

UNIVERSIDAD MAYOR DE SAN ANDRES
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMATICA



PROYECTO DE GRADO

**“SISTEMA DE CONTROL DE VENTAS E INVENTARIOS PARA ALMACENES
DE ALUMINIOS UTILIZANDO DISPOSITIVOS MOVILES**

CASO: TECNICA DE ALUMINIO, VIDRIO Y SERVICIOS (TALVISER)”

PARA OPTAR AL TITULO DE LICENCIATURA EN INFORMATICA

MENCION: INGENIERIA DE SISTEMAS INFORMATICOS

POSTULANTE: GROVER GUTIERREZ VARGAS

TUTOR: LIC. JAVIER REYES PACHECO

ASESOR: LIC. RAMIRO FLORES ROJAS

LA PAZ - BOLIVIA

2015



**UNIVERSIDAD MAYOR DE SAN ANDRÉS
FACULTAD DE CIENCIAS PURAS Y NATURALES
CARRERA DE INFORMÁTICA**



LA CARRERA DE INFORMÁTICA DE LA FACULTAD DE CIENCIAS PURAS Y NATURALES PERTENECIENTE A LA UNIVERSIDAD MAYOR DE SAN ANDRÉS AUTORIZA EL USO DE LA INFORMACIÓN CONTENIDA EN ESTE DOCUMENTO SI LOS PROPÓSITOS SON ESTRICTAMENTE ACADÉMICOS.

LICENCIA DE USO

El usuario está autorizado a:

- a) visualizar el documento mediante el uso de un ordenador o dispositivo móvil.
- b) copiar, almacenar o imprimir si ha de ser de uso exclusivamente personal y privado.
- c) copiar textualmente parte(s) de su contenido mencionando la fuente y/o haciendo la referencia correspondiente respetando normas de redacción e investigación.

El usuario no puede publicar, distribuir o realizar emisión o exhibición alguna de este material, sin la autorización correspondiente.

TODOS LOS DERECHOS RESERVADOS. EL USO NO AUTORIZADO DE LOS CONTENIDOS PUBLICADOS EN ESTE SITIO DERIVARA EN EL INICIO DE ACCIONES LEGALES CONTEMPLADOS EN LA LEY DE DERECHOS DE AUTOR.

Dedicatoria

A mis amados y queridos padres Hipólito Gutierrez Beltrán y Dionicia Vargas Mamani por confiar y estar presente en todos momentos a mi lado brindándome todo su apoyo incondicional en todo momento, y también a mi primo Martín Gutierrez Mamani y a todas aquellas personas que están dentro de mi corazón.

AGRADECIMIENTOS

Agradezco a Dios por poner en mi camino a las personas que llenan mi corazón para darle fuerzas a mi espíritu, y me otorgan oportunidades para ser una persona mejor cada día.

A mis padres Hipólito y Dionicia por darme el incentivo necesario para seguir adelante, apoyándome en los momentos difíciles y por confiar en mí.

A mi tutor Lic. Javier Reyes Pacheco y a mi asesor Lic. Ramiro Flores Rojas docentes de la carrera, por brindarme su tiempo, comprensión y paciencia al realizar el seguimiento y culminación de este proyecto. Por brindarme las valiosas sugerencias para este proyecto y mi vida personal.

Al Gerente Propietario Lic. René Ramos Carlo por otorgarme la oportunidad de desarrollar el sistema e implementarlo en dicha empresa "TALVISER".

Agradecer a todo el personal de la empresa "TALVISER" por la colaboración y confianza depositada en mi persona.

A mi hermana Wilma que está en el cielo y que guía mi camino, y a mi primo Martín que con su confianza me muestra como un modelo a seguir a todos mis familiares.

A mis compañeros de la universidad por brindarme la ayuda y el compañerismo necesario para esos momentos difíciles.

A todos los docentes de la carrera por la guía y formación académica.

RESUMEN

El presente proyecto de grado pretende automatizar los procesos que la empresa realiza como las compras de materiales, venta de productos además de centralizar la información de materiales, usuarios, proveedores, para evitar duplicidad y poder acceder de manera inmediata a la información de los productos que la empresa comercializa, ya que estos procesos son de vital importancia para la empresa y es necesario controlar la información que se genera día a día. El Proyecto de Grado titulado **“SISTEMA DE CONTROL DE VENTAS E INVENTARIOS PARA ALMACENES DE ELUMINIOS UTILIZANDO DISPOSITIVOS MOVILES CASO: TECNICA DE ALUMINIO, VIDRIO Y SERVICIOS (TALVISER)”** ha sido desarrollado en las oficinas de **TALVISER** (Técnica de Aluminio, Vidrio y Servicios) con el objetivo de automatizar los procesos y optimizar los tiempos de producción de los procesos que se realizan en la empresa y mediante la tecnología de los dispositivos móviles se tratara de realizar los procesos en un instante.

Para el desarrollo del proyecto se utilizo la metodología SCRUM, que propone un modelo de proceso incremental, basado en iteraciones y revisiones continuas. También se utilizo en cada una de las 4 iteraciones la metodología UWE, que se especializa en el diseño de las Aplicaciones Web.

Para la conclusión del desarrollo del sistema Web se utilizo como herramienta primordial el lenguaje PHP, la aplicación Android, con el gestor de base de datos Mysql y con la ayuda del Servidor Apache para la función correcta del sistema.

INDICE

	Página
Capítulo I	
Marco Referencial	
1.1 Introducción.....	1
1.2 Antecedentes.....	2
1.2.1 Proyectos similares.....	2
1.2.3 Institucional.....	3
1.3 Planteamiento del problema.....	3
1.3.1 Formulación del problema.....	4
1.4 Objetivos.....	4
1.4.1 Objetivo general.....	4
1.4.2 Objetivos específicos.....	4
1.5 Justificación.....	5
1.5.1 Social.....	5
1.5.2 Técnica.....	5
1.5.3 Económica.....	6
1.6 Alcance.....	6
1.7 Metodología.....	6
Capítulo II	
Marco Teórico	
2.1 Marco Institucional.....	8
2.1.1 Misión.....	8
2.1.2 Visión.....	8
2.1.3 Organigrama.....	8
2.2 Inventario.....	9
2.2.1 Introducción.....	9
2.2.2 Conceptos básicos de inventario.....	9

2.2.3 Modelo de inventarios sin déficit.....	10
2.2.3.1 Análisis de ecuaciones.....	11
2.3 Almacenamiento.....	13
2.3.1 Conceptos básicos.....	13
2.3.2 Definición de almacenista.....	13
2.3.3 Función de los almacenes.....	13
2.3.4 Función de las existencias.....	14
2.3.5 Diseño de almacenes.....	14
2.3.6 Movimiento de materiales.....	14
2.3.7 Seguridad en almacenes.....	16
2.3.8 Almacenamiento de materiales.....	17
2.3.8.1 Estrategias y cajos o casilleros.....	17
2.3.8.2 Función de recepción.....	17
2.3.8.3 Técnicas de almacenamiento de materiales... 18	
2.4 Metodología de desarrollo de software.....	20
2.4.1 Introducción.....	20
2.4.2 Metodologías ágiles VS Tradicionales.....	20
2.5 SCRUM.....	21
2.5.1 Introducción.....	21
2.5.2 Elementos de scrum.....	23
2.5.2.1 Roles.....	24
2.5.2.2 Poda de requerimientos.....	25
2.5.2.3 Product backlog.....	25
2.5.2.4 Sprint.....	26
2.5.2.5 Planificación.....	27
2.5.2.5.1 Pre-juego.....	27
2.5.2.5.2 Juego.....	28
2.5.2.5.3 Post-juego.....	28
2.6 Proceso de desarrollo de SCRUM.....	33
2.6.1 Introducción.....	33

2.6.2 Proceso iterativo e incremental.....	34
2.6.3 Etapas del proceso de desarrollo.....	34
2.6.3.1 Planificación.....	34
2.6.3.2 Análisis.....	35
2.6.3.3 Diseño.....	35
2.6.3.4 Construcción y prueba.....	35
2.6.3.5 Implementación.....	36
2.6.4 EDT del proceso de desarrollo.....	36
2.6.5 Herramientas.....	37
2.6.5.1 Técnicas de relevamiento.....	37
2.6.5.2 Casos de uso.....	37
2.6.5.3 Diagrama de actividades.....	38
2.6.5.4 Diagrama de Entidad Relación (DER).....	38
2.7 Metodología UWE.....	38
2.7.1 Fases de la metodología UWE.....	39
2.8 Ingeniería de WEB.....	40
2.8.1 ¿Qué es un sistema Web?.....	41
2.9 Definición del esquema de base de datos.....	42
2.10 Calidad del software ISO 9126.....	43
2.11 Estudio de costos del sistema Web.....	46
2.11.1 Modelo Cocomo II.....	46
2.12 Seguridad del Sistema.....	48

Capítulo III

Marco Aplicativo

3.1 Introducción.....	50
3.2 Pre- Juego.....	51
3.2.1 Recopilación de requerimientos.....	51
3.2.2 Especificación de requerimientos.....	52
3.2.3 Definición del cronograma de trabajo.....	53

3.2.4 Análisis de Riesgos.....	53
3.3 Juego.....	54
3.3.1 Primera Iteración.....	55
3.3.2 Segunda Iteración.....	56
3.3.3 Tercera Iteración.....	57
3.3.4 Cuarta Iteración.....	58
3.4 Modelo de Casos de Uso.....	60
3.4.1 Diagrama de Casos de Uso General.....	60
3.4.2 Descripción de Casos de Uso.....	61
3.4.2.1 Caso de uso: Registro de usuario en el sistema.....	61
3.4.2.2 Diagrama de proceso: Registro de usuario en el sistema.....	63
3.4.2.3 Caso de uso: Registro de Proveedores.....	64
3.4.2.4 Diagrama de proceso: Registro de Proveedores.....	65
3.4.2.5 Caso de uso: Registro de Clientes.....	66
3.4.2.6 Diagrama de proceso: Registro de clientes.....	67
3.4.3. Modulo de Ventas.....	68
3.4.3.1 Diagrama de proceso: modulo de ventas.....	72
3.4.4. Modulo de Inventarios.....	72
3.4.4.1 Diagrama de proceso: Modulo de inventarios.....	76
3.4.5. Modulo de Compras.....	77
3.4.4.1 Diagrama de proceso: Modulo de compras.....	79
3.5. Modelo de contenidos (Clases).....	80
3.6. Modelo Entidad Relación.....	81
3.6.1 Diccionario de Datos.....	81
3.7. Modelo Físico.....	87
3.8. Modelo Navegacional.....	88
3.8.1. Modulo de Inventarios.....	88
3.8.2. Modulo de Compras.....	89
3.8.3. Modulo de Ventas.....	90
3.9. Diagramas de presentación.....	91
3.9.1. Modulo de Información de Ventas.....	91

3.9.2. Modulo de Información de Materiales y Herramientas.....	91
3.9.3. Modulo de Inventario.....	92
3.9.4. Modulo de Información de compras.....	93
3.10. Post-Juego.....	94
3.11. Modelo e Implementación del programa.....	94
3.11.1. Fase de Transición.....	94
3.11.2. Pruebas de aceptación.....	94
3.12. Diseño e interfaz.....	96
3.13. Fase de construcción.....	96

CAPITULO IV

METRICAS DE CALIDAD Y SEGURIDAD

4.1. Calidad.....	101
4.1.1. Técnica Web Site ISO 9126.....	101
4.1.1.1. Usabilidad.....	101
4.1.1.2. Mantenibilidad.....	102
4.1.1.3. Funcionalidad.....	103
4.1.1.3.1 Punto función.....	103
4.1.1.4. Confiabilidad.....	107
4.1.1.5. Portabilidad.....	107
4.1.2. Resultados.....	108
4.2. Seguridad.....	109
4.2.1. Autenticación.....	109
4.2.2. Logs o Registros.....	109
4.2.3. Base de Datos.....	110

CAPITULO V

ANALISIS DE COSTOS Y BENEFICIOS

5.1. COCOMO II.....	111
5.1.1. Punto función.....	111
5.2 Método de estimación de costo del sistema – CRI.....	112

CAPITULO VI
CONCLUSIONES Y RECOMENDACIONES

6.1. Conclusiones.....	116
6.2. Recomendaciones.....	116
BIBLIOGRAFIA.....	118
Referencias de Internet.....	119
ANEXOS.....	120

INDICE DE FIGURAS

	Página
Figura 2.1: Organigrama de la empresa de Técnica de Aluminio, Vidrio y Servicios (TALVISER).....	8
Figura 2.2: Modelo de inventario sin déficit.....	10
Figura 2.3: Componentes de Costo Total.....	12
Figura 2.4: El flujo de Scrum.....	23
Figura 2.5: Forma de funcionamiento de un Sprint.....	26
Figura 2.6: Desarrollo en pequeñas carreras en Scrum.....	27
Figura 2.7: Artefactos del Scrum.....	28
Figura 2.8: Iteración por fases.....	31
Figura 2.9: Estructura de División de Trabajo.....	37
Figura 3.1: Etapas del Sprint.....	51
Figura 3.2: Caso de Uso General.....	61
Figura 3.3: Caso de uso – Registro de usuario en el sistema.....	62
Figura 3.4: Diagrama de proceso – Registro de usuario en el sistema.....	64
Figura 3.5: Caso de uso – Registro de Proveedores.....	64
Figura 3.6: Diagrama de proceso – Registro de Proveedores.....	66
Figura 3.7: Caso de uso – Registro de Cliente.....	66
Figura 3.8: Diagrama de proceso – Registro de Cliente.....	68
Figura 3.9: Caso de uso – Modulo de Ventas.....	68
Figura 3.10: Diagrama de proceso – Modulo de Ventas.....	72
Figura 3.11: Caso de uso – Modulo de Inventarios.....	73
Figura 3.12: Diagrama de proceso – Modulo de Inventarios.....	77
Figura 3.13: Caso de uso – Modulo de Compras.....	78
Figura 3.14: Diagrama de proceso – Modulo de Inventarios.....	79
Figura 3.15: Diagrama Conceptual (Clases).....	80
Figura 3.16: Modelo Entidad Relación.....	81
Figura 3.17: Modelo Físico.....	87
Figura 3.18: Diseño navegacional – Ingreso al sistema.....	88

Figura 3.19: Diseño navegacional – Modulo de inventario.....	89
Figura 3.20: Diseño navegacional – Modulo de compras.....	90
Figura 3.21: Diseño navegacional – Modulo de venta.....	90
Figura 3.22: Diagrama de presentación – Información de ventas.....	91
Figura 3.23: Diagrama de presentación – información de material y herramientas...	92
Figura 3.24: Diagrama de presentación – Inventarios.....	93
Figura 3.25: Diagrama de presentación – Informe de compras.....	93
Figura 3.26: Inicio se sesión del sistema.....	97
Figura 3.27: Inicio se sesión del sistema en un dispositivo móvil.....	97
Figura 3.28: Menú principal del Administrador del sistema.....	98
Figura 3.29: Menú principal del Personal técnico del sistema.....	98
Figura 3.30: Ingreso de materiales o herramientas a almacenes.....	99
Figura 3.31: Ingreso de materiales o herramientas a almacenes en un dispositivo móvil	99
Figura 3.32: Ingreso de ventas al sistema.....	100
Figura 3.33: Ingreso de compras al sistema.....	100

INDICE DE TABLAS

	Página
Tabla 2.1: Diferencias entre metodologías ágiles y no ágiles.....	21
Tabla 2.2: Calidad ISO 9126.....	45
Tabla 2.3: Multiplicadores de esfuerzo para COCOMO II.....	48
Tabla 3.1 Requerimiento.....	53
Tabla 3.2: Análisis de Riesgo.....	54
Tabla 3.3: Primera Iteración.....	56
Tabla 3.4: Segunda Iteración.....	57
Tabla 3.5: Tercera Iteración.....	58
Tabla 3.6: Cuarta Iteración.....	59
Tabla 3.7: Descripción de los Actores.....	60
Tabla 3.8: Registrar un nuevo usuario.....	63
Tabla 3.9: Registro de proveedores.....	65
Tabla 3.10: Registro de clientes.....	67
Tabla 3.11: Registro de una nueva venta.....	69
Tabla 3.12: Registro del Terminado de un Producto.....	70
Tabla 3.13: Emitir factura.....	71
Tabla 3.14: Consulta del estado de un producto.....	71
Tabla 3.15: Registro de ingreso de un material / herramienta.....	74
Tabla 3.16: Solicitud de un material / herramienta.....	75
Tabla 3.17: Consulta de Stock de un material / herramienta.....	75
Tabla 3.18: Generar reporte de un material / herramienta.....	76
Tabla 3.19: Registro de la compra de un material / herramienta.....	79
Tabla 3.20: Diccionario de datos cliente.....	81
Tabla 3.21: Diccionario de datos venta.....	82
Tabla 3.22: Diccionario de datos contiene.....	82
Tabla 3.23: Diccionario de datos estado.....	82
Tabla 3.24: Diccionario de datos fabrica.....	83
Tabla 3.25: Diccionario de datos herramienta.....	83

Tabla 3.26: Diccionario de datos material.....	83
Tabla 3.27: Diccionario de datos obra.....	84
Tabla 3.28: Diccionario de datos producto.....	84
Tabla 3.29: Diccionario de datos proveedor.....	84
Tabla 3.30: Diccionario de datos compra.....	85
Tabla 3.31: Diccionario de datos realiza.....	85
Tabla 3.32: Diccionario de datos tipo_material.....	85
Tabla 3.33: Diccionario de datos trabajador.....	86
Tabla 3.34: Diccionario de datos usuario.....	86
Tabla 3.35: Diccionario de datos utiliza.....	86
Tabla 3.36: Caso de prueba – Gestión de información de ventas.....	95
Tabla 3.37: Caso de prueba – Gestión de información de inventarios.....	95
Tabla 3.38: Caso de prueba – Gestión de información de compras.....	96
Tabla 4.1: Encuesta de usabilidad del sistema.....	102
Tabla 4.2: Entradas para el cálculo de funcionalidad según punto función.....	104
Tabla 4.3: Cuenta total con factor de ponderación medio.....	104
Tabla 4.4: Ajuste de complejidad del punto función.....	106
Tabla 4.5: Valores de ajuste de complejidad.....	106
Tabla 4.6: Resultado de la evaluación de calidad.....	108
Tabla 4.7: Registro de ingresos correctos.....	109
Tabla 4.8: Registro de ingresos incorrectos.....	110
Tabla 5.1: Resultados de la estimación con COCOMO II.....	111
Tabla 5.2: Estimación de mantenimiento.....	112
Tabla 5.3: Calculo relación de Costo Beneficio.....	112
Tabla 5.4: Complejidad por tipo de dato.....	113
Tabla 5.5: Sumatoria de complejidad de datos por tabla.....	114
Tabla 5.6: Unidades de trabajo de las relaciones.....	115
Tabla 5.7: Unidades de trabajo de las relaciones.....	115

Capítulo I

Marco Referencial

1.4 Introducción

En la actualidad, la empresa necesita contar con estos medios tecnológicos para eliminar o reducir los procesos manuales puesto que generan pérdidas de tiempo, aumento de costos de operación, mala atención a los clientes y por consecuencia reducción de ganancias. La empresa también carece de los constantes cambios tecnológicos que les permite un mayor nivel de competencia, es por tal razón que las tareas manuales requieren un cambio automatizado utilizando también la tecnología de los dispositivos móviles para generar un almacenamiento de la información aún más rápida y confiable, los sistemas de información desempeñan un papel muy importante en el proceso de crecimiento de las organizaciones permitiendo la simplificación de proceso y adecuando control en sus operaciones y por consecuencia permiten un paso importante para su desarrollo.

Las redes informáticas es uno de los medios más eficientes para mantener la información actualizada ya que pueden compartir información entre varias computadoras y usuarios simultáneamente y de esta forma disponer de información oportuna que ayudara al mejor funcionamiento de las empresas, es de esta manera que muchas empresas optan por la implementación de sistemas informáticos para mantener su información en óptimas condiciones.

La empresa de Técnica de Aluminio, Vidrio y Servicios (TALVISER) es una institución que se dedica a la compra venta de aluminios y sus derivados como ser: ventanas, puertas, mamparas, etc. Además cuenta con un mercado consolidado en el departamento de La Paz, la información que genera la empresa es resguardada de manera manual, lo cual ocasiona pérdida de tiempo y pérdida de información y por consecuencia pérdidas económicas para la empresa.

El presente tiene como objetivo desarrollar un Sistema de Control de Ventas e Inventarios para Almacenes de Aluminios utilizando Dispositivos Móviles, proporcionando información precisa oportuna y confiable para la toma de decisiones.

1.5 Antecedentes

Para elaborar el presente proyecto se reviso los siguientes antecedentes:

1.5.1 Proyectos similares

En el proceso de desarrollo se realizó la consulta a los siguientes trabajos realizados en la carrera de Informática:

- Sistema de Control de Ventas e Inventarios Industriales The Firts, plantea el control de inventarios mediante técnicas UEPS, el método usado para el desarrollo de este trabajo es el Proceso Unificado de Desarrollo R.U.P. (Poma, 2006).
 - Sistema de Distribución, Ventas y Control de Inventarios para SOBOLMA LTDA, plantea el diseño e implementación del sistema de control de inventarios, distribución y ventas mediante inventarios perpetuos, para desarrollar este proyecto se utilizó la metodología R.U.P. (Condori, 2007).
 - Sistema de Inventarios para Laboratorios Crespal S.A. Regional Sucre, donde se tiene por objetivo realizar el control de inventarios mediante el uso de la métrica V3 de la ingeniería de software y utilizando como metodología de desarrollo el método R.U.P. (Paye, 2006).
 - Sistema de Control de Venta e Inventarios para la red de Farmacias Virgen del Carmen, donde se plantea en control de inventarios y ventas mediante uso de métricas de calidad y uso de inventarios físicos periódicos realizado a cada producto, la metodología que se utilizó para el desarrollo de este proyecto es el de Proceso Unificado de Desarrollo R.U.P. (Huallpa, 2009).
- Sistema de Información para el Control de Ventas e Inventarios de Sucursales. Empresa I.M.E.R. cuya metodología de desarrollo es la Metodología de Diseño Hipermedia Orientada a Objetos OOHDM (Ríos, 2012).

Dentro del proceso de administración el control de inventarios es de vital importancia y relevancia para cualquier empresa puesto que mediante el control se puede detectar y corregir errores en el registro de productos, reducir perdidas y lo más importante maximizar las utilidades de la empresa, de igual manera se podrá utilizar la tecnología de dispositivos móviles mediante el cual el registro de productos se la puede realizar de forma inmediata.

1.2.3 Institucional

La empresa de Técnica de Aluminio, Vidrio y Servicios (TALVISER) se creó el año 1984 con la finalidad de dar mayor comodidad y mejorar la distribución de productos elaborados con materiales de aluminios, vidrios para hogares y empresas. Actualmente la empresa cuenta con un mercado consolidado en el departamento de La Paz. La materia prima que se utiliza en las líneas de producción para la fabricación de los distintos productos son importados, la calidad de estos materiales tienen amplia garantía lo cual permite asegurar un buen producto con calidad de exportación.

Actualmente, el control de ventas e inventarios que se realiza en la empresa de Técnica de Aluminio, Vidrio y Servicios (TALVISER) es deficiente, el registro de los movimientos de entrada y salida de almacenes de la empresa son realizados manualmente lo cual ocasiona la acumulación de la información en gran cantidad. Además se tiene un retraso considerable para la realización de reportes de los productos vendidos y recibidos o que se encuentran en mal estado, además de estos reportes no ofrecen información precisa y confiable de los movimientos que se realiza.

