el siguiente cuadro:

Concepto	Minimo	Maximo
General	1/4 Oficio	Oficio
Factura o Notas Fiscales Prevaloradas	1/16 Oficio	1/4 Oficio
Tickets (Ancho)	35 mm	50 mm
Factura o Notas Fiscales en Rollo (Ancho)	70 mm	80 mm
Sticker	2 x 7 cm	4 x 7 cm

II. Sin perjuicio de los límites previamente establecidos y sólo a efectos de la Modalidad de Facturación Da Vinci, se aplicarán los siguientes tamaños fijos para las facturas y logos:

Tamaño Fijo	Margen Izquierdo	Margen Superior	Logo
Carta	0.5 cm	1.5 cm	2x2 cm
1/2 Carta	0.5 cm	0.5 cm	1.5x1.5 cm
1/4 Oficio	0.5 cm	0.5 cm	0.9x0.9 cm
Rollo de 75 mm ancho			Al reverso

III. Las facturas o notas fiscales emitidas a través de la modalidad de Facturación en Línea, serán impresas únicamente en tamaño carta.

IV. Para el caso de Facturas Prevaloradas emitidas en formato cuerpo-talón, los límites se aplicarán tanto para el cuerpo como para el talón (ver Anexo 9).

V. Aquellos sujetos pasivos o terceros responsables, que por razones tecnológicas no puedan aplicar los límites previstos en el párrafo I del presente Artículo para Tickets o Facturas o Notas Fiscales en Rollo, podrán solicitar autorización expresa ante el SIN, conforme al procedimiento dispuesto en el Artículo 35 de la presente disposición.

Artículo 14.- (Colores y Materiales) I. Las facturas o notas fiscales deberán preimprimirse o imprimirse y emitirse, mínimamente observando lo siguiente:

1. En las modalidades de Facturación Computarizada y Punto de Venta Da Vinci, las facturas o notas fiscales deberán contener información personalizada del sujeto pasivo, es decir el Logo (si corresponde, pudiendo mantener sus colores originales) y Nombre o Razón Social del sujeto pasivo, en cualquier color distinto de negro. El resto de la información podrá ser impresa en cualquier color.
2. En las modalidades de Facturación en Línea y Electrónica, las facturas o notas fiscales pueden ser emitidas sobre papel en blanco (salvo requerimiento del sujeto pasivo), pudiendo imprimirse la información correspondiente en cualquier color.
3. En la modalidad de Facturación Manual, las facturas o notas fiscales deberán emitirse sobre papel que contenga preimpresa toda la información fija del documento (Datos Básicos, Datos de Dosificación, Título, Subtítulo y Datos Finales) en cualquier color distinto de negro (excepción no aplicable al Logo) debiendo consignar de forma manuscrita y en cualquier color la información relativa a la transacción.
4. En la modalidad de Facturación Prevalorada, las facturas o notas fiscales deberán emitirse en papel que contenga de forma preimpresa la información dispuesta al efecto, en cualquier color distinto de negro (excepción no aplicable al Logo), incluida la fecha de emisión de acuerdo al tratamiento dispuesto en el párrafo IV. del Artículo 10 de la presente Resolución.
5. En la modalidad de Facturación Máquinas Registradoras y Facturación Computarizada en Rollos, las facturas o notas fiscales serán emitidas en papel cuyo reverso contenga preimpresa la información dispuesta en el Artículo 11 de la presente disposición, en cualquier color distinto de negro (excepción no aplicable al

Logo), debiendo imprimirse la información correspondiente al anverso en cualquier color.

II. Las facturas o notas fiscales deberán emitirse en papel (incluso químico) con el gramaje mínimo siguiente: originales 48.8 gr. y para las copias 30 gr., respectivamente.

III. Excepto para las modalidades de Facturación en Línea y Electrónica, las facturas o notas fiscales no deberán ser emitidas utilizando impresión en papel térmico.

IV. Como regla general, los materiales y colores utilizados para la elaboración de los originales y copias físicas de las facturas o notas fiscales, deberán permitir preimprimir, imprimir o consignar de forma nítida, legible, precisa y permanente, la información establecida al efecto.



TABLA DE ANEXOS

ANEXO A-1: NUMEROS ALEATORIOS PARA LA SECCION DE MOLIDO.....	273
ANEXO A-2: NUMEROS ALEATORIOS PARA LA SECCION DE MEZCLADO.....	274
ANEXO A-3: NUMEROS ALEATORIOS PARA LA SECCION DE EMBUTIDO.....	275
ANEXO A-4: NUMEROS ALEATORIOS PARA LA SECCION DE COCINADO.....	277
ANEXO D-1: CODIGO FUENTE, MODULO DE AUTENTICACION	282
ANEXO D-2: CODIGO FUENTE, GESTION DE PERSONAL, MODULO ADMINISTRACION	284
ANEXO D-3: CODIGO FUENTE, MANUAL DE FUNCIONES, MODULO ADMINISTRACION	288
ANEXO D-4: CODIGO FUENTE, BOLETAS DE PAGO, MODULO ADMINISTRACION	292
ANEXO D-5: CODIGO FUENTE, PLANILLA DE SUELDOS PATRONALES, MODULO ADMINISTRACION	297
ANEXO D-6: CODIGO FUENTE, GESTION DE PROVEEDORES, MODULO PROVISION	298
ANEXO D-7: CODIGO FUENTE, COMPRAS, MODULO PROVISION	302
ANEXO D-8: CODIGO FUENTE, PRODUCTOS TERMINADOS, MODULO PRODUCCION ..	304
ANEXO D-9: CODIGO FUENTE, INSUMOS, MODULO PRODUCCION	307
ANEXO D-10: CODIGO FUENTE, PROGRAMA MAESTRO DE PRODUCCION, MODULO PRODUCCION.....	311
ANEXO D-11: CODIGO FUENTE, GESTION DE CLIENTES, MODULO VENTAS	314
ANEXO D-12: CODIGO FUENTE, VENTAS, MODULO VENTAS.....	318
ANEXO D-13: CODIGO FUENTE, REGISTROS CONTABLES, MODULO CONTABILIDAD	326
ANEXO D-14: CODIGO FUENTE, CUENTAS, MODULO CONTABILIDAD.....	336
ANEXO D-15: CODIGO FUENTE, INDICADORES FINANCIEROS, MODULO CONTABILIDAD.....	340
ANEXO D-16: CODIGO FUENTE, ESTADO DE RESULTADOS, MODULO CONTABILIDAD	341
ANEXO D-17: CODIGO FUENTE, BALANCE GENERAL, MODULO CONTABILIDAD.....	343
ANEXO E-1: Nuevo sistema de facturacion.....	349