1.6 Planteamiento del problema

El problema de almacenes, radica principalmente en la gran cantidad de información que se genera en el registro de ingreso, salida y ventas de los productos en los almacenes, dicha información se registra de forma manual, lo cual implica, en muchos casos, pérdida de información, registro erróneo, considerables atrasos en la entrega de informes y mala manipulación de la información lo cual tiene como consecuencia pérdidas económicas y pérdidas de potenciales clientes y como resultado reduce el nivel competitivo de la empresa frente a las demás empresas.

La empresa de Técnica de Aluminio, Vidrio y Servicios (TALVISER) no cuenta con un sistema de información que le permita realizar el control de ventas e inventarios, al no contar con un sistema de información no cuenta con información precisa, fidedigna y confiable que le permita tomar decisiones.

Entre las principales deficiencias que se tiene en las empresas se pueden citar las siguientes:

- Pérdida de información en registro de ingreso de materiales, herramientas y salida de productos.
- Control inadecuado de la compra de materiales y herramientas.
- Control inadecuado del inventario de los materiales y las herramientas.
- Control inadecuado de los productos vendidos.
- Pérdidas de tiempo en la entrega de reportes actualizados de control de stock.
- Registro inadecuado de los productos pedidos.
- Pérdida de dinero excesivo para la entrega de informes actualizados.

1.3.1 Formulación del problema

¿De qué manera se puede contar con un mejor control de la pérdida de información de los productos, el control inadecuado del inventario, la pérdida de tiempo en la entrega de reportes, registro inadecuado de los productos pedidos y la pérdida de dinero que le permita apoyar a la administración de la empresa de Técnica de Aluminio, Vidrios y Servicios (TALVISER)?

1.4 Objetivos

1.4.1 Objetivo general

Desarrollar un Sistema de Control de Ventas e Inventarios para Almacenes de Aluminios de la empresa de Técnica de Aluminio, Vidrio y Servicios (TALVISER) mediante el uso de Dispositivos Móviles.

1.4.2 Objetivos específicos

- Automatizar el registro de compra de materiales.
- Automatizar el registro de venta de productos.
- Control automático del inventario.
- Generar reportes de ventas.

1.5 Justificación

La existencia de normas y lineamientos específicos para el control de ventas e inventarios del almacén en las entidades privadas y/o públicas, hace necesario la creación de procedimientos acorde al reglamento interno de la institución.

1.5.1 Social

El presente desarrollo se justifica socialmente, porque beneficia al personal de Almacenes que será encargado de interactuar con el sistema, pues constituye una ayuda para la adecuada toma de decisiones, facilitar el uso de la información de una manera más rápida, eficiente, donde el sistema general facilitara los procesos que se realizan diariamente.

1.5.2 Técnica

Se justifica técnicamente, porque la microempresa cuenta con las herramientas informáticas en hardware y software suficiente para implementar y mantener el sistema de información para el control de ventas e inventarios, cuenta con los siguientes equipos:

Hardware:

- Un equipo de computadora Windows XP
- 3 dispositivos móviles
- Una impresora HP.

Software:

- Sistema Operativo Windows 7 profesional.
- Gestor de Base de Datos: Mysql.
- Lenguajes de programación: PHP, Java Script y Java.
- Plataforma de desarrollo: Dreamweaver, Eclipse.
- Servidores web: Appserver.

1.5.3 Económica

El gasto en la compra de equipos, tendrá sus frutos a futuro siendo una inversión que cooperara a la institución en la eficiencia y eficacia brindando mejor servicio a la población. Con la implementación de este nuevo sistema de control de ventas e inventario para almacenes de aluminios utilizando dispositivos móviles a futuro podrá planificar mejor su forma de manejo en los almacenes.

1.8 Alcance

El Sistema de Control de Ventas e Inventarios para Almacenes con el uso de Dispositivos Móviles serán utilizados en Almacenes de la empresa de Aluminio, Vidrio y Servicios (TALVISER) que se dedica a la compra y venta de aluminios, vidrios y productos derivados.

El sistema cuenta con los siguientes módulos:

- Registro de venta de productos.
- Registro de materiales y herramientas en almacén, mediante el uso de dispositivos móviles.
- Registro de compra de materiales y herramientas.
- Consultas de cantidad de los productos vendidos
- Consulta de cantidad de materiales y herramientas en el Almacén, mediante el uso de dispositivos móviles.

1.9 Metodología

Para el desarrollo del presente trabajo se utiliza la metodología SCRUM, dicha investigación es empleada a problemas concretos en circunstancias y características concretas. Precisamente la investigación se dirige a su aplicación inmediata y no al desarrollo de teorías.

El método descriptivo comprende la descripción, registro, análisis e interpretación de la problemática actual. Trabaja sobre realidades de hechos, y su característica fundamental es la de presentar una interpretación correcta.

Esta metodología tiene las siguientes etapas:

Descripción del Problema

- Definición y formulación del problema
- Supuestos en que se basa el problema
- Marco teórico
- Selección de técnicas de recolección de datos
- Categorías de datos, a fin de facilitar relaciones.
- Verificación de validez del instrumento.
- Descripción, análisis e interpretación de datos.

Mediante los pasos expuestos anteriormente se utiliza el método de análisis, se logra caracterizar un objeto de estudio o una situación concreta, señalar sus características y propiedades. Combinada con ciertos criterios de clasificación sirve para ordenar, agrupar o sistematizar los objetos involucrados en el trabajo indagatorio (Tamayo, 1995).



Capítulo II

Marco Teórico

2.1 Marco Institucional

La Empresa de Técnica de Aluminio, Vidrio y Servicio (TALVISER) fue creada en 1984 en el ciudad de La Paz se encuentra ubicada en la zona de San Pedro cerca a la plaza Líbano en la calle Venancio Burgoa 983.

2.1.1 Misión

La empresa de Técnica de Aluminio, Vidrio y Servicios (TALVISER), su misión principal es la producción de ventanas, puertas, división de ambientes, fachadas flotantes, box de baños, techos, etc. En general es la Carpintería en Aluminio y estructuras de vidrio para las diferentes empresas o clientes que necesiten de este servicio.

2.1.2 Visión

La empresa de Técnica de Aluminio, Vidrio y Servicios (TALVISER), tiene como visión realizar un trabajo garantizado para la satisfacción del cliente o las empresas a los que brinda sus servicios.

2.1.3 Organigrama

La empresa está organizada como muestra la figura 1.

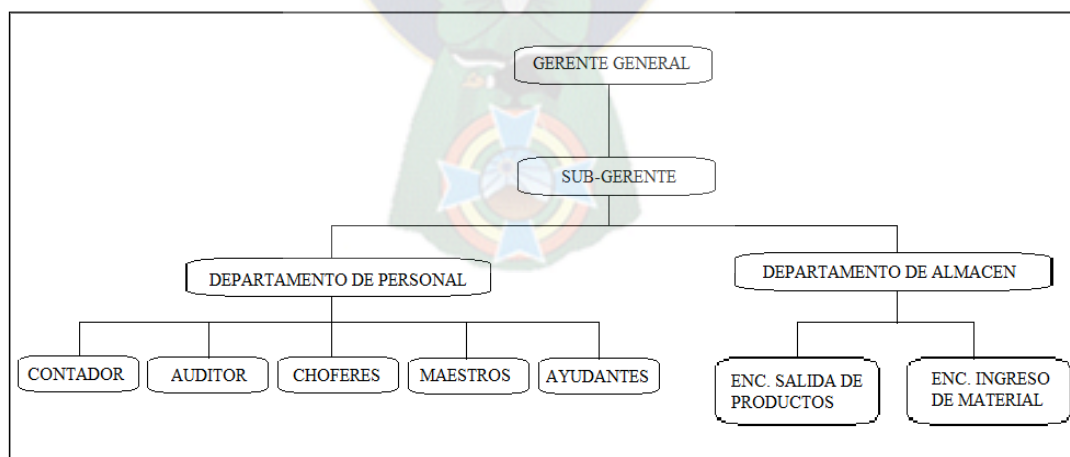


Figura 2.1: Organigrama de la empresa de Técnica de Aluminio, Vidrio y Servicios (TALVISER)

Fuente: [TALVISER]

2.2 Inventario

2.2.1 Introducción

La base de toda empresa es la compra o venta de servicios, ahí la importancia del manejo de inventario por parte de la misma. Este manejo contable permite a la empresa mantener el control oportuno de los productos (en nuestro caso el almacenamiento de aluminios).

Los inventarios comprenden, además de las materias primas, productos en proceso y productos terminados o mercancías para la venta, bienes reparados para la prestación de servicios.

Ahora bien, el inventario constituye las partidas del activo corriente que están listas para la venta, es decir, toda aquella mercancía que posee una empresa en el almacén valorada al costo de adquisición, para la venta o actividades productivas. Por medio del siguiente trabajo se darán a conocer algunos conceptos básicos de todo lo relacionado a los inventarios en una empresa, métodos, sistema y control.

En contabilidad, el termino inventario significa una existencia de bienes con propósitos específicos según la naturaleza de la empresa.

2.2.2 Conceptos básicos de inventario

Los inventarios son un puente de unión entre la producción y las ventas. En una empresa manufacturera el inventario equilibra la línea de producción si algunas maquinas operan a diferentes volúmenes de otras, pues una forma de compensar este desequilibrio es proporcionando inventarios temporales o bancos. Los inventarios de materias primas, productos semi terminados y productos terminados absorben la holgura cuando fluctúan las ventas. Estos tienden a proporcionar un flujo constante de producción, facilitando su programación.

Los inventarios de materia dan flexibilidad al proceso de compra de la empresa.

Sin ellos en la empresa existen una situación “de la mano a la boca”, comprándose la materia prima estrictamente necesaria para mantener el plan de producción, es decir comprando y consumiendo.

2.2.3 Modelo de inventarios sin déficit

Este modelo tiene como base el mantener un inventario sin falta de productos para desarrollar las actividades de cualquier empresa. Este es un modelo de inventarios que se encuentra basado en las siguientes suposiciones:

- La demanda se efectúa a tasa constante.
- El reemplazo es instantáneo (la tasa de reemplazo es infinita).
- Todos los coeficientes de costos son constantes.

En este modelo no se permite la falta de productos para la venta, es decir, una empresa que maneje este modelo de inventario no se puede quedar sin mercancías para la venta.

En la siguiente figura se ilustra esquemáticamente este modelo.

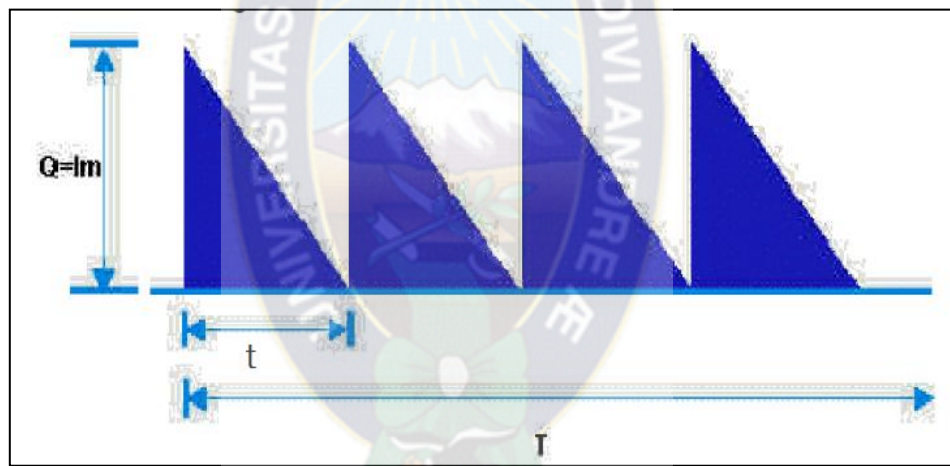


Figura 2.2: Modelo de inventario sin déficit

Fuente: [Teoría de Inventarios o stock.pdf]

Símbolos:

Q = Cantidad optima a pedir

Im = Inventario Máximo

t = Periodo entre pedidos

T = Periodo de Planeación

En este modelo se representan iguales el inventario máximo y la cantidad económica pedida.

Cabe mencionar que esto no siempre es verdadero.

El costo total para un periodo en este modelo está conformado por tres componentes de costo:

- a. Costo unitario del producto (C1).
- b. Costo de ordenar una compra (C2).
- c. Costo de mantener un producto en almacén (C3).

El costo para un periodo estará conformado de la siguiente manera:

Costo por periodo = [Costo unitario por periodo] + [Costo de ordenar un pedido] + [Costo de mantener el inventario en un periodo]

El costo total para el periodo de planeación estará conformado de la manera siguiente:

Costo total = Costo por periodo x Numero de pedidos a realizar.

2.2.3.1 Análisis de ecuaciones

Costo unitario por periodo.

El costo unitario por periodo simplemente es el costo de la cantidad óptima a pedir.

$$C1 * Q$$

Costo de ordenar una compra.

Puesto que solo se realiza una compra en un periodo el costo de ordenar una compra está definido por:

$$C2$$

El inventario promedio por periodo es $[Q / 2]$. Por consiguiente el costo de mantenimiento del inventario por periodo es:

$$(Q/2) C3 * t$$

Para determinar el costo en un periodo se cuenta con la siguiente ecuación:

$$\text{Costo } (Q^*) = C1 * t + C2 + C3 * t (Q/2)$$

El tiempo de un periodo se expresa de la siguiente manera:

$$t = Q/D$$

Nota: La demanda del artículo en un periodo de planeación se define con la letra D.

El número de periodos se expresa de la manera siguiente:

$$N = D/Q$$

Si se desea determinar el costo total en el periodo de planeación (T) se multiplica, el costo de un periodo por el número de interperiodos (t) que contenga el periodo de planeación. Para determinar este costo se aplica la siguiente ecuación:

$$\text{Costo Total} = \text{Costo } (Q^*) t$$

Otra manera de representar el costo total para el periodo de planeación es por medio de la siguiente ecuación:

$$\text{Costo Total} = C_1 * D + C_2 * (D/Q) + C_3 * (Q/2)$$

Cuando los componentes del costo total se representan gráficamente se obtiene un punto óptimo (de costo mínimo).

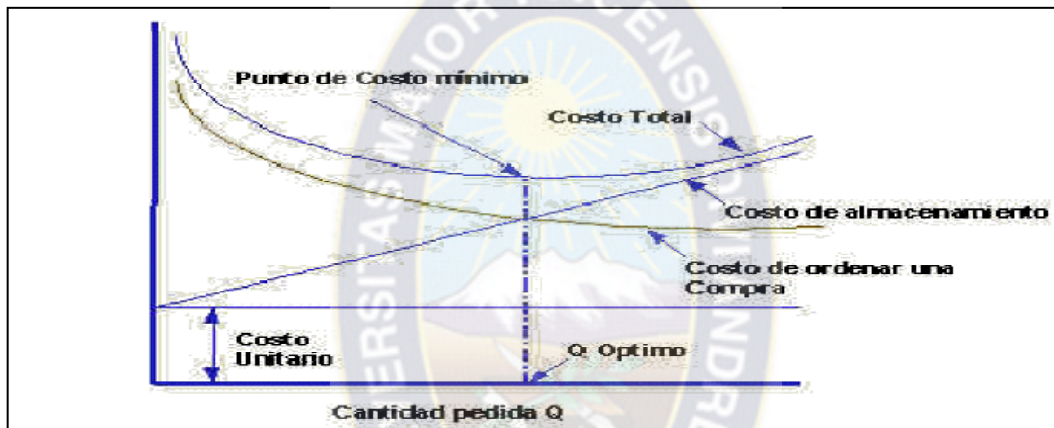


Figura 2.3: Componentes de Costo Total

Fuente: [Teoría de Inventarios o stock.pdf.]

Una forma de determinar la cantidad óptima a pedir es suponer diversos valores de Q y sustituir en la ecuación anterior hasta encontrar el punto de costo mínimo. Un procedimiento más sencillo consiste en derivar la ecuación del costo total con respecto a Q e igualar la derivada a cero.

$$dC/dQ = C_1 * D + C_2 * (D/Q) + C_3 * (Q/2)$$

Al resolver esta derivada tenemos la ecuación para determinar la cantidad óptima a pedir.

$$Q = \sqrt{2 * C_2 * D / C_3}$$

Esta ecuación ocasiona un costo mínimo y tiene como base un balance entre los dos costos variables (costo de almacenamiento y costo de compra) incluidos en el modelo. Cualquier otra cantidad pedida ocasiona un costo mayor (Teoría de Inventarios o stock.pdf, 2007).

2.3 Almacenamiento

2.3.1 Conceptos básicos

Es el sitio o lugar destinado a guardar, proteger, custodiar y despachar toda clase de materiales y/o artículos. La palabra Almacén **proviene de Almagacen**, vocablo árabe que significa "Tesoro", por tal significación se identifica el almacén y sus mercancías como un tesoro muy valioso, apreciado que se debe guardar, custodiar y cuidar para el futuro.

La formulación de una política de inventario para un departamento de almacén depende de la información respecto a tiempos de entrega, disponibilidades de materiales, tendencias en los precios y materiales de compras, son las mejores fuentes de información.

Esta función controla físicamente y mantiene todos los artículos inventariados, se deben establecer resguardo físicos adecuados para proteger los artículos de algún daño de uso innecesario debido a procedimientos de rotación de inventarios defectuosos de rotación de inventarios defectuosos y a robos. Los registros de deben mantener, lo cual facilitan la localización inmediata de los artículos.

2.3.2 Definición de almacenista

- Es aquella persona capaz de guardar, proteger, custodiar y despachar toda clase de materiales y/o artículos.
- Tiene que contar, medir y pesar la mercancía.
- Se debe delegar en una sola persona la responsabilidad de la descarga.
- Comparar el resultado del conteo con los documentos respectivos (A la hora cuando se despacha y cuando se recibe mercancía, observar la factura).
- Debe utilizar equipos móviles en las descarga (Seguridad Industrial).
- El Almacenista debe evitar la indisciplina.

2.3.3 Función de los almacenes

- Mantienen las materias primas a cubierto de incendios, robos y deterioros.
- Permitir a las personas autorizadas el acceso a los materiales almacenados.
- Mantiene informado al departamento de compras, sobre las existencias reales de materia prima.

- Lleva en forma minuciosa controles sobre las materias primas (entradas y salidas)
- Vigila que no se agoten los materiales (máximos – mínimos).

2.3.4 Función de las existencias

- Garantiza el abastecimiento e invalida los efectos de:
 - ✓ Retraso en el abastecimiento de materiales.
 - ✓ Abastecimiento parcial
- Compra o producción en lotes económicos.
- Rapidez y eficacia en atención a las necesidades.

2.3.5 Diseño de almacenes

El diseño de almacenes es una actividad especializada, sobre la que ofrecemos algunos comentarios generales:

- El uso de planos arquitectónicos permite visualizar los almacenes, lo cual facilita su diseño.
- Los almacenes deben ser diseñados en términos de metros cúbicos y no de metros cuadrados, ya que el espacio vertical puede ser convenientemente utilizado de manera de aumentar al máximo el volumen de almacenamiento sin aumentar la superficie requerida.
- El almacén debe planificarse de manera que el ambiente de trabajo resulte agradable y se facilite el crecimiento futuro.
- La buena iluminación agiliza la localización de materiales, evita robos y reduce accidentes.
- La pérdida de los inventarios puede ser fatal para una empresa. Un buen sistema de detección y extinción de incendios protege los inventarios.

2.3.6 Movimiento de materiales

El movimiento de materiales es el proceso que estos siguen desde su llegada a los almacenes hasta su despacho.

- Los materiales deben tener localizaciones físicas específicas que permitan las actividades normales de almacén, como son la entrada y salida de inventario y la realización de inventarios físicos, para permitir un adecuado movimiento de materiales es aconsejable:
- Planificar, siempre que sea posible, un flujo de materiales en línea recta (recepción, almacenamiento, despacho).
- Separar las áreas de recepción y despacho de materiales.
- Considerar áreas separadas físicamente para materiales dañados, reparables o no, que esperan ser enviados a reparar o que acaban de ser reparados.
- Ubicar los materiales que requieren condiciones especiales de almacenamiento en áreas especialmente acondicionadas para tal fin.
- Diseñar los pasillos de circulación (aunque no sean espacios productivos) lo suficientemente anchos para permitir la circulación de montacargas y otros vehículos. Todos los pasillos secundarios deben fluir a un pasillo principal.
- Indicar claramente el sentido y las velocidades máximas de circulación en los pasillos, a fin de reducir el riesgo de accidentes. Un buen sistema de señalización es una excelente inversión.
- Establecer áreas especialmente protegidas para materiales valiosos.
- Construir las plataformas de descarga a la altura de los vehículos de transporte típicos.
- Las puertas de acceso y salida de los almacenes deben ser fácilmente manipulables por los operadores de los vehículos de movimiento de materiales.
- Disponer de suficiente espacio en el área de recepción de materiales para el control de calidad.

2.3.7 Seguridad en almacenes

Además del problema de protección de incendios, deberá prestársele atención especial al problema de los hurtos, considerado junto a la obsolescencia, el principal elemento en los costos de almacenamiento. Algunas reglas simples que ayudan a reducir este problema son las siguientes:

- Diseñar con la seguridad en mente: asegurarse de que las áreas de almacenes estén cercadas o protegidas, de manera de minimizar la presencia de intrusos; iluminar adecuadamente las áreas de almacenes; no permitir la entrada de usuarios y empleados que no tengan que ver con el movimiento de materiales o inventario de los mismos al área de almacenes.
- Crear controles de acceso: contratar o conformar un cuerpo de vigilancia; verificar los vehículos que entren y salgan; realizar inspecciones no anunciadas de los casilleros, escritorios y vehículos del personal.
- Hacer inventarios periódicos o permanentes para verificar las pérdidas de material.
- Involucrar al personal en el control de los materiales: concienciarlo con charlas y concursos sobre el problema; aceptar sugerencias; informarles sobre los resultados de los inventarios, en lo que a pérdidas se refiere y sobre todo, hacerle sentirse parte importante de la organización.

Un simple candado puede ser la diferencia entre utilidades o pérdidas. Debe destinarse un área específica para guardar las mercancías y materia prima que vamos a vender. El acceso al almacén debe ser restringido a una o dos personas como máximo y sólo debe entrar más personal cuando sea necesario llevar a cabo inventarios físicos.

Los materiales de alto costo deben guardarse bajo llave si no se van a utilizar en el día, así como también si se encuentran en congelación.

Existen negocios en los que se han implementado el uso de cámaras, vigilantes, costosos sistemas de cómputo, etc. El mejor control sin lugar a duda es aquel que le llamaremos "democrático": el costo del material faltante se reparte entre las personas que tengan acceso al área de almacén y/o preparaciones, resultando ser prácticamente todo el

personal. Pero esto no puede llevarse a cabo sino tenemos conocimiento de si en realidad nos falta o no material, es decir, sin registros de control.

Desafortunadamente es usual que sean los mismos empleados (o aún los clientes) quienes lleven a cabo el robo hormiga, otro factor que lleva al aumento de costos por falta de control del inventario.

2.3.8 Almacenamiento de materiales

Existen distintas posibilidades, según las características de los materiales que van a ser almacenados. La más empleada es la estantería, que puede ser obtenida en variedad de tamaños: cerradas (para protección adicional, pero con dificultades de acceso), con entrepaños para la colocación de materiales de volumen intermedio, con vigas horizontales para paletas, con vigas inclinadas para elementos cilíndricos, solo con laterales para perfiles y elementos longitudinales, etc.

La ubicación física de los materiales en los almacenes debe ser establecida de manera que permita la localización rápida y sin errores de los materiales.

2.3.8.1 Estrategias y cajos o casilleros

Puede aumentar mucho la eficiencia total y la flexibilidad de los procedimientos que se emplea en el almacenamiento mediante el uso de un equipo adecuado. En algunas empresas, el almacén incluye las estanterías, los casilleros, compartimientos, entre otros, que se hacen con madera ordinaria y chapada.

2.3.8.2 Función de recepción

La recepción adecuada de materiales y de otros artículos es de vital importancia, ya que una gran parte de las empresas tienen como resultado de su experiencia centralizada la recepción total bajo un departamento único, las excepciones principales son aquellas grandes empresas con plantas múltiples. La recepción está estrechamente ligada a la compra.

Al recibir un embarque: Se le someterá a verificación para comprobar si está en orden y en buenas condiciones, si el contenedor está dañado o no se recibió el número de paquetes requeridos.

De Manera Similar: El material que se recibe en una instalación de la empresa también debe ser sometido a una inspección preliminar, antes de introducirles en el área de

almacenamiento, en el caso de que en la inspección inicial se detecte materiales de calidad inferior o en malas condiciones se le debe rechazar.

2.3.8.3 Técnicas de almacenamiento de materiales

Es un hecho que el desorden en almacén o en el área de trabajo provoca graves pérdidas a la empresa. Podemos desconocer que tenemos existencias en almacén y comprar demás o bien, simplemente no encontrar material que necesitamos y este pierda su vida útil.

El almacenamiento de materiales depende de la dimensión y características de los materiales. Estos pueden requerir una simple estantería hasta sistemas complicados, que involucran grandes inversiones y tecnologías complejas. La elección del sistema de almacenamiento de materiales depende de los siguientes factores:

- Espacio disponible para el almacenamiento de los materiales.
- Tipos de materiales que serán almacenados.
- Número de artículos guardados.
- Velocidad de atención necesaria.
- Tipo de embalaje.

Las principales técnicas de almacenamiento de materiales son:

- ***Carga unitaria.*** Se da el nombre de carga unitaria a la carga constituida por embalajes de transporte que arreglan o acondicionan una cierta cantidad de material para posibilitar su manipulación, transporte y almacenamiento como si fuese una unidad. Se usan pallets (plataformas), que es una plataforma de madera esquematizado de diversas dimensiones. Sus medidas convencionales básicas son 1,100mm x 1,100mm como patrón internacional para adecuarse a los diversos medios de transporte y almacenamiento.
- ***Las plataformas pueden clasificarse de la siguiente manera:***
 - ✓ En cuanto al número de entradas en: plataformas de 2 y de 4 entradas.
 - ✓ Plataforma de 2 entradas: se usan cuando el sistema de movimiento de materiales no requieren utilizar equipos de manejo de materiales.

- ✓ Plataforma de 4 entradas: Son usados cuando el sistema de movimiento de materiales requiere utilizar equipos de maniobras.

Otros medios de almacenamiento:

- **Cajas o cajones.** Es la técnica de almacenamiento ideal para materiales de pequeñas dimensiones, como tornillos, anillos o algunos materiales de oficina, como plumas, lápices, entre otros. Algunos materiales en procesamiento, semiacabados pueden guardarse en cajas en las secciones productivas pueden ser de metal, de madera o de plástico con tamaños muy variados.
- **Estanterías:** Es una técnica de almacenamiento destinada a materiales de diversos tamaños y para el apoyo de cajones y cajas estandarizadas. Las estanterías pueden ser de madera o perfiles metálicos, de varios tamaño y dimensiones, los materiales que se guardan en ellas deben estar identificadas y visibles, la estanterías constituye el medio de almacenamiento más simple y económico. Es la técnica adoptada para piezas pequeñas y livianas cuando las existencias no son muy grandes.
- **Columnas:** Las columnas se utilizan para acomodar piezas largas y estrechas como tubos, barras, correas, varas gruesas, flejes entre otras. Pueden ser montadas en rueditas para facilitar su movimiento, su estructura puede ser de madera o de acero.
- **Apilamientos:** Se trata de una variación de almacenamiento de cajas para aprovechar al máximo el espacio vertical. Las cajas o plataformas son apilados una sobre otras, obedeciendo a una distribución equitativa de cargas, es una técnica de almacenamiento que reduce la necesidad de divisiones en las estanterías, ya que en la práctica, forma un gran y único estante. El apilamiento favorece la utilización de las plataformas y en consecuencia de las pilas, que constituyen el equipo ideal para moverlos. La configuración del apilamiento es lo que define el número de entradas necesarias a las plataformas (controymanejodeinventarios.pdf, 1999).

2.4 Metodología de desarrollo de software

2.4.1 Introducción

Durante el transcurso de los años 90 el ambiente cambiante y turbulento era cada vez más la regla que la excepción. Por lo tanto surgió la necesidad de desarrollar metodologías livianas y maniobrables que pudieran operar en ese ambiente turbulento. Estas metodologías son conocidas colectivamente hoy en día como “metodologías ágiles”.

2.4.2 Metodologías ágiles VS Tradicionales

A continuación vamos a enumerar las principales diferencias respecto de las metodologías tradicionales (no ágiles).

El siguiente cuadro recoge esquemáticamente estas diferencias que no se refieren sólo al proceso en sí, sino también al contexto de equipo y organización que es más favorable a cada una de estas filosofías de desarrollo de software.

Metodologías Ágiles	Metodologías Tradicionales
La planificación del trabajo sólo comprende el ciclo en el que se está trabajando (normalmente 30 días).	Trabajo y gestión guiada por un plan general del proyecto que comprende todo su ciclo de desarrollo.
Descubrimiento progresivo de requisitos, e incorporación de cambios en cualquier iteración del desarrollo.	Conocimiento detallado de los requisitos antes de comenzar el diseño del proyecto.
“Refactorización” de código como modelo de trabajo compatible con el punto anterior.	“Hacerlo bien a la primera”. Evitar la re-codificación y el re-trabajo que supone una pérdida de eficiencia.
Comunicación directa entre los integrantes del equipo (incluidos cliente y usuarios) prefiriendo la verbal directa.	Comunicación formal según el plan de comunicación del proyecto.
Equipos auto-gestionados.	Gestión de equipos y personas centralizada en el gestor del proyecto.

No existe contrato tradicional o al menos es bastante flexible.	Existe un contrato prefijado.
El cliente es parte del equipo de desarrollo.	El cliente interactúa con el equipo de desarrollo mediante reuniones.
Grupos pequeños (hasta 20 integrantes) y trabajando en el mismo sitio.	Grupos grandes y posiblemente distribuidos.
Pocos artefactos.	Más artefactos.
Pocos roles.	Más roles.
Menos énfasis en la arquitectura del software.	La arquitectura del software es esencial y se expresa mediante modelos.

Tabla 2.1: Diferencias entre metodologías ágiles y no ágiles.

Fuente: [masyxp.pdf.]

Las metodologías ágiles están revolucionando la manera de producir software, y a la vez generando un amplio debate entre sus adeptos y quienes por prejuicio no las ven como alternativa a las metodologías tradicionales.

Aunque los creadores de las metodologías ágiles han suscrito el manifiesto ágil, y todas ellas coinciden con los principios enunciados anteriormente, cada una tiene características propias y hace hincapié en algunos aspectos más específicos.

Las metodologías ágiles más populares son: XP (Extreme Programming – Programación Extrema), Cristal y Scrum. Por ser esta última la elegida para desarrollar el presente trabajo mencionamos los tres principales motivos que nos llevaron a su elección.

- Sirve para gestionar proyectos de cualquier tipo, no solamente tecnológicos.
- Deja un vacío en la parte de definiciones del área de ingeniería lo que permite una elaboración propia para completarla.
- Tiene escasa documentación por lo que se piensa que este trabajo puede resultar en un aporte significativo (masyxp.pdf, 2004).

2.5 SCRUM

2.5.1 Introducción

Scrum es una metodología ágil de gestión de proyectos cuyo objetivo primordial es elevar al máximo la productividad de un equipo. Reduce al máximo la burocracia y actividades no orientadas a producir software que funcione y produce resultados en periodos muy breves de tiempo. Como método, Scrum enfatiza valores y prácticas de gestión, sin pronunciarse sobre requerimientos, prácticas de desarrollo, implementación y demás cuestiones técnicas. Más bien delega completamente en el equipo la responsabilidad de decidir la mejor manera de trabajar para ser lo más productivos posibles.

La palabra Scrum procede de la terminología del juego de rugby, donde designa al acto de preparar el avance del equipo en unidad pasando la pelota a uno y otro jugador. Igual que el juego, Scrum es adaptable, ágil, auto-organizante y con pocos tiempos muertos.

Scrum fue desarrollado por Jeff Sutherland y elaborado más formalmente por Ken Schwaber. Poco después Sutherland y Schwaber se unieron para refinar y extender Scrum. Se la ha llegado a conocer como una herramienta de hiperproductividad. Schwaber se dio cuenta entonces de que un proceso necesita aceptar el cambio, en lugar de esperar predictibilidad. Se enfoca en el hecho de que procesos definidos y repetibles sólo funcionan para atacar problemas definidos y repetibles con gente definida y repetible en ambientes definidos y repetibles. Toma el cambio como una forma de entregar al final del desarrollo algo más cercano a la verdadera necesidad del Cliente. Puede ser aplicado teóricamente a cualquier contexto en donde un grupo de gente necesita trabajar junta para lograr una meta común.

Se basa en los principios ágiles:

- Privilegiar el valor de la gente sobre el valor de los procesos.
- Entregar software funcional lo más pronto posible.
- Predisposición y respuesta al cambio.
- Fortalecer la comunicación y la colaboración.
- Comunicación verbal directa entre los implicados en el proyecto.
- Simplicidad; supresión de artefactos innecesarios en la gestión del proyecto.

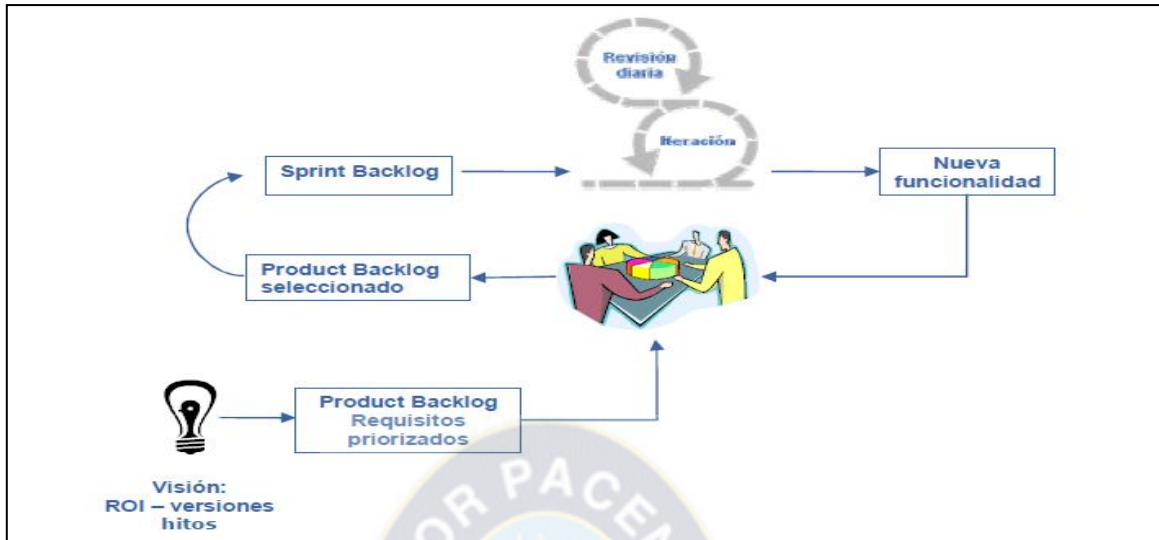


Figura 2.4: El flujo de Scrum

Fuente: [e-metodoagilscrum-100513145255-phpapp02.pdf.]

2.5.2 Elementos de scrum.

- Roles
 - ✓ Product Owner
 - ✓ Scrum Master
 - ✓ Team (Equipo)
- Poda de requerimientos
- Product Backlog
- Sprint
 - ✓ Planificación
 - ✓ Sprint Backlog
 - ✓ Scrum
 - ✓ Estimaciones
 - ✓ Builds continuos
 - ✓ Revisión del Sprint
 - ✓ Reunión retrospectiva
- Valores
- Foco, comunicación, respeto y coraje.

2.5.2.1 Roles.

La dimensión del equipo total de Scrum no debería ser superior a veinte personas. Si hay más, lo más recomendable es formar varios equipos. No hay una técnica oficial para coordinar equipos múltiples, pero se han documentado experiencias de hasta 800 miembros, divididos en Scrums de Scrum, definiendo un equipo central que se encarga de la coordinación, las pruebas cruzadas y la rotación de los miembros.

Scrum tiene una estructura muy simple. Todas las responsabilidades del proyecto se reparten en 3 roles:

➤ ***Product owner (Dueño del producto).***

Representa a todos los interesados en el producto final.

Sus áreas de responsabilidad son:

- ✓ Financiación del proyecto
- ✓ Requisitos del sistema
- ✓ Retorno de la inversión del proyecto
- ✓ Lanzamiento del proyecto

Es el responsable oficial del proyecto, gestión, control y visibilidad de la lista de acumulación o lista de retraso del producto (Product Backlog).

Toma las decisiones finales de las tareas asignadas al registro y convierte sus elementos en rasgos a desarrollar.

➤ ***Scrum Master (Líder del proyecto).***

Responsable del proceso Scrum, de cumplir la meta y resolver los problemas. Así como también, de asegurarse que el proyecto se lleve a cabo de acuerdo con las prácticas, valores y reglas de Scrum y que progrese según lo previsto.

Interactúa con el cliente y el equipo. Coordina los encuentros diarios, y se encarga de eliminar eventuales obstáculos. Debe ser miembro del equipo y trabajar a la par.

➤ ***Team (Equipo).***

Responsable de transformar el Backlog de la iteración en un incremento de la funcionalidad del software. Tiene autoridad para reorganizarse y definir las acciones necesarias o sugerir remoción de impedimentos.

- ✓ Auto-gestionado
- ✓ Auto-organizado

✓ Multi-funcional

La dimensión del equipo total de Scrum no debería ser superior a veinte.

El número ideal es diez, más o menos dos. Si hay más, lo más recomendable es formar varios equipos. No hay una técnica oficial para coordinar equipos múltiples, pero se han documentado experiencias de hasta 800 miembros, divididos en Scrums de Scrums, definiendo un equipo central que se encarga de la coordinación, las pruebas cruzadas y la rotación de los miembros.

Scrum diferencia claramente entre estos dos grupos para garantizar que quienes tienen la responsabilidad tienen también la autoridad necesaria para poder lograr el éxito, y que quienes no tienen la responsabilidad, los observadores externos, no produzcan interferencias innecesarias.

Comprometidos en el proyecto	Implicados en el proyecto
<ul style="list-style-type: none">➤ Dueño del producto➤ Equipo➤ Scrum Master	<ul style="list-style-type: none">➤ Marketing➤ Comercial➤ Etc.

2.5.2.2 Poda de requerimientos

La primera actividad es armar una lista exhaustiva de los requerimientos originales del sistema. Luego se procede a ver qué requerimientos son realmente necesarios, cuáles pueden posponerse y cuáles eliminarse.

Para ello debe identificarse un representante con capacidad de decisión, priorizar los requerimientos en base a su importancia y acordar cuáles son los prioritarios para la fecha de entrega.

La poda de requerimientos es una buena práctica implícita en modelos ágiles, se hace lo que el cliente realmente desea, no más.

2.5.2.3 Product backlog

Con los requerimientos priorizados y podados armamos el Backlog de Producto. Este es una forma de registrar y organizar el trabajo pendiente para el producto (actividades y requerimientos).

Es un documento dinámico que incorpora constantemente las necesidades del sistema. Por lo tanto, nunca llega a ser una lista completa y definitiva. Se mantiene durante todo el ciclo de vida (hasta la retirada del sistema) y es responsabilidad del Product Owner.

2.5.2.4 Sprint

Scrum está basado en el control empírico de procesos. Se utiliza cuando la capacidad de predicción es vaga, la incertidumbre alta o el proceso es demasiado complejo para ser modelado y definido.

En el enfoque empírico de control de procesos se establecen reglas simples y se crea una disciplina de inspección frecuente para adaptarse rápidamente a situaciones imprevistas o problemas.

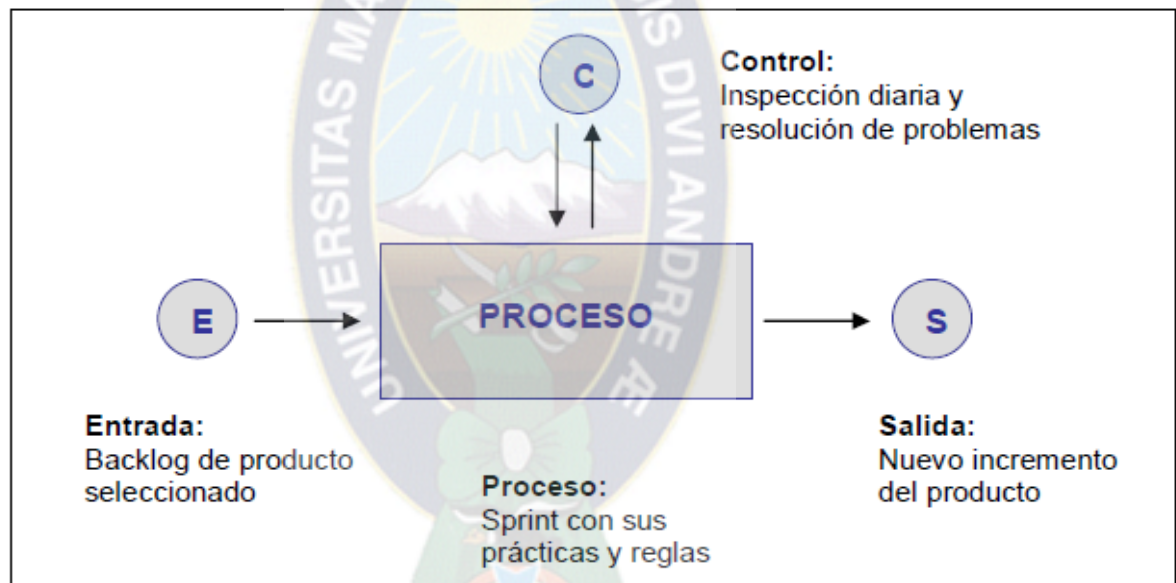


Figura 2.5: Forma de funcionamiento de un Sprint

Fuente: [e-metodoagilscrum-100513145255-phpapp02.pdf.]

Un Sprint es el periodo de tiempo durante el que se desarrolla un incremento de funcionalidad. Constituye el núcleo de Scrum, que divide de esta forma el desarrollo de un proyecto en un conjunto de pequeñas “carreras”.

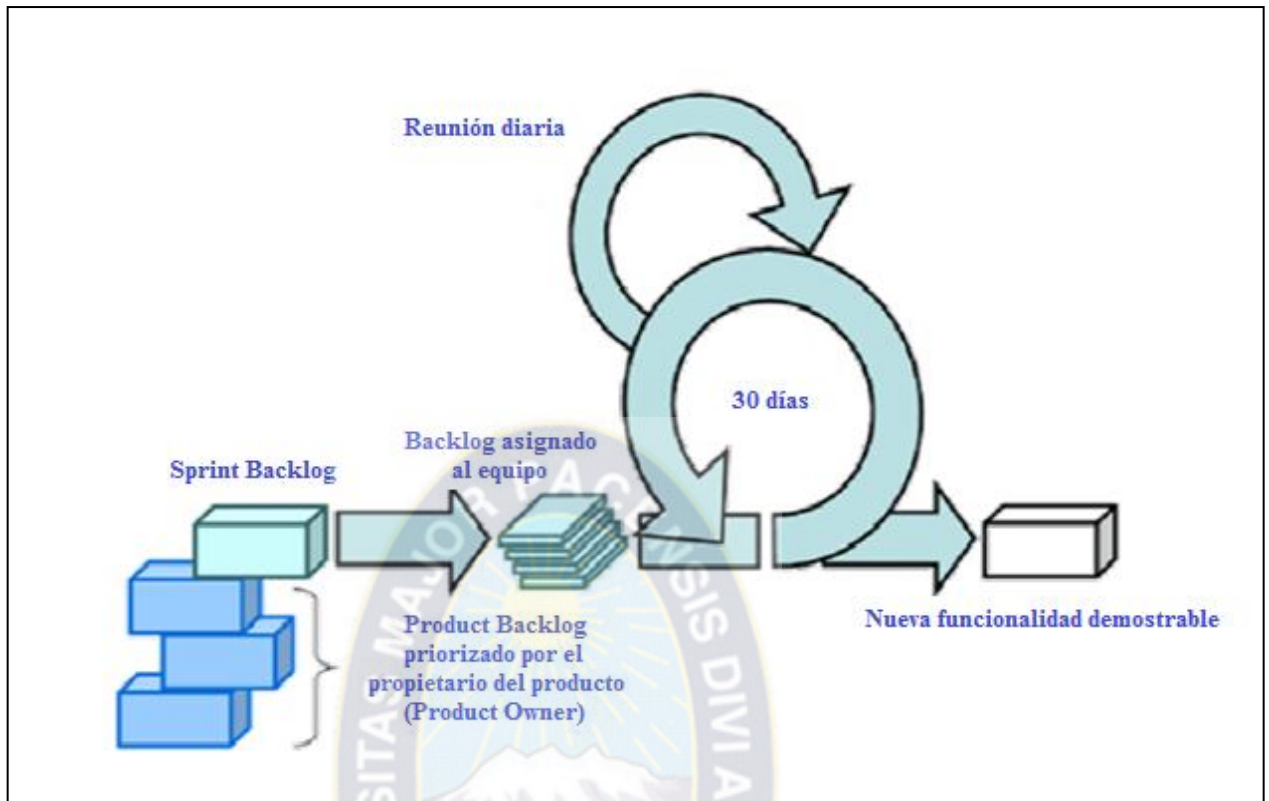


Figura 2.6: Desarrollo en pequeñas carreras en Scrum

Fuente: [e-metodoagilscrum-100513145255-phpapp02.pdf.]

- Duración máxima del Sprint: 30 días.
- Durante el Sprint no se puede modificar el trabajo que se ha acordado en el Backlog.
- Sólo es posible cambiar el curso de un Sprint, abortándolo, y sólo lo puede hacer el Scrum Master si decide que no es viable por alguna de las razones siguientes:
 - ✓ La tecnología acordada no funciona.
 - ✓ Las circunstancias del negocio han cambiado.
 - ✓ El equipo ha tenido interferencias.

2.5.2.5 Planificación

2.5.2.5.1 Pre-juego

Planificación: Definición de una nueva versión basada en la pila actual, junto con una estimación de coste y agenda. Si se trata de un nuevo sistema, esta fase abarca tanto la visión como el análisis. Si se trata de la mejora de un sistema existente comprende un

análisis de alcance más limitado. Arquitectura: Diseño de la implementación de las funcionalidades de la pila. Esta fase incluye la modificación de la arquitectura y diseño generales.

2.5.2.5.2 Juego

Desarrollo de sprints: Desarrollo de la funcionalidad de la nueva versión con respeto continuo a las variables de tiempo, requisitos, costo y competencia. La interacción con estas variables define el final de esta fase. El sistema va evolucionando a través de múltiples iteraciones de desarrollo o sprints.

2.5.2.5.3 Post-juego

Preparación para el lanzamiento de la versión, incluyendo la documentación final y pruebas antes del lanzamiento de la versión.

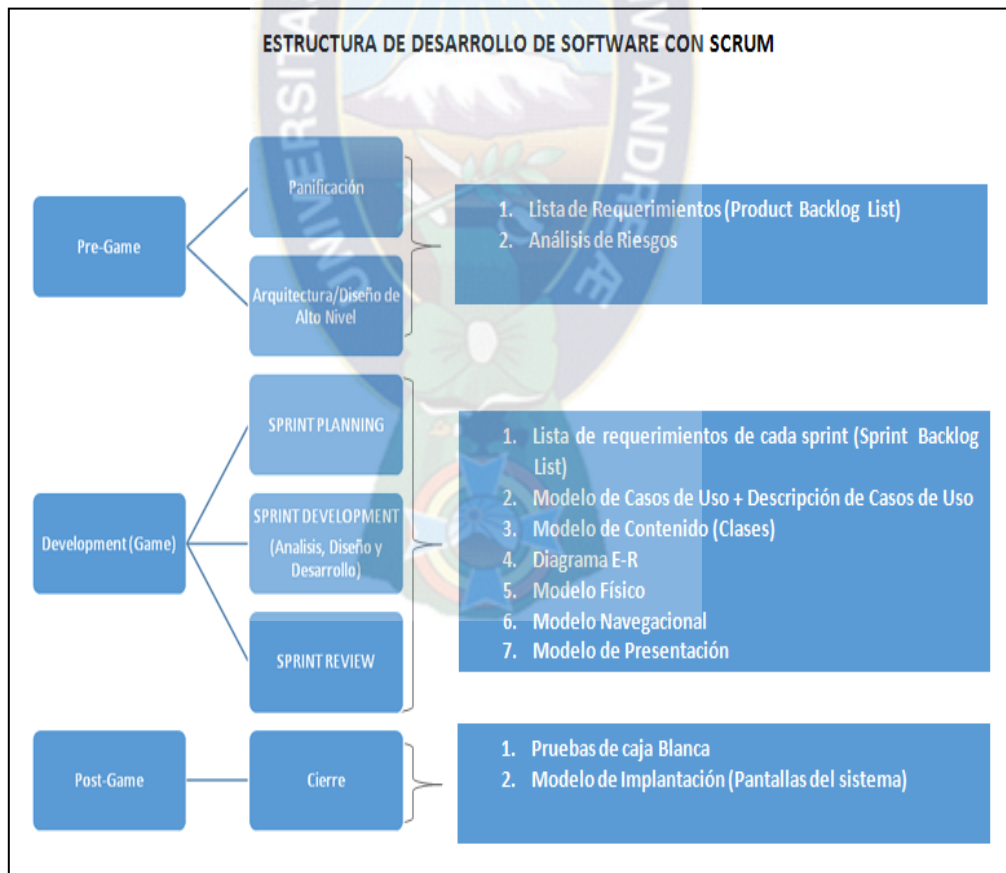


Figura 2.7: Artefactos del Scrum

Fuente: [e-metodoagilscrum-100513145255-phpapp02.pdf.]

Se planifica en detalle el trabajo al inicio de cada Sprint asumiendo que los objetivos no van a cambiar durante el mismo. De esta manera se atenúa el riesgo.

Aspectos a tener en cuenta sobre la planificación de un Sprint:

- ✓ Una determinación general de alcance, frecuentemente basada en una EDT (Estructura de División del Trabajo).
- ✓ Estimaciones de esfuerzo de alto nivel realizadas durante la etapa de concepción del proyecto.
- ✓ Esfuerzo dedicado a labores de soporte o de preparación de los ambientes requeridos por el proyecto.
- ✓ Esfuerzo asociados a las reuniones diarias, de planificación y de revisión.
- ✓ Requerimientos de recursos de infraestructura o logísticos (máquinas, redes, licencias, papel, pizarras, etc.).
- ✓ Habilidades presentes y necesarias en el equipo.
- ✓ Restricciones asociadas al conocimiento del negocio, la tecnología o externas (legales, reglamentarias, estándares, etc.).

Rol del Scrum Master durante la planificación:

- Dirige la planificación.
- Es vínculo entre el equipo y el Product Owner del proyecto.
- Registra problemas y riesgos detectados durante la planificación.
- Registra las tareas, asignaciones y estimaciones.
- Inicia el Backlog del Sprint.
- ✓ ***Sprint Backlog.***

Trabajo o tareas determinadas por el equipo para realizar en un Sprint.

- ❖ Tareas a convertir en producto funcional.
- ❖ Las tareas se estiman en una duración entre 1 a 20 horas de trabajo. Las de mayor duración deben intentar descomponerse en sub-tareas de ese rango de tiempo.
- ❖ La estimación se actualiza día a día.
- ✓ ***Scrum diario.***

Scrum asume que el proceso es complejo y que es necesario inspeccionarlo frecuentemente, por eso se realiza una reunión diaria de seguimiento. El encuentro diario

impide caer en el dilema señalado por Fred Brooks: “¿Cómo es que un proyecto puede atrasarse un año? Un día a la vez”.

El foco de la reunión es determinar el avance en las tareas y detectar problemas o “bloqueos” que estén haciendo lento el progreso del equipo o que eventualmente impidan a un equipo cumplir con la meta del Sprint. La idea es que ningún problema quede sin resolver o, por lo menos, sin iniciar alguna acción de respuesta dentro de las 24 horas después de su detección.

La reunión es además un espacio definido para que cada miembro del equipo comunique a los demás el estado de su trabajo y por lo tanto reafirme el compromiso.

Tips para un buen Scrum diario:

- ✓ Todos los días en el mismo sitio y a la misma hora. Con una duración máxima de 15 minutos.
- ✓ Se recomienda que sea la primera actividad del día.
- ✓ Deben acudir todos los miembros del equipo.
- ✓ Hacer un círculo, permitir que todos vean a todos.
- ✓ Mantener el foco.
- ✓ Solo habla una persona a la vez y los demás escuchan. No se permiten interrupciones, entrar y salir, hablar por teléfono, etc.
- ✓ Sólo se habla de: ¿En que trabajé ayer? ¿En qué voy a trabajar hoy? ¿Que problemas impiden mi progreso?
- ✓ No iniciar discusiones muy largas sobre temas técnicos o sobre los problemas, esto hay que registrarlo y manejarlo después de la reunión.
- ✓ Si están permitidas las preguntas para aclarar el alcance de un trabajo o de un problema.
- ✓ Cuando un miembro informa de algo de interés para otros, o necesita ayuda de otros, estos se reúnen al terminar la revisión diaria.
- ✓ Las gallinas no pueden intervenir ni distraer, y el Scrum Master puede limitar el número de gallinas asistentes si lo considera oportuno.

Rol del Scrum Master durante el Scrum:

- ✓ Dirige la reunión y mantiene el foco.
- ✓ Hace preguntas para aclarar dudas.

- ✓ Registra (escribe o documenta) los problemas para su resolución después de la reunión.
- ✓ Se asegura que los miembros cuenten con el ambiente adecuado para la reunión.

➤ **Estimaciones.**

Las estimaciones se realizan por primera vez en la reunión de planificación al inicio del Sprint. Luego las tareas se re-estiman todos los días y se registran sus cambios en el Backlog del Sprint; esta actividad puede ser realizada inmediatamente antes o después del Scrum diario.

Algunas claves para la estimación:

- ✓ Siempre se realizan estimaciones de esfuerzo, no de duración.
- ✓ Siempre se estima el esfuerzo total pendiente para terminar la tarea, no se estima el esfuerzo consumido.
- ✓ Se buscan unidades manejables, lo usual es que estén en un mínimo de 2 horas y un máximo de 20. Si la tarea es muy corta se trata de juntarla con otras relacionadas. Si la tarea es muy grande se trata de descomponerla.

➤ **Builds continuos y pruebas básicas.**

La estrategia que generalmente se utiliza es la de Builds continuos y “smoke test” (prueba básica para la funcionalidad del sistema).

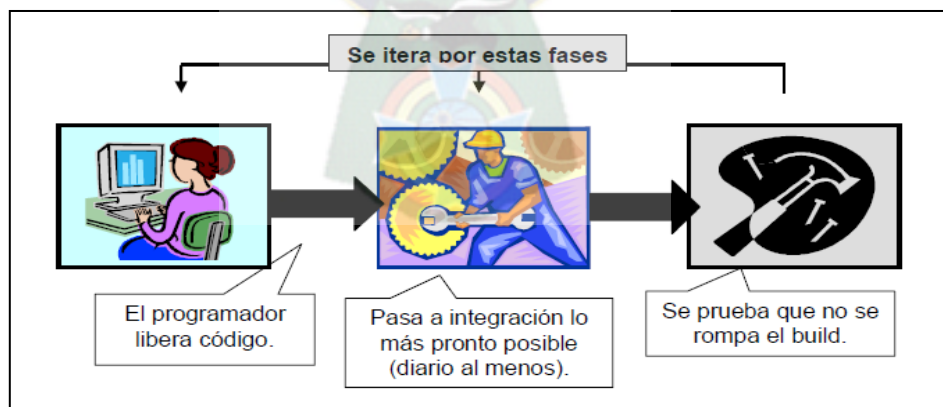


Figura 2.8: Iteración por fases

Fuente: [e-metodoagilscrum-100513145255-phpapp02.pdf.]

Procedimiento de Builds continuos.

- ✓ Los programadores desarrollan según el Backlog del Sprint, y al finalizar, notifican al integrador.
- ✓ El integrador toma el código y lo integra con el resto del producto.
- ✓ Se compila el software y se prueba “por arriba” el producto, para verificar que no se haya roto.
- ✓ Si se encuentran problemas se devuelve al desarrollador.
- ✓ Se notifica al equipo que hay una nueva versión “estable” del código para usar como base.

➤ ***Revisión del Sprint.***

El objetivo de la reunión de revisión es presentar el producto o porción del producto desarrollada por el equipo a los usuarios. La reunión se utiliza para detectar inconformidades mayores que se vuelven elementos del Backlog de Producto y que eventualmente se resuelven en el siguiente Sprint.

A la reunión asisten el equipo, el Scrum Master, el Product Owner y todas las personas implicadas en el proyecto (gallinas).

Tips para una buena revisión:

- ✓ Duración máxima de la reunión: 4 horas.
- ✓ Preparar la presentación, teniendo en cuenta que lo único que se presenta es el producto (no Power Point o similares).
- ✓ Mantener el foco durante la reunión de revisión, el foco es el producto solamente.
- ✓ Finalidad: presentar al propietario del producto y a las gallinas las nuevas funcionalidades implementadas.
- ✓ Las funcionalidades no implementadas no se presentan.
- ✓ Registrar todos los comentarios e inconformidades declaradas de los usuarios sobre el producto para determinar cuáles de ellas formarán parte del Backlog.

➤ **Reunión retrospectiva.**

Scrum involucra el concepto de mejora continua a través de las reuniones de retrospectión. Las reuniones buscan detectar los puntos positivos y negativos del Sprint para generar propuestas de mejora para futuros Sprints.

Las reuniones de retrospectión son el concentrador del aprendizaje organizacional sobre el Scrum. Los puntos positivos y negativos se registran y se definen ítems de acción para cada uno. Los ítems de acción definidos se toman en cuenta en los siguientes Sprints.

Acuden el equipo y el Scrum Master, y opcionalmente el Product Owner del Producto.

Tips para una buena retrospectión:

- ✓ Todos los miembros del equipo responden a dos preguntas:
- ✓ ¿Qué cosas fueron bien en el último Sprint?
- ✓ ¿Qué cosas se podrían mejorar?
- ✓ El Scrum Master anota todas las respuestas.
- ✓ El equipo prioriza las mejoras posibles.
- ✓ El Scrum Master no proporciona respuestas, sino que ayuda al equipo a encontrar la mejor forma de trabajar con Scrum.
- ✓ Las acciones de mejora localizadas que se puedan implementar en el próximo Sprint deben introducirse en el Backlog como elementos no funcionales.

2.6 Proceso de desarrollo de SCRUM

2.6.1 Introducción

En este apartado se presenta el proceso de desarrollo elaborado para complementar la metodología Scrum ya que esta no contiene definiciones en cuestiones técnicas.

Se utiliza un proceso ágil iterativo e incremental que respeta las cinco etapas tradicionales de un proyecto que facilitan su gestión y control; ellas son: planificación, análisis, diseño, construcción y prueba, e implementación. Como el objetivo final de la metodología es llegar al éxito del proyecto se definen, en forma clara, los entregables de cada etapa y el alcance global, reflejando estos puntos en la planificación de todas las tareas involucradas.

Se considera que: etapas delimitadas, entregables definidos y tareas acotadas son claves para el cumplimiento del plan.

2.6.2 Proceso iterativo e incremental

Este tipo de proceso de desarrollo permite hacer entregas parciales que se van complementando según avanza el proyecto. De esta manera se reducen los riesgos y a la vez el cliente va experimentando los resultados de su proyecto.

Dado que los proyectos de software son largos es común dividir el trabajo en mini proyectos. Cada mini proyecto es una iteración que resulta en un incremento. Para ser más efectivas las iteraciones deben ser controladas, es decir deben ser seleccionadas y llevadas a cabo de una forma planeada.

Se ha propuesto un *proceso iterativo* para garantizar la realimentación de información y de requisitos una vez iniciado el desarrollo, así como la validación continua del sistema. Esto permite que cada iteración contemple ciclos de desarrollos completos y cortos, y obtener así rápidamente, una nueva versión con mejoras sobre las versiones anteriores.

Se ha propuesto un *proceso incremental* en el sentido de añadir capacidades y funcionalidades al software de acuerdo con el crecimiento de las necesidades; con el propósito de obtener el sistema final tras la realización de diferentes ciclos. El final de cada ciclo proporciona además, una versión estable del software. Esto permite entregas al cliente de forma rápida y ágil.

Al entregar partes de la aplicación a tiempo y regularmente, el equipo de desarrollo también gana y establece credibilidad en su entorno y aumenta su auto-estima. A la vez que este tipo de estrategia se enfoca más en las necesidades de los usuarios, instándolos a identificar sus prioridades en cada etapa del proyecto.

A continuación detallamos las etapas por las cuales transita nuestro proceso de desarrollo y la combinación de herramientas utilizadas en la metodología.

2.6.3 Etapas del proceso de desarrollo

2.6.3.1 Planificación

Objetivo: Es la etapa más importante de todas, ya que se define el proyecto propiamente dicho.

Tareas: Relevamiento preliminar de los procesos del negocio, definición y secuenciamiento de actividades, definición del alcance, estimación de tiempos, definición de recursos, análisis de riesgos, estimación de costos.

Entregables: Documento de definición del proyecto o del Sprint.

En esta etapa es importante aclarar que, al comienzo, la planificación se realiza en forma general para determinar el alcance, la duración y el precio del proyecto, una vez que el cliente decide llevarlo a cabo, las siguientes planificaciones son a nivel de iteración, se planifica el Sprint.

2.6.3.2 Análisis

Objetivo: Obtener todas las definiciones y especificaciones funcionales para poder llevar adelante las fases de Diseño y Construcción. Es una etapa clave ya que el alcance y las características de la solución quedan acordados, lo cual permite mitigar los principales riesgos de un proyecto.

Tareas: Afianzamiento de las definiciones funcionales, definición de los requisitos a través de casos de uso, planificación de las etapas posteriores y ajuste de los tiempos preestablecidos.

Entregable: Documento de alcance, casos de uso y sus respectivas descripciones.

2.6.3.3 Diseño

Objetivo: Generar el modelo de datos para que la solución cumpla con los requerimientos definidos. El diseño generado deberá contemplar las posibles modificaciones futuras, crecimiento de la solución, mayor carga e incorporación de nuevas funcionalidades.

Tareas: Diagrama Entidad Relación (DER), diseño de las interfaces de usuario, diseño de las integraciones a realizar. Durante esta etapa también se realizan pruebas para puntos críticos del proyecto.

Entregables: Entre los entregables típicos de esta etapa se encuentran: DER, esqueleto del software armado, guía de diseño, diseño de la infraestructura, y la planificación ajustada con la evolución y avances obtenidos.

2.6.3.4 Construcción y prueba

Objetivo: Construir la solución del Release (Sprint), cumpliendo con las definiciones y especificaciones de los documentos de alcance.

Generalmente es la etapa de mayor duración y con mayor dinámica de trabajo.

Tareas: Programación y desarrollo de todos los componentes y funcionalidades. Implementación de las estructuras de datos, y sus procedimientos, elaboración de documentación técnica y ajustes funcionales, implementación de las integraciones y todas las actividades necesarias para poner en marcha la solución. En esta etapa se realizarán las pruebas de usabilidad, funcionalidad y carga de datos.

Entregables: El entregable principal es el incremento de software funcionando.

2.6.3.5 Implementación

Objetivo: Disponer del sistema productivo con sus ambientes de producción, metodología de trabajo y manuales operativos. Se incluye, de ser necesario, el personal operativo capacitado. Obtención de nuevas funciones a agregar o posibles errores a reparar.

Tareas: Puesta en marcha de la aplicación en el ambiente de producción, elaboración de manuales operativos, y todas las actividades relacionadas al éxito del lanzamiento como la integración del ambiente de producción con terceras partes, etcétera.

Entregables: El sistema productivo con sus manuales operativos, de mantenimiento y de procedimientos. Esquemas de auditoría y seguridad.

Integraciones con terceras partes operativas. Sistema totalmente probado.

2.6.4 EDT del proceso de desarrollo

Presentamos nuestro proceso de desarrollo a través de una Estructura de División del Trabajo para verlo gráficamente.

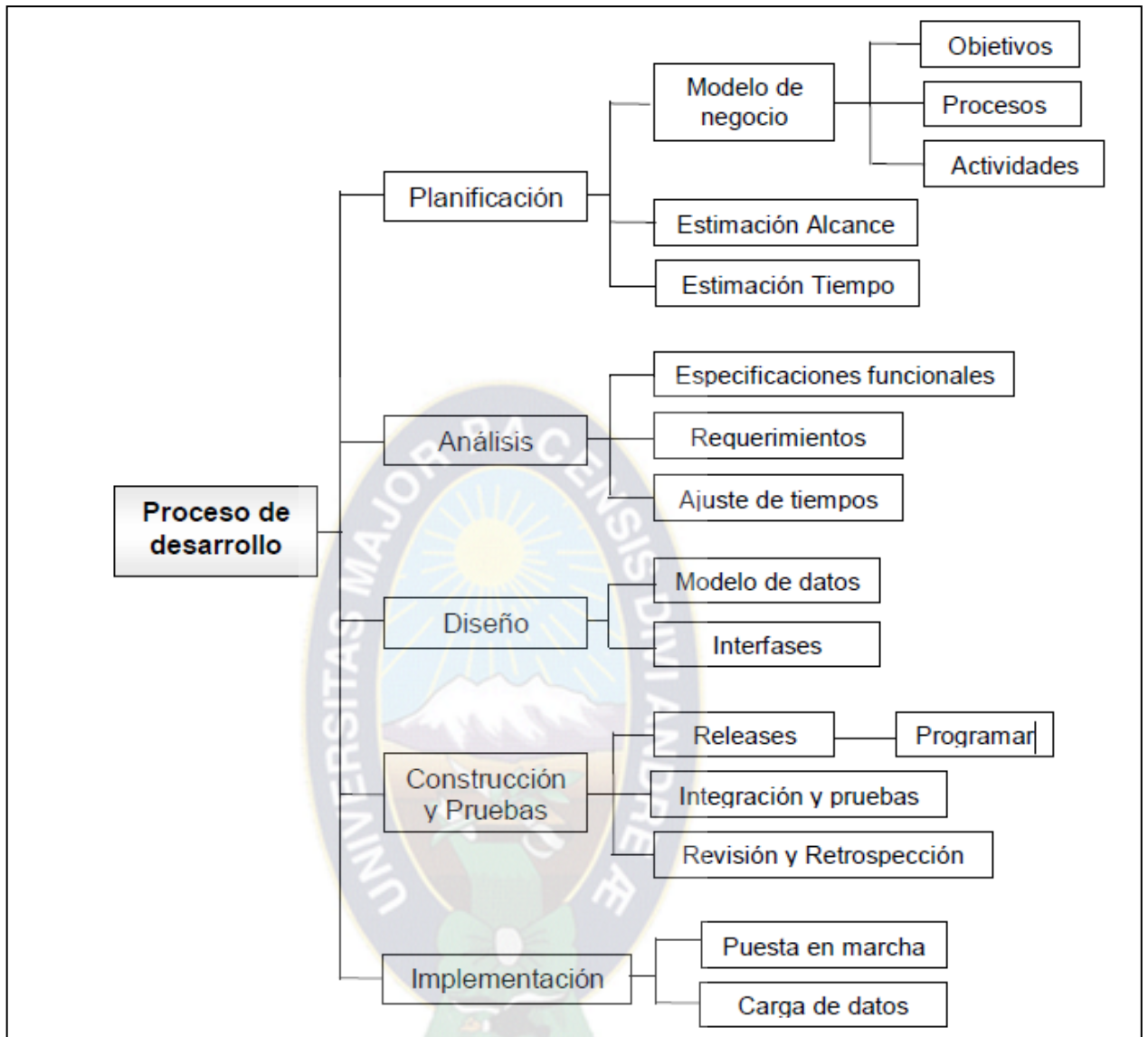


Figura 2.9: Estructura de División de Trabajo

Fuente: [e-metodoagilscrum-100513145255-phpapp02.pdf.]

2.6.5 Herramientas

2.6.5.1 Técnicas de relevamiento

Entrevistas con el cliente y los usuarios; revisión de registros, y observación.

2.6.5.2 Casos de uso

Son un método práctico para explorar requerimientos. Ayudan a describir qué es lo que el sistema debe hacer desde el punto de vista del usuario.

Por lo tanto, consideramos que los casos de uso proporcionan un modo claro y preciso de comunicación con el cliente.

Descripciones de casos de uso: detallan los casos de uso, en ellas se explica la forma de interactuar entre el sistema y el usuario.

Tanto los casos de uso como las descripciones de los mismos se utilizan en la etapa del análisis para definir los requisitos.

2.6.5.3 Diagrama de actividades

Sirven fundamentalmente para modelar el flujo de control entre actividades. La idea es generar una especie de diagrama en el que se puede ver el flujo de actividades que tienen lugar a lo largo del tiempo, así como las tareas concurrentes que pueden realizarse a la vez. El diagrama de actividades sirve para representar el sistema desde otra perspectiva.

Desde un punto de vista conceptual, la actividad es alguna tarea que debe ser realizada. Por este motivo, en un diagrama de actividades aparecerán acciones y actividades correspondientes a distintas clases; colaborando todas ellas para conseguir un mismo fin.

Los diagramas de actividad son similares a los diagramas de flujo procesales, con la diferencia de que todas las actividades están claramente asociadas a un caso de uso. También se utilizan en la etapa del análisis.

2.6.5.4 Diagrama de Entidad Relación (DER)

- Un modelo de datos describe de una forma abstracta cómo se representan los datos, sea en una empresa, en un sistema de información o en un sistema de base de datos.

Los DER son una herramienta para el modelado de datos de un sistema de información. Estos diagramas expresan entidades relevantes y sus inter-relaciones. Formalmente, son un lenguaje gráfico para describir conceptos. Informalmente, son simples dibujos o gráficos que (si se saben interpretar) describen la información que trata un sistema de información y el software que lo automatiza.

2.7 Metodología UWE

UWE es un proceso del desarrollo para aplicaciones Web enfocado sobre el diseño sistemático, la personalización y la generación semiautomática de escenarios que guíen el proceso de desarrollo de una aplicación Web. UWE describe una metodología de diseño sistemático, basada en la técnica de UML.

Es una herramienta que nos permitirá modelar aplicaciones web, utilizada en la ingeniería Web, prestando especial atención en sistematización y personalización (sistemas adaptivos). UWE es una propuesta basada en el proceso unificado UML pero adoptamos a

la Web. En requisitos separa las fases de captura, definición y validación. Hace además una clasificación y un tratamiento especial dependiendo del carácter de cada requisito.

En el marco de UWE es necesario la definición de un perfil UML, (extensión) basados en estéreo tipos con este perfil se logra la asociación de una semántica distinta a los diagramas de UML puro, con el propósito de acoplar el UML a un dominio específico, en este caso, las aplicaciones Web. Entre los principales modelos de UWE podemos citar: el modelo lógico-conceptual, modelo navegacional, modelo de presentación, visualización de escenarios Web y la interacción temporal, entre los diagramas: diagramas de estado, secuencia, colaboración, actividad.

UWE define vistas especiales representadas gráficamente por diagramas UML. Además UWE no limita el número de vistas posibles de una aplicación UML proporciona mecanismos de extensión basados en estereotipos. Estos mecanismos de extensión son los que UWE utiliza para definir estereotipos que son los que finalmente se utilizarán en las vistas especiales para el modelado de aplicaciones Web. De esta manera, se obtiene una notación UML adecuada en un dominio específico a la cual se le conoce como perfil UML.

2.7.1 Fases de la metodología UWE

UWE cubre todo el ciclo de vida de este tipo de aplicaciones centrando además su atención en aplicaciones personalizadas o adaptativas.

Las fases o etapas a utilizar son:

1) Captura, análisis y especificación de requisitos: En simples palabras y básicamente, durante esta fase, se adquieren, reúnen y especifican las características funcionales y no funcionales que deberá cumplir la aplicación web.

Trata de diferente forma las necesidades de la información, las necesidades de adaptación y las de interfaz de usuario, así como algunos requisitos adicionales. Centra el trabajo en el estudio de los casos de uso, la generación de los glosarios y el prototipado de la interfaz de usuario.

2) Diseño del sistema: Se basa en la especificación de requerimientos producido por el análisis de los requerimientos (fase de análisis), el diseño define como estos requisitos se cumplirán, la estructura que debe darse a la aplicación web.

- 3) **Codificación del software:** Durante esta etapa se realizan las tareas que comúnmente se conocen como programación; que consiste, esencialmente, en llevar a código fuente, en el lenguaje de programación elegido, todo lo diseñado en la fase anterior.
- 4) **Pruebas:** Las pruebas se utilizan para asegurar el correcto funcionamiento de secciones de código.
- 5) **La instalación o fase de Implementación:** Es el proceso por el cual los programas desarrollados son transferidos apropiadamente al computador destino, inicializados y eventualmente configurados; todo ello con el propósito de ser ya utilizados por el usuario final. Esto incluye la implementación de la arquitectura, de la estructura del hiperespacio, del modelo de usuario, de la interfaz de usuario, de los mecanismos adaptativos y las tareas referentes a la integración de todas estas implementaciones.
- 6) **El Mantenimiento:** Es el proceso de control, mejora y optimización del software ya desarrollado e instalado, que también incluye depuración de errores y defectos que puedan haberse filtrado en la fase de pruebas de control.

2.8 Ingeniería de WEB

El grupo de investigación de Ingeniería de software e Ingeniería Web es un grupo de científicos multidisciplinares que trabaja en diversas líneas dentro de MONDRAGON UNIBERTSITATEA. Se trata de una línea eminentemente transversal que contribuye a la mayoría de las temáticas y acciones de investigación vislumbradas por las estrategias Europeas y nacionales en los dominios de la industria, administración, pública, construcción, salud, turismo, transporte y energía. El objetivo de la investigación es investigar para dar soluciones integrales relacionadas con las metodologías y el desarrollo de software en estos dominios.

Áreas de investigación

Las principales áreas de investigación del grupo son:

Ingeniería del software: El objetivo es industrializar el proceso de desarrollo de software para reducir costos y mejorar la calidad de los productos software. Para ello se centra las líneas de productos software y el desarrollo de software dirigido por modelos

como paradigmas de desarrollo que proporcionan numerosos beneficios respecto al desarrollo de software tradicional. Ambos enfoques utilizados conjuntamente permiten:

- Reducir los costes del ciclo de vida del software.
- Rápida adaptación del sistema a nuevas tecnologías.
- Reducción del tiempo de desarrollo de nuevos sistemas.
- Mejora de la calidad del software permitiendo una fácil evolución del mismo.

Este área investiga las tecnologías Web semántica y las estructuras de datos Linked data desde la prisma de los sistemas abiertos (open source y open data). Estas tecnologías posibilitan estructuras de datos enriquecidas que se utilizan para:

- Facilitar la búsqueda de contenidos y servicios.
- Crear agentes inteligentes para la automatización de procesos.
- Extraer, procesar y almacenar conocimiento (Knowledge management).
- Integrar sistemas heterogéneos.
- Componer sistemas complejos.
- Filtrar información.
- Posibilitar el proceso semántico por parte de las máquinas.
- Estandarizar y flexibilizar vocabularios.

En esta área se investiga en el diseño, desarrollo de plataformas interoperables, basadas en servicios web y el cumplimiento de estándares, y la construcción de arquitecturas SOA, que posibilitan la integración, multiplataforma y multidispositivo.

Integración de herramientas y simulación distribuida: El objetivo es integrar herramientas para facilitar el desarrollo de sistemas ciberfísicos y su validación temprana mediante la simulación distribuida. Para ello, utilizamos los siguientes estándares:

- Open Software Lifecycle Collaboration (OSLC): para la integración de herramientas del ciclo de desarrollo de sistemas.
- Functional Mockups Interfaces (FMI): para la cosimulación de diferentes aspectos de un sistema en la fase de validación temprana.

2.8.1 ¿Qué es un sistema Web?

Los “Sistemas Web” o también conocidos como “Aplicaciones Web” son aquellos están creados o instalados no sobre una plataforma o sistemas operativos (Windows, Linux). Sino que se alojan en un servidor en Internet o sobre una intranet (red local). Su

aspecto es muy similar a páginas Web que vemos normalmente, pero en realidad los “Sistemas Web” tienen funcionalidades muy potentes que brindan respuestas a casos particulares.

Los sistemas Web se pueden utilizar en cualquier navegador Web (chrome, firefox, Internet Explorer, etc.) sin importar el sistema operativo. Para realizar las aplicaciones Web no es necesario instalarlas en cada computadora ya que los usuarios se conectan a un servidor donde se aloja el sistema.

Las aplicaciones Web trabajan con bases de datos que permiten procesar y mostrar información de forma dinámica para el usuario. Los sistemas desarrollados en plataformas Web, tienen marcadas diferencias con otros tipos de sistemas, lo que lo hacen muy beneficiosos tanto para las empresas que lo utilizan, como para los usuarios que lo operan en el sistema.

Este tipo de diferencias se ven reflejada en los costos, en la rapidez de obtención de la información, en la optimización de las tareas por parte de los usuarios y en alcanzar una gestión estable.

2.9 Definición del esquema de base de datos

Consiste en determinar el esquema de base de datos que se utilizará, en este caso se vio conveniente el estudio de bases de datos objetos relacionales.

En la generalidad de las aplicaciones es necesario guardar y recuperar la información en un mecanismo de almacenamiento persistente, una base de datos relacional por ejemplo. Dado el predominio de estas últimas, a menudo se requiere su uso en vez de otras bases más manejables orientadas a objetos. De ser así, surgen varios problemas a causa de la desigualdad entre representaciones de datos orientadas a registros, y las que se orientan a objetos, se requieren servicios especiales de ambos tipos en las bases de datos relacionales.

¿Como mapear un objeto a un archivo o a un esquema de base de datos relacional? El patrón de representación de objetos a tablas propone definir una tabla para cada clase de objeto persistente. Los atributos de objetos que contienen tipos primitivos de datos (numero, booleano, cadena y otros) se mapean en las columnas.

Conviene contar con un medio que relacione los objetos con los registros y de asegurarse de que la repetición de la materialización de un registro no culmine en la

duplicación de objetos. El patrón identificador de objetos (IDO) se propone asignar un IDO a cada registro y objeto (o al agente de un objeto). Un identificador de objetos suele ser un valor alfanumérico, es único para un objeto en específico, toda tabla de base de datos relacional tiene un IDO como clave primaria, y los objetos también contarán (directa o indirectamente) con un identificador (Larma, 1999).

Si todos los objetos se asocian a un IDO y si todas las tablas poseen una clave primaria IDO los objetos podrán mapearse de modo singular en una región de alguna tabla.

¿Como representar las relaciones de objetos en una tabla de una base de datos relacional? La respuesta se da en el patrón de representación de las relaciones de objetos como tablas que propone lo siguiente:

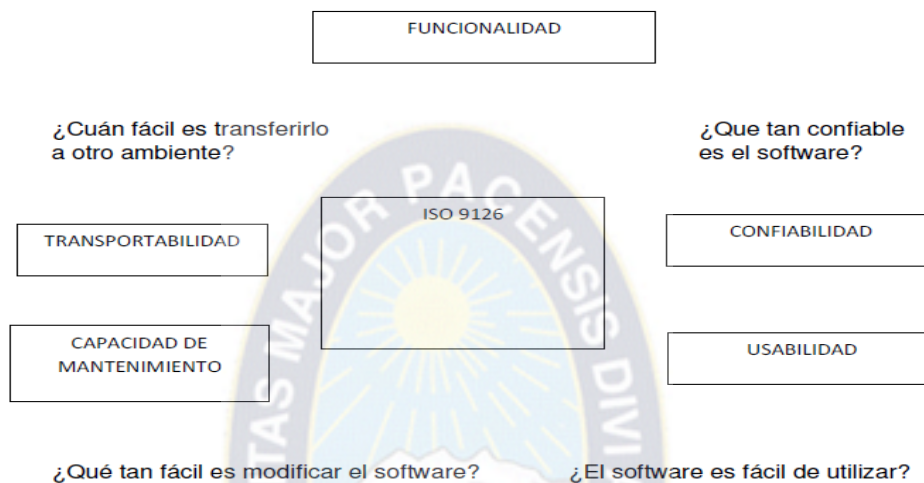
- Asociación uno a uno
 - ✓ Colocar una clave foránea de identificador de objetos en una o en las dos tablas que representan los objetos en la relación.
 - ✓ Crear una tabla asociativa que registre los identificadores de objetos de cada relación.
- Asociaciones de uno a muchos, una colección por ejemplo
 - ✓ Crear una tabla asociativa que registre los identificadores de cada objeto en la relación.
- Asociaciones de muchos a muchos
 - ✓ Crear una tabla asociativa que registre todos los identificadores de objetos en la relación (e-metodoagilscrum-100513145255-phpapp02.pdf, 2005).

2.10 Calidad del software ISO 9126

La organización internacional para la estandarización ISO fue creada en 1946 con el fin de facilitar el comercio internacional, la coordinación internacional y la unificación de estándares industriales promoviendo una serie de simples patrones de estándares que deberán ser reconocidos y respetados ISO 9126 fue originalmente desarrollado en 1991 para proporcionar un esquema para la evaluación de la calidad del software y así refinarlo en un periodo de 10 años. Muchos estudios criticaron la ISO 9126 por no recomendar requerimientos específicos de calidad en vez de definir un esquema general para la evaluación de calidad del software (Valenti, 2002). Nosotros creemos que este es de hecho

una de las fortalezas y así es más adaptable y puede ser usado a través de varios sistemas incluso sistemas de aprendizaje virtual. El producto original definió seis características del producto estas seis características son divididas en un número de sub características.

¿Están disponibles las funciones requeridas en el software?



Funcionalidad	<p>Conveniencia</p> <p>Precisión</p> <p>Interoperabilidad</p> <p>seguridad</p>	<ul style="list-style-type: none"> ➤ Puede el software desempeñar las tareas requeridas. ➤ ¿El resultado es el esperado? ➤ ¿el sistema puede interactuar con otro? ➤ ¿el sistema impide el acceso no autorizado?
Confiabilidad	<p>Vencimiento</p> <p>Tolerancia a las fallas</p> <p>Capacidad de recuperación.</p>	<ul style="list-style-type: none"> ➤ ¿muchas de las fallas han sido eliminadas durante el tiempo? ➤ ¿el software es capaz de manejar errores? ➤ ¿Puede el software reasumir el funcionamiento y restaurar datos perdidos después de la falla?
Utilidad	Claridad	<ul style="list-style-type: none"> ➤ ¿el usuario comprende fácilmente como usar el sistema?

	<p>Capacidad de aprendizaje</p> <p>Operatividad</p> <p>atractivo</p>	<ul style="list-style-type: none"> ➤ ¿Puede el usuario aprender fácilmente a utilizar el sistema? ➤ ¿el usuario puede utilizar el sistema sin mucho esfuerzo? ➤ ¿la interfaz se ve bien?
Eficiencia	<p>Comportamiento del tiempo.</p> <p>Utilización de recursos</p>	<ul style="list-style-type: none"> ➤ ¿Qué tan rápido responde el sistema? ➤ ¿el sistema utiliza los recursos de manera eficiente?
Capacidad de Mantenimiento	<p>Capacidad de análisis</p> <p>Variabilidad</p> <p>Estabilidad</p> <p>Capacidad de prueba</p>	<ul style="list-style-type: none"> ➤ ¿las fallas Pueden ser fácilmente diagnosticadas? ➤ ¿el sistema puede ser fácilmente modificado? ➤ ¿el sistema puede seguir funcionando si se hacen cambios? ➤ ¿el sistema puede ser probado fácilmente?
Transportabilidad	<p>Adaptabilidad</p> <p>Capacidad de instalación</p> <p>Conformidad</p> <p>Capacidad para reemplazar</p>	<ul style="list-style-type: none"> ➤ ¿El software se puede trasladar a otros ambientes? ➤ ¿el software se puede instalar fácilmente? ➤ ¿el software cumple con los estándares de Transportabilidad? ➤ ¿el software puede reemplazar fácilmente otro software?
Todas las características	Cumplimiento	<ul style="list-style-type: none"> ➤ ¿el software cumple con todas las leyes y reglamentos?

Tabla 2.2: Calidad ISO 9126

Fuente: [Valenti, 2002]

Estas características y sub -características representan un modelo detallado para la evaluación de cualquier sistema de software. En efecto Abran, Khelifi, Suryan & Seffah (2003) reclamaron que aunque no es exhaustivo, estas series constituyen el más extenso modelo de calidad de software desarrollado a la fecha esto es solo un modelo fácil para los no especialistas en contratar, por ejemplo más simple que el modelo IEEE P1484.1 LTSA, SCORM o IMS. A diferencia de otros esquemas, ISO 9126 cubre un amplio espectro de rasgos del sistema

2.11 Estudio de costos del sistema Web

2.11.1 Modelo Cocomo II

COCOMO II es un modelo que permite estimar el coste, el esfuerzo y el tiempo cuando se planifica una nueva actividad de desarrollo software, y está asociado a los ciclos de vida modernos. Fue desarrollado a partir de COCOMO, incluyendo actualizaciones y nuevas extensiones más adecuadas a los requerimientos de los ingenieros software.

Está construido para satisfacer aquellas necesidades expresadas por los estimadores software, como por ejemplo el apoyo a la planificación de proyectos, la previsión de personal de proyecto, re planificación, seguimiento, negociaciones de contrato o la evaluación del diseño.

COCOMO II proporciona una familia de modelos de estimación muy detallados, y que tiene muy en cuenta el tipo información disponible. Este conjunto de modelos se divide en tres:

- Modelo de composición de aplicaciones, para proyectos de construcción de interfaces gráficas de usuario.
- Modelo de diseño preliminar, utilizado para obtener estimaciones sobre el coste de un proyecto, antes de que esté determinada por completo su arquitectura.
- Modelo post-arquitectura, que es el más detallado, y debe ser usado una vez determinada la arquitectura al completo.

El usar un modelo u otro depende del nivel de detalle del proyecto, de la fidelidad requerida de las estimaciones, y de la definición de los requerimientos y de los detalles de la arquitectura. De una manera general, podemos determinar en qué momentos es más adecuado la utilización de un modelo u otro:

- El modelo de composición de aplicaciones se aplicará sobre todo en las primeras fases o ciclos en espiral.
- El modelo de diseño preliminar es útil para las siguientes fases o ciclos espirales, en los que se incluye la exploración de arquitecturas alternativas o estratégicas de desarrollo incremental.
- Una vez que el proyecto está listo para desarrollar y sostener un sistema especializado, el submodelo de Post-arquitectura proporciona información más precisa de los controladores de coste de entradas y permite cálculos de coste más exactos.

Los modelos usan la siguiente ecuación:

$$PM = 2.45 * EAF * \text{Tamaño B}$$

Donde EAF (Effort Adjustment Factor) es el producto de siete multiplicadores de esfuerzo para el modelo de desarrollo temprano, y diecisiete para el modelo post-arquitectural.

B es un factor que toma valores comprendidos entre 1.01 y 1.26, que viene dado por:

$B = 1.01 + \sum w_i$, siendo $\sum w_i$ la suma de cinco factores que causan efecto en la economía del proyecto, y que son:

- Existencia de precedentes al proyecto.
- Flexibilidad del desarrollo.
- Resolución de riesgos.
- Cohesión del equipo.
- Madurez del proceso.

Los multiplicadores de esfuerzo para COCOMO II son:

MODELO DE DESARROLLO TEMPRANO	MODELO POST-ARQUITECTURAL
Fiabilidad y complejidad del producto (RCPX)	Fiabilidad requerida del software (RELY)

	Tamaño de la base de datos (DATA) Complejidad del producto (CPLX) Necesidad de documentación (DOCU)
Necesidad de reutilización (RUSE)	Necesidad de reutilización (RUSE)
Dificultad de la plataforma (PDIF)	Restricción de tiempo de ejecución (TIME) Restricción de almacenamiento principal (STOR) Estabilidad de la plataforma (PVOL)
Capacidad del personal (PERS)	Capacidad del analista (ACAP) Capacidad del programador (PCAP) Continuidad del personal (PCON)
Experiencia del personal (PREX)	Experiencia en aplicaciones (AEXP) Experiencia con la plataforma (PEXP) Experiencia con el lenguaje y herramientas (LTEX)
Facilidades (FCIL)	Uso de herramientas software (TOOL) Desarrollo distribuido (SITE)
Programación (SCED)	Programación requerida del desarrollo (SCED)

Tabla 2.3: Multiplicadores de esfuerzo para COCOMO II

Fuente: [p6-estimacionCostesSoftware.pdf.]

2.12 Seguridad del Sistema

Intranet es una infraestructura basada en los estándares y tecnologías de Internet que soporta el compartir información dentro de un grupo bien definido y limitado (Geoci, 2000).

Aunque una Intranet sea una red privada en la que se tengan grupos bien definidos y limitados esta no se encuentra exenta de ataques que pudiesen poner en riesgo la información que maneja, ya que la mayoría de estos son provocados por sus mismos usuarios.

La mayoría de las estadísticas de seguridad en computo indican que cerca del 80% de los fraudes relacionados con las computadoras provienen de los usuarios internos, por esto las intranets son las más vulnerables a ataques de esta índole.

Por tal es importante establecer normas y políticas de seguridad a fin de tener un sistema confiable, y seguro. Un punto muy importante dentro de las políticas es el que tienen que ir acompañadas de sanciones, las cuales deberán también ser redactadas, revisadas, autorizadas, aplicadas y actualizadas (Geoci,2000) .

- **Políticas de contraseñas:** Son una de las políticas más importantes, ya que por lo general, las contraseñas constituyen la primera y tal vez única manera de autenticación y, por tanto, la única línea de defensa contra ataques. Estas establecen quien asignara la contraseña, que longitud debe tener, a que formato deberá apegarse, como será comunicada.

En PHP se utiliza la función MD5 (Message Digest 5), que es una función hash irreversible (de un solo sentido), es decir, inscripta el password tecleado por el usuario y es imposible que partiendo desde la cadena encriptado se vuelva a la contraseña origen. Por esto mismo no hay

problema de que alguien pueda acceder al campo encriptado de la base de datos.

- **Políticas de uso adecuado:** Especifican lo que se considera un uso adecuado o inadecuado del sistema por parte de los usuarios, así como lo que está permitido y lo que está prohibido dentro del sistema de computo.
- **Políticas de respaldos:** Especifican que información debe respaldarse, con que periodicidad, que medios de respaldo utilizar, como deberá ser restaurada la información, donde deberán almacenarse los respaldos (seginformatic.pdf, 2002).

Capítulo III

Marco Aplicativo

3.3 Introducción

El presente capítulo tiene como finalidad describir el análisis y diseño correspondiente al sistema, para dicho desarrollo se aplica la metodología SCRUM cuyas fases y otras características de la metodología fueron descritas en el capítulo anterior correspondiente al marco teórico.

Para la aplicación de la ingeniería de requerimientos se aplica las actividades correspondientes a la conceptualización, análisis, diseño e implementación del sistema de acuerdo al proceso de desarrollo. Al comenzar con cada iteración (Sprint) se determinan que partes se van a construir, tomando como criterios la prioridad para el negocio, la cantidad de trabajo que podrá abordar durante la iteración.

La metodología ágil SCRUM que utiliza un modelo de proceso incremental, y es complementada con la metodología UWE para las etapas de iteración, la metodología SCRUM fue también requerimiento de la empresa Técnica de Aluminio, Vidrio y Servicios (TALVISER), en la siguiente figura se puede apreciar gráficamente el modelo de los procesos que utilizo en el presente Proyecto de Grado.

En este capítulo se efectuara el análisis y diseño correspondientes al sistema, dando curso que SCRUM es una metodología de desarrollo muy simple, que requiere trabajo duro, porque a gestión no se basa en el seguimiento de un plan, sino en la adaptación continua a las circunstancias de la evolución del proyecto.

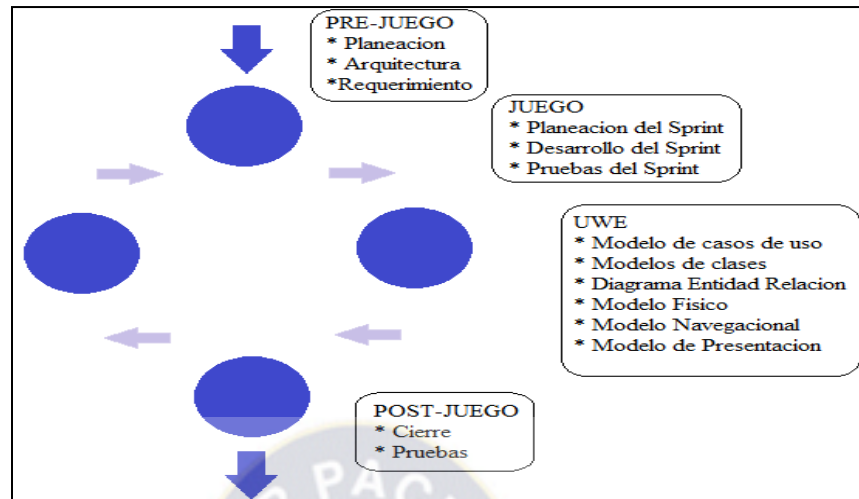


Figura 3.1: Etapas del Sprint

Fuente: [Elaboración Propia]

3.4 Pre- Juego

3.4.1 Recopilación de requerimientos

Ya que el presente proyecto constituye una solución e-learning, muchos de los requerimientos fueron tomados en base los elementos del e-learning que son la plataforma, los contenidos y los sistemas de comunicación.

A continuación se presenta el Backlog del producto, que contiene los requerimientos y características finales del sistema:

Las reuniones.- Esta reunión genera el “Sprint Backlog” o lista de tareas que se van a realizar, y en ella se determina los requerimientos del sistema.

Los requisitos del sistema, se parte de la visión del resultado que se desea obtener, y evolucionar durante el desarrollo. Es el inventario de características que el propietario del producto desea obtener, ordenado por orden de prioridad.

Los elementos.- Product Backlog: requisitos del sistema es parte de la visión del resultado que se desea obtener y que evoluciona durante el desarrollo, y es el inventario de la característica que el propietario del producto desea obtener, ordenado por orden de prioridad.

3.4.2 Especificación de requerimientos

Los requerimientos se obtuvieron a través de las entrevistas con los usuarios interesados de la empresa “TALVISER”, como resultado, se presenta a continuación los requerimientos funcionales que debe cumplir el sistema a desarrollar. Así mismo el control interno es parte de los procesos básicos de planificación, ejecución, supervisión y está incorporado en ellas.

ID	DESCRIPCION	MODULO	PRIORIDAD	ESTADO
R1	Base de Datos único para la empresa “TALVISER”	Plataforma	Alta	Terminado
R2	Diseño de registros o formularios o impresos a aplicar para una información fiable.	Plataforma	Media	Terminado
R3	Búsqueda de registros mediante filtros.	Plataforma	Media	Terminado
R4	Control y acceso seguro y diferenciado a usuarios.	Plataforma	Alta	Terminado
R5	Interfaz Amigable.	Comunicación, contenidos	Media	Terminado
R6	Desarrollo de una interfaz web para el uso del dispositivo móvil donde se enviara información a los almacenes.	Contenidos	Media	Terminado
R7	Desarrollo de una interfaz web para el uso del dispositivo móvil donde se obtendrá los reportes de almacenes.	Contenidos	Media	Terminado
R8	Control de Stocks de almacenes	Plataforma	Alta	Terminado

R9	Registro adecuado de la venta de los productos realizados.	Plataforma	Alta	Terminado
R10	Registro adecuado de la compra de materiales y herramientas a utilizar.	Plataforma	Alta	Terminado
R11	Registro adecuado de Datos.	Plataforma	Alta	Terminado

Tabla 3.1 Requerimiento

Fuente: [Elaboración propia]

3.4.3 Definición del cronograma de trabajo

El cronograma de trabajo está definida en base al ciclo de vida de la metodología SCRUM en el cual se identifican 3 etapas principales que son: pre-juego, juego, post juego, la descripción detallada del cronograma de trabajo del proyecto se la puede observar en el diagrama de Gantt que se encuentra en el Anexo.

3.2.4 Análisis de Riesgos

Un riesgo es la probabilidad de que ocurra algo adverso, existen 3 tipos de riesgos:

- **Riesgo del Proyecto.-** Es el que afecta a la calendarización o recursos del proyecto.
- **Riesgo del Producto.-** Afectan a la calidad o rendimiento del software que se está desarrollando.
- **Riesgo del Negocio.-** Afecta a la organización que desarrolla o suministra el software.

RIESGO	TIPO	DESCRIPCION	PROB.	EFECTO	ESTRATEGIA
No se cumplen las fechas establecidas en el cronograma.	Proyecto	Es probable que en la fecha del diagrama de Gantt no se cumplan al pie	Alta	Tolerable	Realizar un segundo cronograma que sea más flexible.

		de la letra.			
Cambio en los requerimientos del cliente.	Proyecto Producto	Riesgos de que haya cambios en la empresa.	Moderada	Tolerable	Realizar una revisión constante a los requerimientos. Programar reuniones con el personal.
No se cumple con los plazos de entrega del producto.	Producto	Los plazos de entrega del producto están determinados por el gerente general de la empresa.	Moderada	Serio	Agilizar los procesos de desarrollo del producto. Realizar una correcta planificación considerando el tiempo y los alcances del proyecto.
No existe la infraestructura necesaria para la implementación del sistema.	Proyecto	Es probable que no cuente con una infraestructura necesaria para realizar el sistema.	Alto	Tolerable	Solicitar con anticipación o contar con un equipo portátil.

Tabla 3.2: Análisis de Riesgo

Fuente: [Elaboración propia]

3.3 Juego

Durante esta etapa del proyecto se desarrollaron 3 iteraciones, cada una de ellas corresponde a un elemento del e-learning: plataforma, contenido y sistema de comunicación. A continuación se desglosan las actividades realizadas a cada una de estas etapas.

Las estrategias que se usan para el desarrollo de cada iteración fue construir en el principio los modelos de principios de la metodología UWE y posteriormente implementarlos utilizando como elemento central la Base de Datos, la solución por la que se optó para la persistencia de objetos fue hacer corresponder cada clase de modelo conceptual con una tabla de base de datos en donde las filas representan las instancias de los objetos y las columnas a los atributos de la clase. Las clases y sus métodos fueron implementados en el lenguaje PHP, finalmente se desarrollaron las páginas WEB en base a los modelos de presentación y de navegación.

3.3.1 Primera Iteración

Durante la primera iteración se desarrollaron los elementos pertenecientes a la plataforma Registro, Control y Seguimiento de los procesos. Las actividades realizadas durante esta iteración se observa en la siguiente tabla que constituye el Backlog del Sprint.

		SPRINT	INICIO	DURACION
ID	TAREA	TIPO	DIAS DE TRABAJO	ESTADO
		1	18 / 08 / 2015	30 Días
1.1	Realizar la planificación de la iteración	Planificación	4	Terminado
1.2	Analizar los requerimientos del Backlog del producto	Planificación	3	Terminado
1.3	Analizar los requerimientos de la iteración con casos de uso	Desarrollo	3	Terminado
1.4	Construir el modelo de contenidos	Desarrollo	3	Terminado

1.5	Construir el modelo de usuario	Desarrollo	2	Terminado
1.6	Construir el modelo de navegación	Desarrollo	2	Terminado
1.7	Construir el modelo de procesos	Desarrollo	2	Terminado
1.8	Construir el modelo de procesos	Desarrollo	2	Terminado
1.9	Desarrollar la pagina de ingreso al sistema.	Desarrollo	4	Terminado
1.10	Desarrollar el modulo de registros	Desarrollo	6	Terminado

Tabla 3.3: Primera Iteración

Fuente: [Elaboración Propia]

Las funcionalidades correspondientes al incremento de la iteración:

- Base de Datos del Sistema.
- Paginas de ingreso con acceso a los usuarios.
- Modulo de registros de usuarios.

3.3.2 Segunda Iteración

En la segunda iteración se desarrollaron los modulo de ingreso de materiales y herramientas en el sistema.

		SPRINT	INICIO	DURACION
		2	24 / 09 /2015	30 Días
ID	TAREA	TIPO	DIAS DE TRABAJO	ESTADO
2.1	Realizar la planificación de la iteración.	Planificación	4	Terminado

2.2	Analizar los requerimientos del Backlog del producto.	Planificación	3	Terminado
2.3	Analizar los requerimientos de la iteración con casos de uso.	Desarrollo	3	Terminado
2.4	Complementar el modulo de contenidos.	Desarrollo	1	Terminado
2.5	Complementar el modelo de usuario.	Desarrollo	1	Terminado
2.6	Completar el modelo de navegación	Desarrollo	1	Terminado
2.7	Construir el modelo de presentación	Desarrollo	1	Terminado
2.8	Desarrollo el modulo de ingreso de materiales.	Desarrollo	8	Terminado
2.9	Desarrollo el modulo de ingreso de herramientas	Desarrollo	8	Terminado

Tabla 3.4: Segunda Iteración

Fuente: [Elaboración Propia]

En la segunda iteración se desarrollaron las siguientes funcionalidades para el sistema:

- Modulo de ingreso de materiales al sistema.
- Modulo de ingreso de herramientas al sistema.

3.3.3 Tercera Iteración

En la tercera iteración se desarrollaron el modulo de inventarios para el sistema.

		SPRINT	INICIO	DURACION
		3	29 / 10 / 2015	30 Días
ID	TAREA	TIPO	DIAS DE TRABAJO	ESTADO
3.1	Realizar la planificación de la	Planificación	1	Terminado

	iteración.			
3.2	Analizar los requerimientos del producto del Backlog.	Planificación	1	Terminado
3.3	Analizar los requerimientos de la iteración con los casos de uso.	Desarrollo	1	Terminado
3.4	Complementar el modelo de contenidos.	Desarrollo	1	Terminado
3.5	Complementar el modelo de usuario	Desarrollo	1	Terminado
3.6	Complementar el modelo de navegación.	Desarrollo	1	Terminado
3.7	Construir el modelo de procesos	Desarrollo	1	Terminado
3.8	Construir el modelo de presentación.	Desarrollo	1	Terminado
3.9	Desarrollo del modulo de inventarios	Desarrollo	6	Terminado
3.10	Desarrollo de control de entradas y salidas de materiales de almacén.	Desarrollo	6	Terminado
3.11	Desarrollo del modulo de informes.	Desarrollo	6	Terminado
3.12	Desarrollo del modulo de búsqueda	Desarrollo	4	Terminado

Tabla 3.5: Tercera Iteración

Fuente: [Elaboración propia]

En la tercera iteración se desarrollaron las siguientes funcionalidades para el sistema.

- Desarrollo del modulo de búsqueda.

- Desarrollo del modulo de informes.
- Desarrollo del control de entradas y salidas de materiales de almacén.
- Desarrollo del modulo de inventarios.

3.3.4 Cuarta Iteración

Finalmente durante la cuarta iteración se desarrollaron los sistemas para los procesos y seguimientos. Las actividades durante esta iteración se las puede observar en el siguiente Backlog.

		SPRINT	INICIO	DURACION
		4	30 / 11 / 2015	29 Días
ID	TAREA	TIPO	DIAS	ESTADO
4.1	Realizar la planificación de la iteración.	Planificación	1	Terminado
4.2	Analizar los requerimientos del Backlog.	Planificación	1	Terminado
4.3	Analizar los requerimientos de la iteración.	Desarrollo	1	Terminado
4.4	Complementar el modelo de contenidos.	Desarrollo	1	Terminado
4.5	Complementar el modelo de usuario.	Desarrollo	1	Terminado
4.6	Complementar el modelo de navegación.	Desarrollo	1	Terminado
4.7	Construir el modelo de procesos.	Desarrollo	1	Terminado
4.8	Construir el modelo de presentación.	Desarrollo	1	Terminado
4.9	Desarrollo de control de ventas de	Desarrollo	5	Terminado

	los productos.			
4.10	Desarrollo de control de compras de materiales y herramientas.	Desarrollo	5	Terminado
4.11	Desarrollo de la interfaz de la aplicación móvil.	Desarrollo	2	Terminado
4.12	Desarrollo del modulo de inventario en la aplicación móvil.	Desarrollo	4	Terminado
4.13	Desarrollo del modulo de ingreso de materiales a los almacenes con la aplicación móvil.	Desarrollo	5	Terminado

Tabla 3.6: Cuarta Iteración

Fuente: [Elaboración propia]

En la cuarta iteración se desarrollaron las siguientes funcionalidades para el sistema.

- Desarrollo de control de ventas de los productos.
- Desarrollo de control de compras de materiales y herramientas.
- Desarrollo de la interfaz de la aplicación móvil.
- Desarrollo del modulo de inventario en la aplicación móvil.
- Desarrollo del modulo de ingreso de materiales a los almacenes con la aplicación móvil.

3.4 Modelo de Casos de Uso

Un modelo de caso de uso es un modelo de sistema que tiene actores, caso de uso y sus respectivas relaciones.

A continuación se describe las características de los actores identificados en el manejo e implementación del sistema.

ACTORES	DESCRIPCION
Gerente Propietario	<p>Es la persona que dentro del sistema web a desarrollarse tendrá todos los privilegios sobre el manejo de la información.</p> <p>Toma conocimiento del informe enviado al sistema en cada etapa y realiza la previa evaluación de los recursos.</p>
Personal Técnico	<p>Es la persona que enviara información solo para la parte de almacenes ya sea mediante una aplicación móvil o mediante el Gerente Propietario.</p>
Cliente	<p>Es la persona o institución encargada de realizar la compra de productos de la empresa.</p>
Proveedor	<p>Es la persona o institución encargada de materiales y herramientas a la empresa.</p>

Tabla 3.7: Descripción de los Actores

Fuente: [Elaboración propia]

3.4.1 Diagrama de Casos de Uso General

Los casos de uso describirán la secuencia de eventos de un actor, es decir es un documento narrativo de los todos actores que interviene en el sistema para un mejor entendimiento de la funcionalidad del sistema.

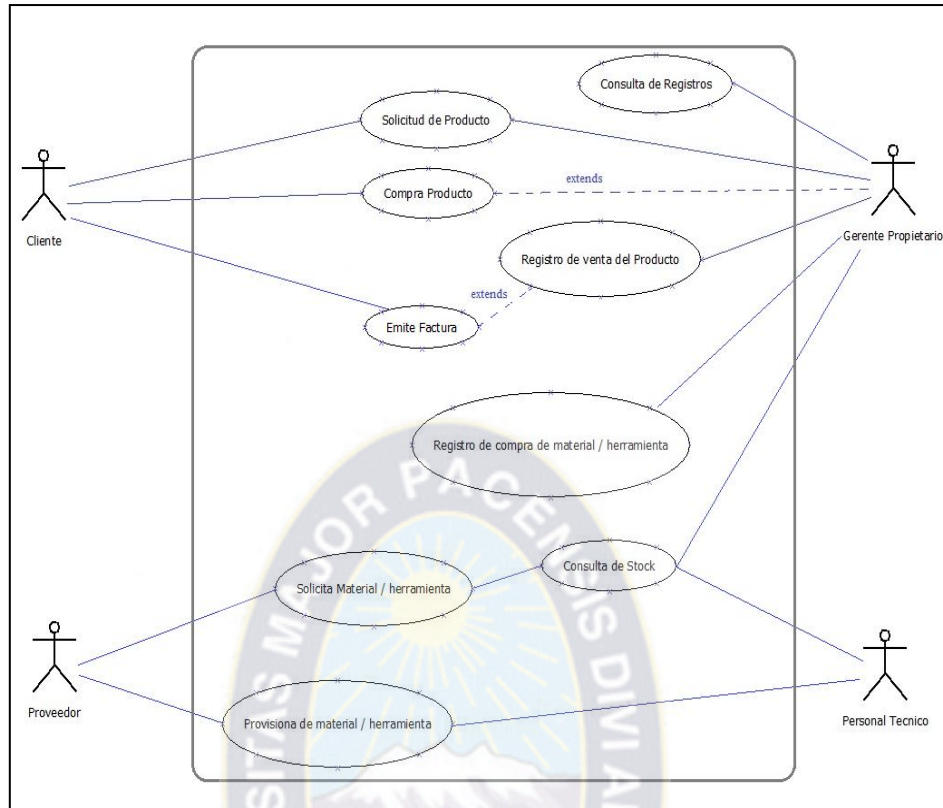


Figura 3.2: Caso de Uso General

Fuente: [Elaboración propia]

3.4.2 Descripción de Casos de Uso

En este diagrama se representara la interacción de los diferentes actores con el sistema basado en el modelo de negocio presentado.

3.4.2.1 Caso de uso: Registro de usuario en el sistema

Para el registro de usuarios al sistema el actor principal es el Gerente Propietario, el se encarga de llenar un formulario de datos del nuevo usuario, darle un rol y habilitar o en algún caso deshabilitar al usuario como se muestra en la siguiente figura.

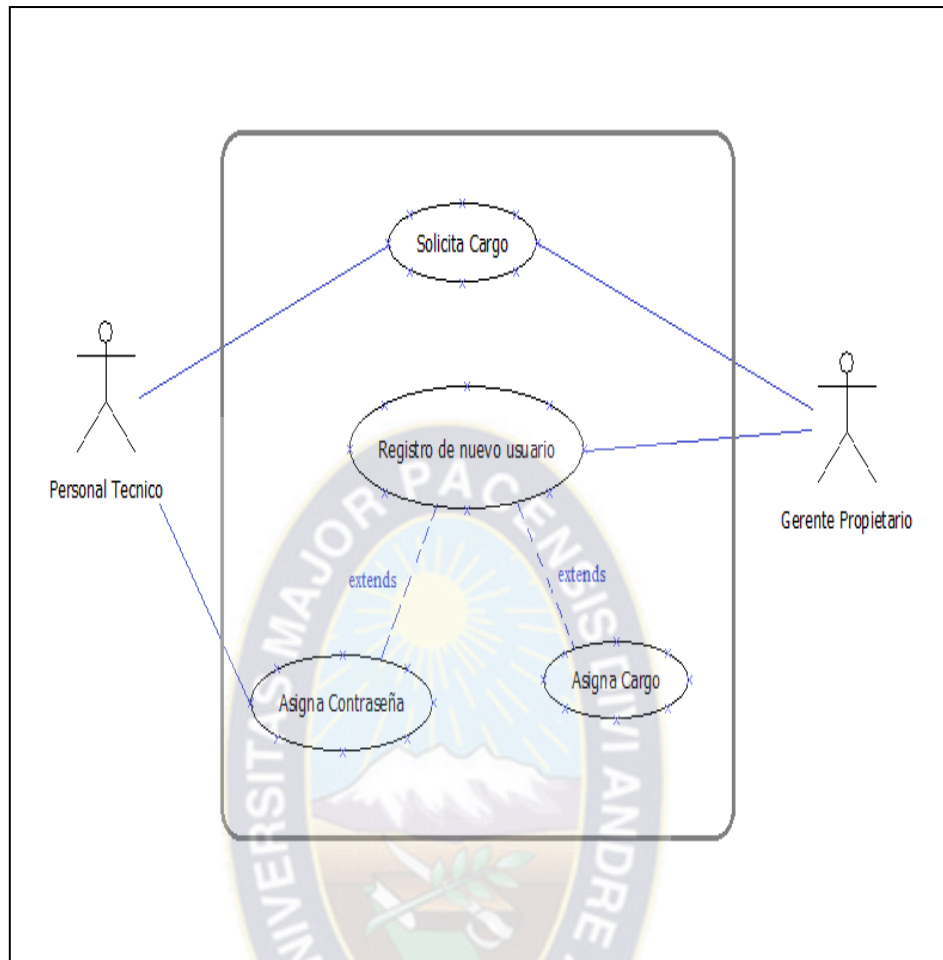


Figura 3.3: Caso de uso – Registro de usuario en el sistema

Fuente: [Elaboración propia]

A continuación se muestra o describe la especificación de los casos de uso representados por la figura anterior.

CASO DE USO	Registro de usuario en el sistema
ACTORES	Gerente Propietario, Personal Técnico
PRE - CONDICION	El Gerente propietario debe estar autenticado en el sistema y el técnico debe ser nuevo o que tenga un nuevo cargo
ESCENARIO BASICO	El caso de uso comienza cuando el usuario Gerente Propietario requiere registrar un nuevo Personal Técnico o realizar el

	<p>cambio de cargo de dicho Personal Técnico.</p> <ul style="list-style-type: none"> ➤ El usuario Gerente propietario selecciona la opción registro de un nuevo personal. ➤ El sistema despliega la interfaz de registro de nuevo Personal Técnico. ➤ El usuario Gerente Propietario ingresa los datos solicitados. ➤ El usuario Gerente Propietario selecciona guardar los datos. ➤ El usuario Gerente Propietario selecciona modificar el registro de un personal. ➤ El usuario Gerente Propietario realiza el cambio de cargo del personal. ➤ El usuario Gerente Propietario guarda la modificación de datos. ➤ Despliega alerta de conformidad.
<p>ESCENARIOS ITERATIVOS</p>	
<p>ALTERNATIVA – 1</p>	<p>Si en el escenario 8 del flujo básico el sistema encuentra datos que no concuerdan con los requerimientos, entonces el sistema despliega alertas que no fueron introducidos correctamente.</p>
<p>POST - CONDICION</p>	<p>Los datos de registro del nuevo personal están almacenados en la Base de Datos.</p>

Tabla 3.8: Registrar un nuevo usuario

Fuente: [Elaboración propia]

3.4.2.2 Diagrama de proceso: Registro de usuario en el sistema

En el proceso de logueo, es necesario contar con una cuenta de usuario o de cliente con su rol correspondiente. La contraseña y los datos se encuentran en la base de datos, los cuales se verifican antes de permitir al acceso al sistema.

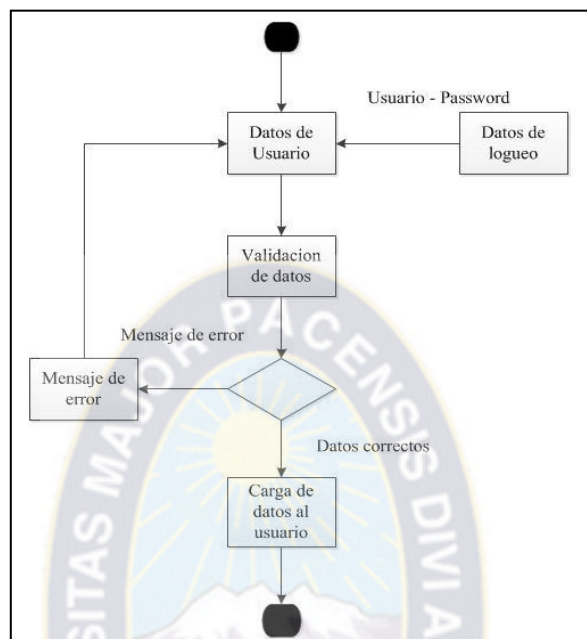


Figura 3.4: Diagrama de proceso – Registro de usuario en el sistema

Fuente: [Elaboración propia]

3.4.2.3 Caso de uso: Registro de Proveedores

Para el registro de proveedores al sistema el actor principal es el Gerente Propietario, el se encarga de llenar un formulario de datos del nuevo proveedor, para facilitar la compra de materiales o herramientas todo esto para la elaboración de productos derivados de los materiales.

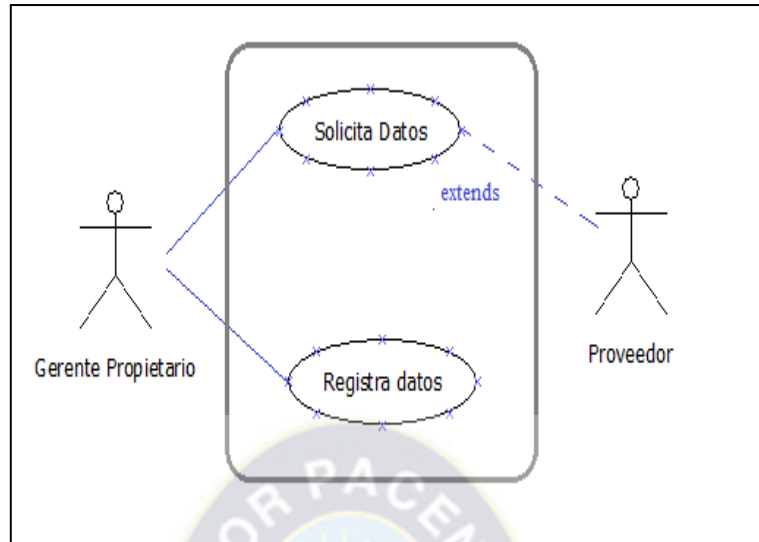


Figura 3.5: Caso de uso – Registro de Proveedores

Fuente: [Elaboración propia]

A continuación se muestra o describe la especificación de los casos de uso representados por la figura anterior.

CASO DE USO	Registro de proveedores
ACTORES	Gerente Propietario
PRE - CONDICION	El Gerente propietario debe estar autenticado en el sistema.
ESCENARIO BASICO	<p>El caso de uso comienza cuando el usuario Gerente Propietario requiere registrar un nuevo proveedor al sistema.</p> <ul style="list-style-type: none"> ➤ El usuario Gerente propietario selecciona la opción registro de un nuevo proveedor. ➤ El sistema despliega la interfaz de registro de nuevo proveedor. ➤ El usuario Gerente Propietario ingresa los datos solicitados. ➤ El usuario Gerente Propietario selecciona guardar los datos. ➤ Despliega alerta de conformidad.
ESCENARIOS ITERATIVOS	

ALTERNATIVA – 1	Si en el escenario 5 del flujo básico el sistema encuentra datos que no concuerdan con los requerimientos, entonces el sistema despliega alertas que no fueron introducidos correctamente.
POST - CONDICION	Los datos de registro del nuevo proveedor están almacenados en la Base de Datos.

Tabla 3.9: Registro de proveedores

Fuente: [Elaboración propia]

3.4.2.4 Diagrama de proceso: Registro de Proveedores

En el registro de proveedores es necesario estar logueado para luego realizar el registro correspondiente realizando sus ABM de los proveedores.

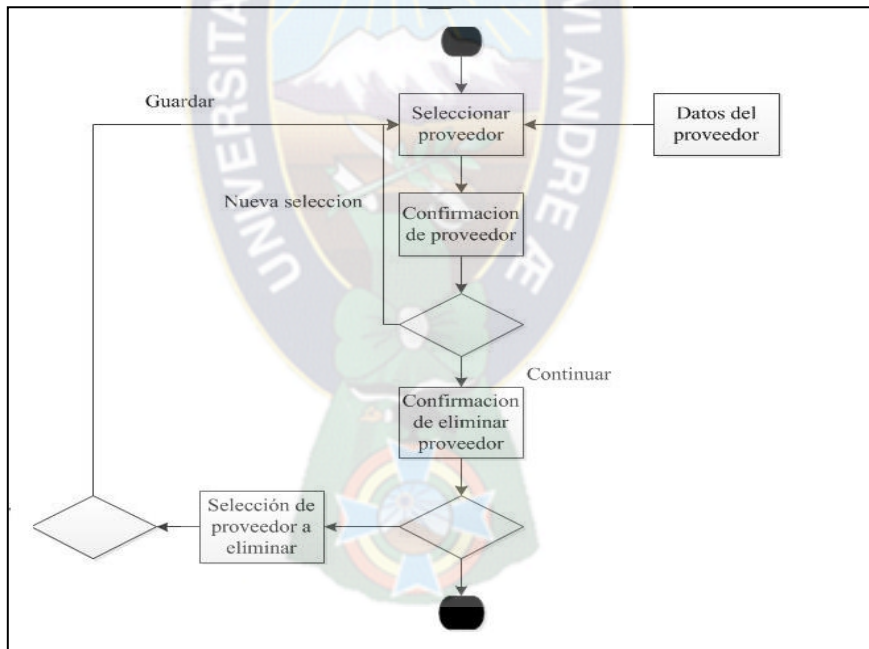


Figura 3.6: Diagrama de proceso – Registro de Proveedores

Fuente: [Elaboración propia]

3.4.2.5 Caso de uso: Registro de Clientes

Para el registro de clientes al sistema el actor principal es el Gerente Propietario, el se encarga de llenar un formulario de datos del nuevo cliente, para facilitar la venta de productos y el lugar de instalación de dicho producto.

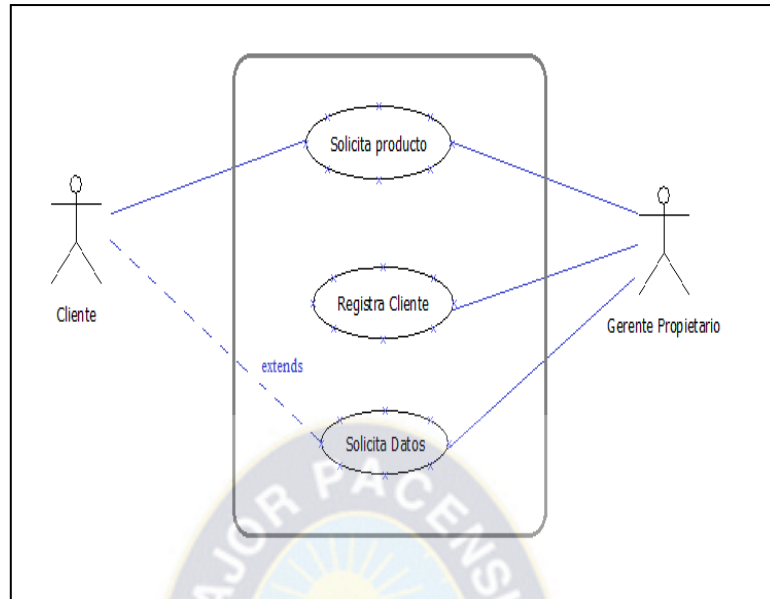


Figura 3.7: Caso de uso – Registro de Cliente

Fuente: [Elaboración propia]

A continuación se muestra o describe la especificación de los casos de uso representados por la figura anterior.

CASO DE USO	Registro de Clientes
ACTORES	Gerente Propietario
PRE - CONDICION	El Gerente propietario debe estar autenticado en el sistema.
ESCENARIO BASICO	<p>El caso de uso comienza cuando el usuario Gerente Propietario requiere registrar un nuevo cliente al sistema.</p> <ul style="list-style-type: none"> ➤ El usuario Gerente propietario selecciona la opción registro de un nuevo cliente. ➤ El sistema despliega la interfaz de registro de nuevo cliente. ➤ El usuario Gerente Propietario ingresa los datos solicitados. ➤ El usuario Gerente Propietario selecciona guardar los datos. ➤ Despliega alerta de conformidad.

ESCENARIOS ITERATIVOS	
ALTERNATIVA – 1	Si en el escenario 5 del flujo básico el sistema encuentra datos que no concuerdan con los requerimientos, entonces el sistema despliega alertas que no fueron introducidos correctamente.
POST - CONDICION	Los datos de registro del nuevo cliente están almacenados en la Base de Datos.

Tabla 3.10: Registro de clientes

Fuente: [Elaboración propia]

3.4.2.6 Diagrama de proceso: Registro de clientes

En el proceso de registro de cliente es necesario estar autenticado en el sistema para realizar las ABM del registro de clientes.

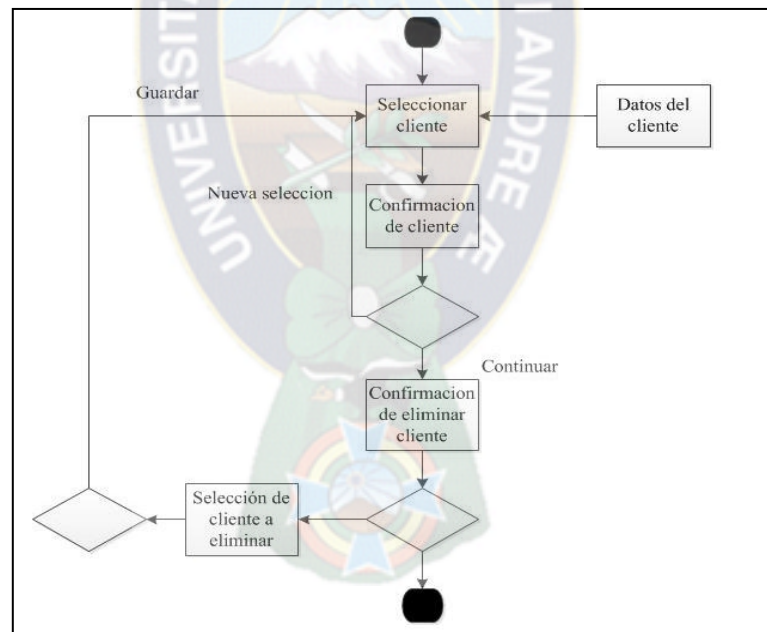


Figura 3.8: Diagrama de proceso – Registro de Cliente

Fuente: [Elaboración propia]

3.4.3. Modulo de Ventas

El modulo que representa a la venta de productos, tiene como actores principales al Gerente Propietario, Personal Técnico y el Cliente y su respectivo caso de uso esta descrita en la figura.

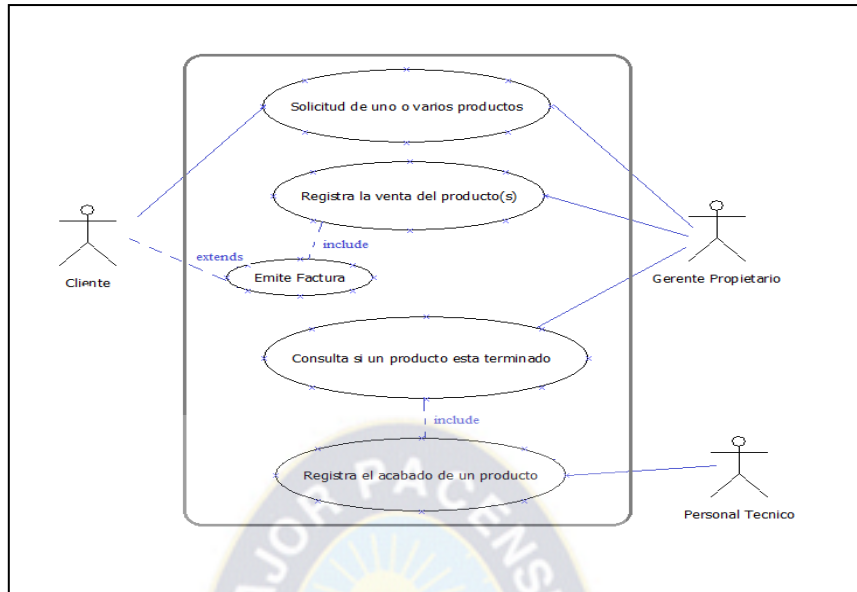


Figura 3.9: Caso de uso – Modulo de Ventas

Fuente: [Elaboración propia]

A continuación se muestra o describe la especificación de los casos de uso representados por la figura anterior.

CASO DE USO	Registro de Ventas
ACTORES	Gerente Propietario
PRE - CONDICION	El Gerente propietario debe estar autenticado en el sistema.
ESCENARIO BASICO	<p>El caso de uso comienza cuando el usuario Gerente Propietario requiere registrar una nueva venta al sistema.</p> <ul style="list-style-type: none"> ➤ El usuario Gerente propietario selecciona la opción registro de una nueva venta. ➤ El sistema despliega la interfaz de registro de una nueva venta. ➤ El usuario Gerente Propietario ingresa los datos solicitados. ➤ El usuario Gerente Propietario selecciona guardar los datos. ➤ Despliega alerta de conformidad.

ESCENARIOS ITERATIVOS	
ALTERNATIVA – 1	Si en el escenario 5 del flujo básico el sistema encuentra datos que no concuerdan con los requerimientos, entonces el sistema despliega alertas que no fueron introducidos correctamente.
POST - CONDICION	Los datos de registro de una nueva venta están almacenados en la Base de Datos.

Tabla 3.11: Registro de una nueva venta

Fuente: [Elaboración propia]

CASO DE USO	Registro del Terminado de un Producto
ACTORES	Personal Técnico
PRE - CONDICION	El Personal Técnico debe estar autenticado en el sistema.
ESCENARIO BASICO	<p>El caso de uso comienza cuando el usuario Personal Técnico requiere registrar el acabado de un Producto en el sistema.</p> <ul style="list-style-type: none"> ➤ El usuario Personal Técnico selecciona la opción registro del terminado de un producto. ➤ El sistema despliega la interfaz de registro del terminado de un producto. ➤ El usuario Personal Técnico ingresa los datos solicitados. ➤ El usuario Personal Técnico selecciona guardar los datos. ➤ Despliega alerta de conformidad.
ESCENARIOS ITERATIVOS	
ALTERNATIVA – 1	Si en el escenario 5 del flujo básico el sistema encuentra datos que no concuerdan con los requerimientos, entonces el sistema

	despliega alertas que no fueron introducidos correctamente.
POST - CONDICION	Los datos de registro del Terminado de un producto están almacenados en la Base de Datos.

Tabla 3.12: Registro del Terminado de un Producto

Fuente: [Elaboración propia]

CASO DE USO	Emitir Factura
ACTORES	Gerente Propietario
PRE - CONDICION	El Gerente propietario debe estar autenticado en el sistema.
ESCENARIO BASICO	<p>El caso de uso comienza cuando el usuario Gerente Propietario requiere registrar la venta al sistema.</p> <ul style="list-style-type: none"> ➤ El usuario Gerente propietario selecciona imprimir factura. ➤ El sistema despliega la interfaz de emisión de factura. ➤ El usuario Gerente Propietario selecciona imprimir factura. ➤ Despliega alerta de conformidad.
ESCENARIOS ITERATIVOS	
ALTERNATIVA – 1	Si en el escenario 4 del flujo básico el sistema encuentra datos que no concuerdan con los requerimientos, entonces el sistema despliega alertas que no fueron introducidos correctamente.
POST - CONDICION	Emite la factura.

Tabla 3.13: Emitir factura

Fuente: [Elaboración propia]

CASO DE USO	Consulta del estado de un Producto
-------------	------------------------------------

ACTORES	Gerente Propietario
PRE - CONDICION	El Gerente propietario debe estar autenticado en el sistema.
ESCENARIO BASICO	<p>El caso de uso comienza cuando el usuario Gerente Propietario requiere consultar acerca del estado de un producto al sistema.</p> <ul style="list-style-type: none"> ➤ El usuario Gerente propietario selecciona la opción consulta de un producto. ➤ El sistema despliega la interfaz de consulta de un producto. ➤ El usuario Gerente Propietario ingresa los datos solicitados. ➤ El usuario Gerente Propietario selecciona consultar los datos del producto. ➤ Despliega alerta de conformidad.
ESCENARIOS ITERATIVOS	
ALTERNATIVA – 1	Si en el escenario 5 del flujo básico el sistema encuentra datos que no concuerdan con los requerimientos, entonces el sistema despliega alertas que no fueron introducidos correctamente.
POST - CONDICION	Muestra los datos del producto solicitado.

Tabla 3.14: Consulta del estado de un producto

Fuente: [Elaboración propia]

3.4.3.1 Diagrama de proceso: modulo de ventas

En el proceso del modulo de ventas para el ingreso de una nueva venta y posteriormente realizar la facturación correspondiente para realizar esta operación es necesario estar loguedo en el sistema.

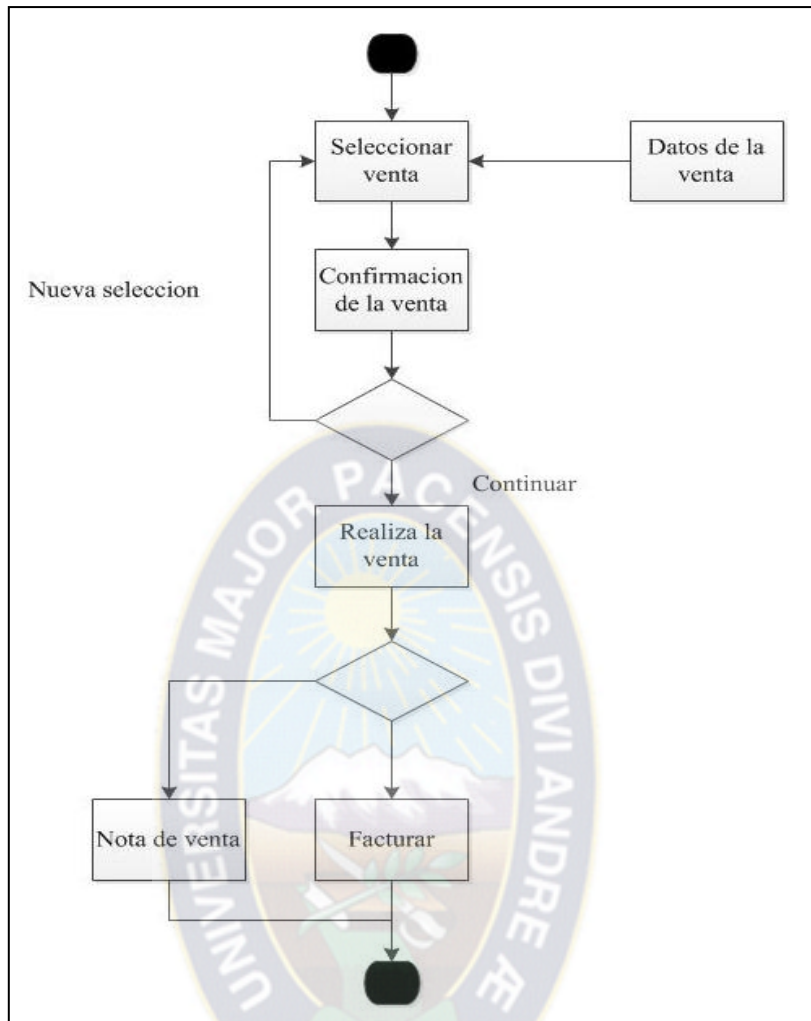


Figura 3.10: Diagrama de proceso – Modulo de Ventas

Fuente: [Elaboración propia]

3.4.4. Modulo de Inventarios

El modulo que representa al inventario, tiene como actores principales al Gerente Propietario y el Personal Técnico y su respectivo caso de uso esta descrita en la figura.

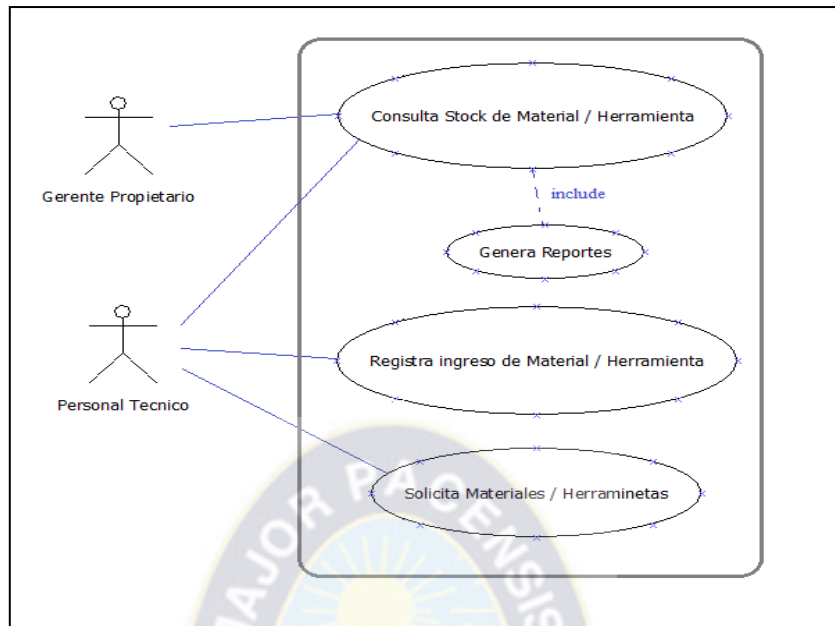


Figura 3.11: Caso de uso – Modulo de Inventarios

Fuente: [Elaboración propia]

A continuación se muestra o describe la especificación de los casos de uso representados por la figura anterior.

CASO DE USO	Registro de ingreso de Material / Herramienta
ACTORES	Personal Técnico
PRE - CONDICION	El Personal Técnico debe estar autenticado en el sistema.
ESCENARIO BASICO	<p>El caso de uso comienza cuando el usuario Personal Técnico requiere registrar un material / herramienta al almacén del sistema.</p> <ul style="list-style-type: none"> ➤ El usuario Personal Técnico selecciona la opción registrar un material / herramienta al almacén del sistema. ➤ El sistema despliega la interfaz de registro de material / herramienta. ➤ El usuario Personal Técnico ingresa los datos solicitados.

	<ul style="list-style-type: none"> ➤ El usuario Personal Técnico selecciona guardar los datos ingresados. ➤ Despliega alerta de conformidad.
ESCENARIOS ITERATIVOS	
ALTERNATIVA – 1	Si en el escenario 5 del flujo básico el sistema encuentra datos que no concuerdan con los requerimientos, entonces el sistema despliega alertas que no fueron introducidos correctamente.
POST - CONDICION	Los datos de registro de un material /herramienta se han ingresado al almacén de la Base de Datos de la empresa.

Tabla 3.15: Registro de ingreso de un material / herramienta

Fuente: [Elaboración propia]

CASO DE USO	Solicitud de un Material / Herramienta
ACTORES	Personal Técnico
PRE - CONDICION	El Personal Técnico debe estar autenticado en el sistema.
ESCENARIO BASICO	<p>El caso de uso comienza cuando el usuario Personal Técnico requiere un material / herramienta del almacén del sistema.</p> <ul style="list-style-type: none"> ➤ El usuario Personal Técnico selecciona la opción solicitar un material / herramienta del almacén del sistema. ➤ El sistema despliega la interfaz de solicitud de material / herramienta. ➤ El usuario Personal Técnico ingresa los datos solicitados. ➤ El usuario Personal Técnico selecciona pedir los datos ingresados. ➤ Despliega alerta de conformidad.
ESCENARIOS ITERATIVOS	

ALTERNATIVA – 1	Si en el escenario 5 del flujo básico el sistema encuentra datos que no concuerdan con los requerimientos, entonces el sistema despliega alertas que no fueron introducidos correctamente.
POST - CONDICION	Los datos de solicitud de un material /herramienta se han pedido al almacén de la Base de Datos de la empresa.

Tabla 3.16: Solicitud de un material / herramienta

Fuente: [Elaboración propia]

CASO DE USO	Consulta el Stock de un Material / Herramienta
ACTORES	Gerente Propietario y Personal Técnico
PRE - CONDICION	El Gerente Propietario y el Personal Técnico deben estar autenticados en el sistema.
ESCENARIO BASICO	<p>El caso de uso comienza cuando el usuario Gerente Propietario o el Personal Técnico requiere obtener información acerca de un material / herramienta del almacén del sistema.</p> <ul style="list-style-type: none"> ➤ El usuario Gerente Propietario o el Personal Técnico selecciona la opción consultar el informe de un material / herramienta del almacén del sistema. ➤ El sistema despliega la interfaz de consulta de material / herramienta. ➤ El usuario Gerente Propietario o el Personal Técnico ingresa los datos solicitados. ➤ El usuario Gerente Propietario o el Personal Técnico selecciona consultar. ➤ Despliega alerta de conformidad.
ESCENARIOS ITERATIVOS	
ALTERNATIVA – 1	Si en el escenario 5 del flujo básico el sistema encuentra datos que no concuerdan con los requerimientos, entonces el sistema

	despliega alertas que no fueron introducidos correctamente.
POST - CONDICION	Los datos de consulta de un material /herramienta se han obtenido del almacén de la Base de Datos.

Tabla 3.17: Consulta de Stock de un material / herramienta

Fuente: [Elaboración propia]

CASO DE USO	Generar reporte de un Material / Herramienta
ACTORES	Gerente Propietario y Personal Técnico
PRE - CONDICION	El Gerente Propietario y el Personal Técnico deben estar autenticados en el sistema.
ESCENARIO BASICO	<p>El caso de uso comienza cuando el usuario Gerente Propietario o el Personal Técnico requiere obtener información acerca de un material / herramienta del almacén del sistema.</p> <ul style="list-style-type: none"> ➤ El usuario Gerente Propietario o el Personal Técnico selecciona la opción imprimir el informe de un material / herramienta del almacén del sistema. ➤ El sistema despliega la interfaz de impresión de material / herramienta. ➤ El usuario Gerente Propietario o el Personal Técnico selecciona imprimir. ➤ Despliega alerta de conformidad.
ESCENARIOS ITERATIVOS	
ALTERNATIVA – 1	Si en el escenario 4 del flujo básico el sistema encuentra datos que no concuerdan con los requerimientos, entonces el sistema despliega alertas que no fueron introducidos correctamente.
POST - CONDICION	Obtiene el impreso del informe de un material /herramienta

que se han obtenido del almacén de la Base de Datos.

Tabla 3.18: Generar reporte de un material / herramienta

Fuente: [Elaboración propia]

3.4.4.1 Diagrama de proceso: Modulo de inventarios

En el proceso del modulo de inventario es necesario estar autenticado al sistema para poder realizar los reportes de los almacenes y del stock existente dentro de estos almacenes.

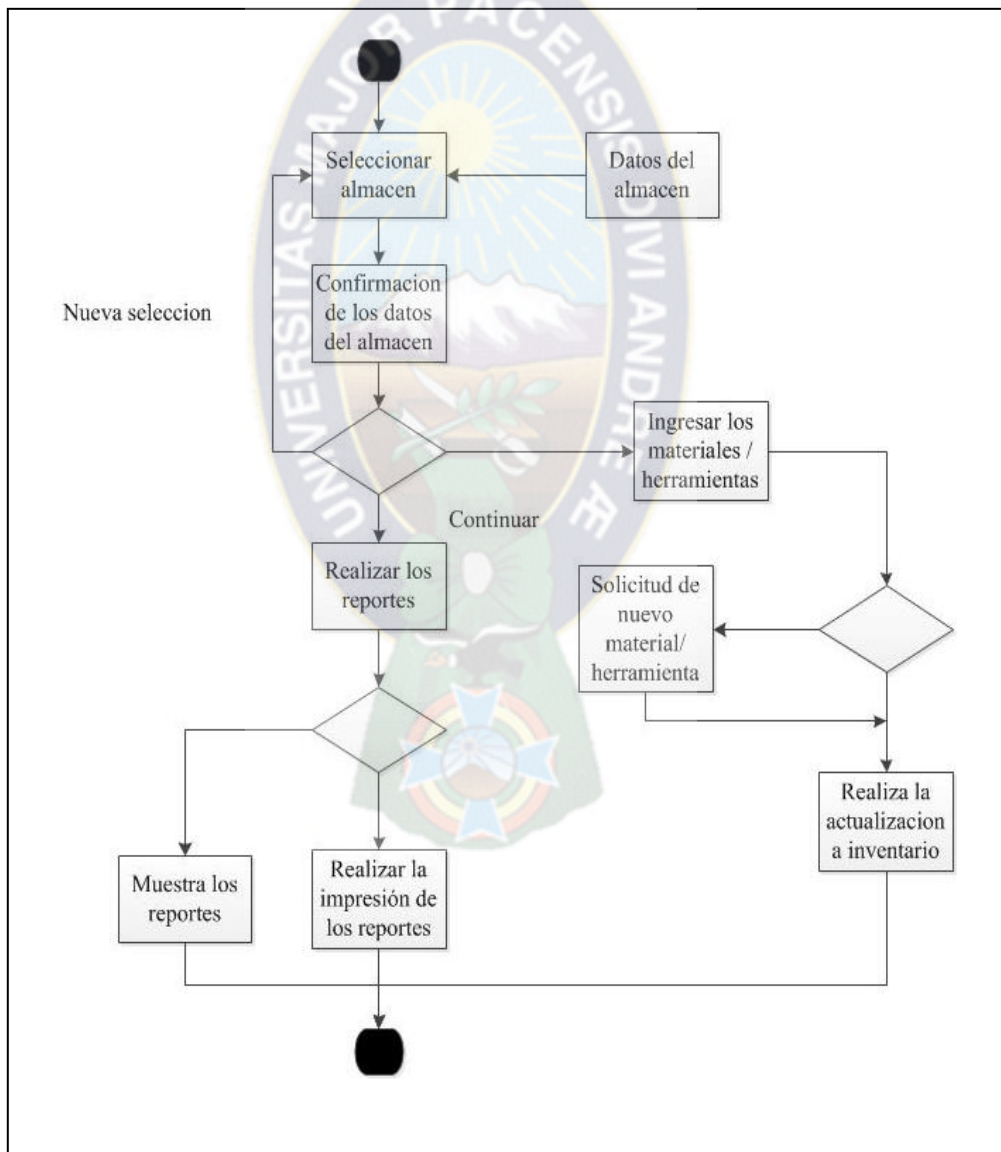


Figura 3.12: Diagrama de proceso – Modulo de Inventarios

Fuente: [Elaboración propia]

3.4.5. Modulo de Compras

El modulo que representa al inventario, tiene como actores principales al Gerente Propietario y el Proveedor y su respectivo caso de uso esta descrita en la figura.

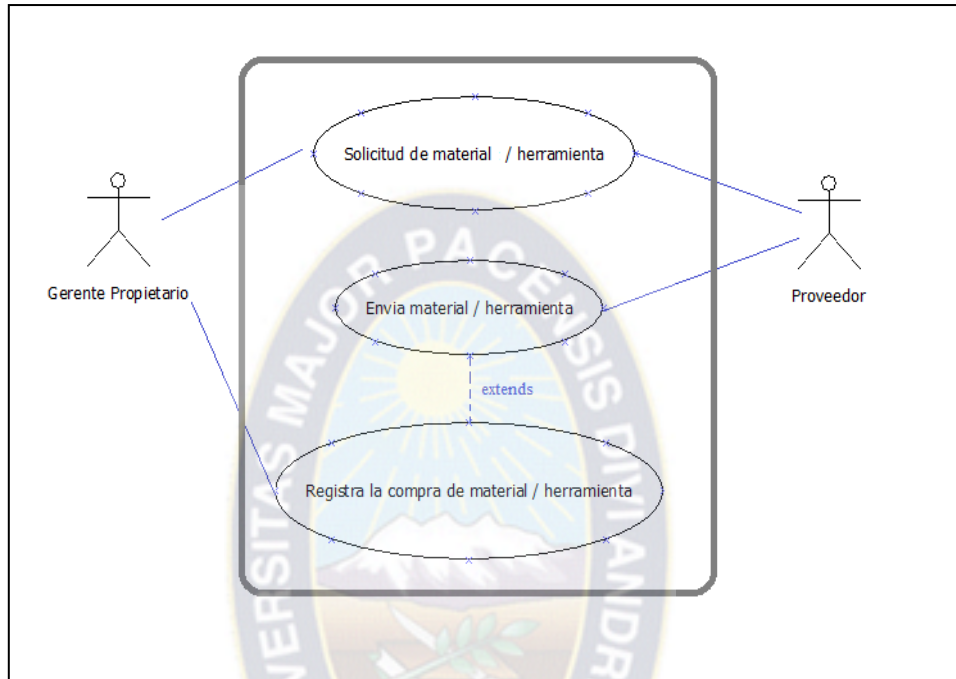


Figura 3.13: Caso de uso – Modulo de Compras

Fuente: [Elaboración propia]

A continuación se muestra o describe la especificación de los casos de uso representados por la figura anterior.

CASO DE USO	Registro de la compra de un Material / Herramienta
ACTORES	Gerente Propietario
PRE - CONDICION	El Gerente Propietario debe estar autenticado en el sistema.
ESCENARIO BASICO	<p>El caso de uso comienza cuando el usuario Gerente Propietario requiere registrar la compra de un material / herramienta al sistema.</p> <ul style="list-style-type: none"> ➤ El usuario Gerente Propietario selecciona la opción registrar una compra de un material / herramienta al

	<p>sistema.</p> <ul style="list-style-type: none"> ➤ El sistema despliega la interfaz de registro de la compra de un material / herramienta. ➤ El usuario Gerente Propietario ingresa los datos solicitados. ➤ El usuario Gerente Propietario selecciona guardar los datos ingresados. ➤ Despliega alerta de conformidad.
ESCENARIOS ITERATIVOS	
ALTERNATIVA – 1	Si en el escenario 5 del flujo básico el sistema encuentra datos que no concuerdan con los requerimientos, entonces el sistema despliega alertas que no fueron introducidos correctamente.
POST - CONDICION	Los datos de registro de la compra de un material /herramienta se han ingresado a la Base de Datos.

Tabla 3.19: Registro de la compra de un material / herramienta

Fuente: [Elaboración propia]

3.4.4.1 Diagrama de proceso: Modulo de compras

En el proceso del modulo de compras el usuario debe estar autenticado en el sistema donde se realiza las compras de los materiales o herramientas de los respectivos proveedores.

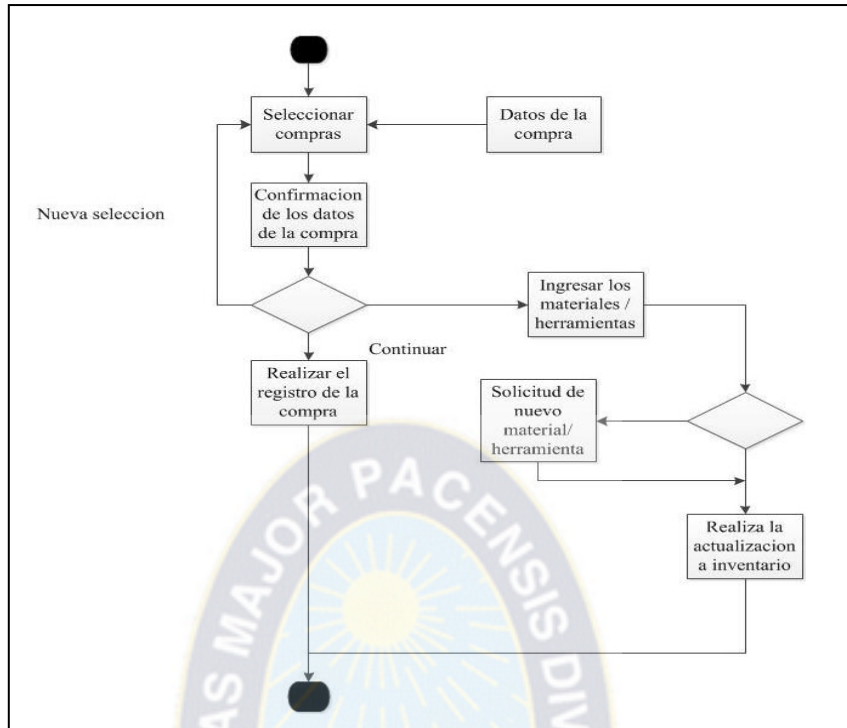


Figura 3.14: Diagrama de proceso – Modulo de Inventarios

Fuente: [Elaboración propia]

3.5. Modelo de contenidos (Clases)

Este modelo especifica cómo se encuentran relacionadas los contenidos del sistema, es decir, define la estructura de los datos que se encuentran almacenados en el sitio Web el modelo de contenidos contiene la información relevante almacenada en el sistema, como se estructura y como se relaciona. Esto se representa mediante un diagrama de clases de UML.

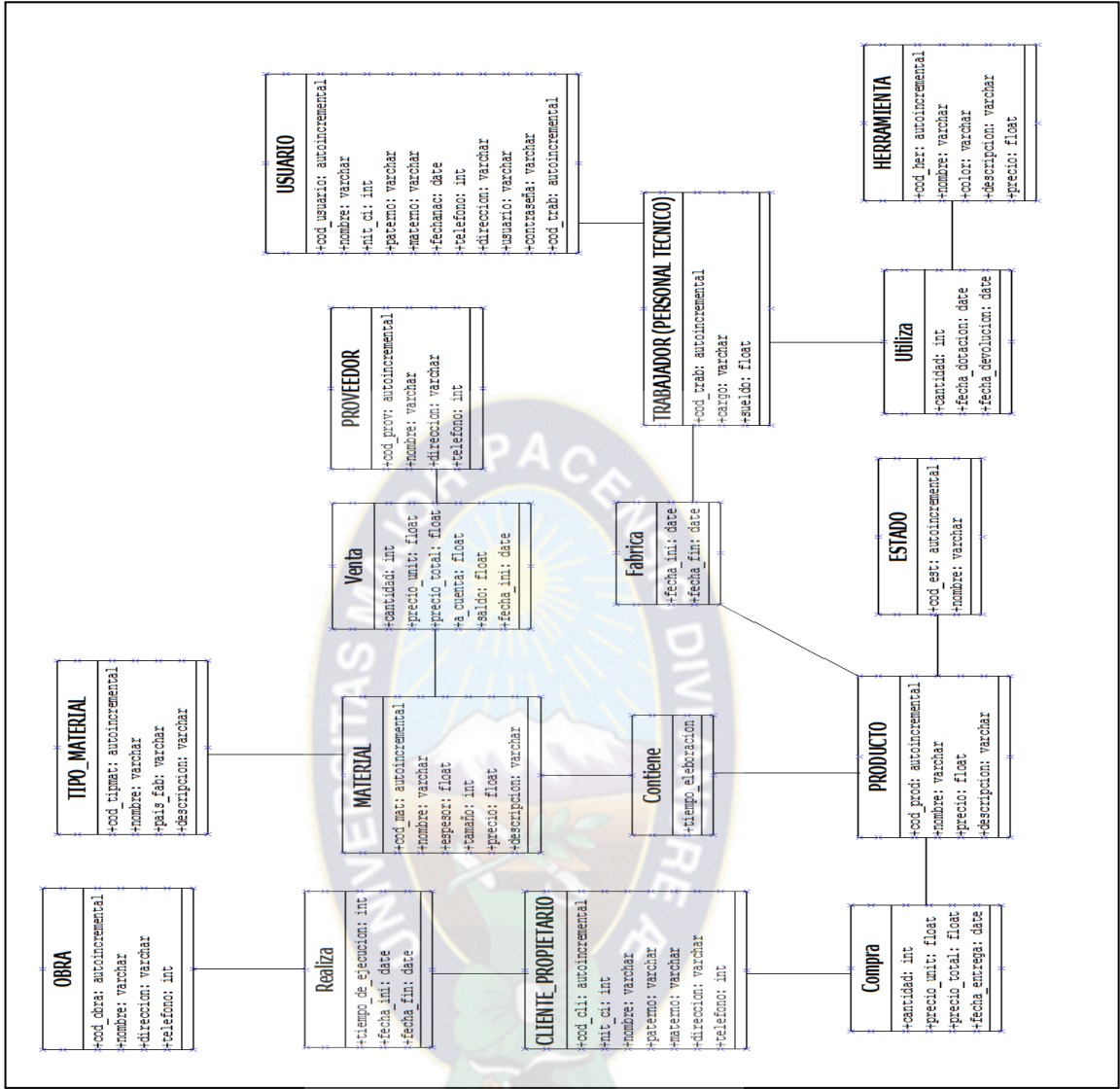


Figura 3.15: Diagrama Conceptual (Clases)

Fuente: [Elaboración propia]

3.6. Modelo Entidad Relación

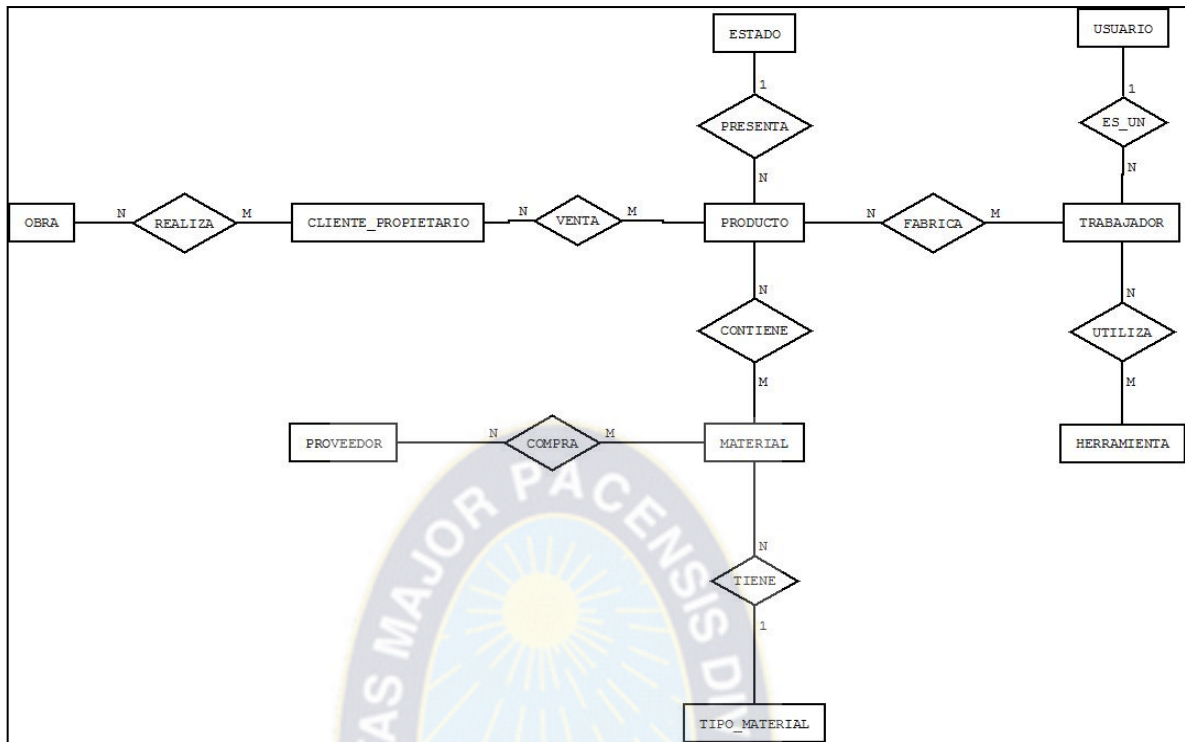


Figura 3.16: Modelo Entidad Relación

Fuente: [Elaboración propia]

3.6.1 Diccionario de Datos

Cliente_Propietario

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/> 1	cod_cli	int(11)			No	Ninguna	AUTO_INCREMENT
<input type="checkbox"/> 2	nit_ci	int(11)			No	Ninguna	
<input type="checkbox"/> 3	nombre	varchar(20)	utf8_general_ci		No	Ninguna	
<input type="checkbox"/> 4	paterno	varchar(20)	utf8_general_ci		No	Ninguna	
<input type="checkbox"/> 5	materno	varchar(20)	utf8_general_ci		No	Ninguna	
<input type="checkbox"/> 6	telefono	int(11)			Sí	NULL	
<input type="checkbox"/> 7	direccion	varchar(200)	utf8_general_ci		Sí	NULL	

Tabla 3.20: Diccionario de datos cliente

Fuente: [Elaboración propia]

Venta

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/>	1 cod_cli	int(11)			Sí	NULL	
<input type="checkbox"/>	2 cod_prod	int(11)			Sí	NULL	
<input type="checkbox"/>	3 cantidad	int(11)			Sí	NULL	
<input type="checkbox"/>	4 precio_unit	float			Sí	NULL	
<input type="checkbox"/>	5 precio_total	float			Sí	NULL	
<input type="checkbox"/>	6 acuenta	float			Sí	NULL	
<input type="checkbox"/>	7 saldo	float			Sí	NULL	
<input type="checkbox"/>	8 fecha_ini	date			Sí	NULL	
<input type="checkbox"/>	9 fecha_fin	date			Sí	NULL	
<input type="checkbox"/>	10 fecha_actual	date			No	Ninguna	

Tabla 3.21: Diccionario de datos venta

Fuente: [Elaboración propia]

Contiene

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/>	1 cod_prod	int(11)			Sí	NULL	
<input type="checkbox"/>	2 cod_mat	int(11)			No	Ninguna	
<input type="checkbox"/>	3 tiempo_elaboracion	int(11)			No	Ninguna	

Tabla 3.22: Diccionario de datos contiene

Fuente: [Elaboración propia]

Estado

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/>	1 cod_est	int(11)			No	Ninguna	AUTO_INCREMENT
<input type="checkbox"/>	2 nombre	varchar(20)	utf8_general_ci		No	Ninguna	

Tabla 3.23: Diccionario de datos estado

Fuente: [Elaboración propia]

Fabrica

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/> 1	<u>cod_trab</u>	int(11)			Sí	NULL	
<input type="checkbox"/> 2	<u>cod_prod</u>	int(11)			Sí	NULL	
<input type="checkbox"/> 3	<u>fecha_ini</u>	date			Sí	NULL	
<input type="checkbox"/> 4	<u>fecha_fin</u>	date			Sí	NULL	

Tabla 3.24: Diccionario de datos fabrica

Fuente: [Elaboración propia]

Herramienta

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/> 1	<u>cod_her</u>	int(11)			No	Ninguna	AUTO_INCREMENT
<input type="checkbox"/> 2	<u>nombre</u>	varchar(60)	utf8_general_ci		No	Ninguna	
<input type="checkbox"/> 3	<u>color</u>	varchar(20)	utf8_general_ci		No	Ninguna	
<input type="checkbox"/> 4	<u>descripcion</u>	varchar(200)	utf8_general_ci		Sí	NULL	
<input type="checkbox"/> 5	<u>precio</u>	float			Sí	NULL	

Tabla 3.25: Diccionario de datos herramienta

Fuente: [Elaboración propia]

Material

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/> 1	<u>cod_mat</u>	int(11)			No	Ninguna	AUTO_INCREMENT
<input type="checkbox"/> 2	<u>cod_tipmat</u>	int(11)			Sí	NULL	
<input type="checkbox"/> 3	<u>nombre</u>	varchar(20)	utf8_general_ci		No	Ninguna	
<input type="checkbox"/> 4	<u>espesor</u>	int(11)			No	Ninguna	
<input type="checkbox"/> 5	<u>tamaño</u>	int(11)			Sí	NULL	
<input type="checkbox"/> 6	<u>descripcion</u>	varchar(200)	utf8_general_ci		Sí	NULL	
<input type="checkbox"/> 7	<u>precio</u>	float			Sí	NULL	

Tabla 3.26: Diccionario de datos material

Fuente: [Elaboración propia]

Obra

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/> 1	<u>cod_obra</u>	int(11)			No	Ninguna	AUTO_INCREMENT
<input type="checkbox"/> 2	nombre	varchar(20)	utf8_general_ci		No	Ninguna	
<input type="checkbox"/> 3	direccion	varchar(200)	utf8_general_ci		No	Ninguna	
<input type="checkbox"/> 4	telefono	int(11)			Sí	NULL	

Tabla 3.27: Diccionario de datos obra

Fuente: [Elaboración propia]

Producto

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/> 1	<u>cod_prod</u>	int(11)			No	Ninguna	AUTO_INCREMENT
<input type="checkbox"/> 2	nombre	varchar(20)	utf8_general_ci		No	Ninguna	
<input type="checkbox"/> 3	precio	float			Sí	NULL	
<input type="checkbox"/> 4	descripcion	varchar(200)	utf8_general_ci		Sí	NULL	
<input type="checkbox"/> 5	cod_est	int(11)			Sí	NULL	

Tabla 3.28: Diccionario de datos producto

Fuente: [Elaboración propia]

Proveedor

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/> 1	<u>cod_prov</u>	int(11)			No	Ninguna	AUTO_INCREMENT
<input type="checkbox"/> 2	nombre	varchar(20)	utf8_general_ci		No	Ninguna	
<input type="checkbox"/> 3	direccion	varchar(200)	utf8_general_ci		No	Ninguna	
<input type="checkbox"/> 4	telefono	int(11)			Sí	NULL	

Tabla 3.29: Diccionario de datos proveedor

Fuente: [Elaboración propia]

Compra

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/> 1	cod_prov	int(11)			Sí	NULL	
<input type="checkbox"/> 2	cod_mat	int(11)			No	Ninguna	
<input type="checkbox"/> 3	cantidad	int(11)			Sí	NULL	
<input type="checkbox"/> 4	precio_unit	float			Sí	NULL	
<input type="checkbox"/> 5	precio_total	float			Sí	NULL	
<input type="checkbox"/> 6	fecha_entrega	date			Sí	NULL	

Tabla 3.30: Diccionario de datos compra

Fuente: [Elaboración propia]

Realiza

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/> 1	cod_obra	int(11)			Sí	NULL	
<input type="checkbox"/> 2	cod_cli	int(11)			Sí	NULL	
<input type="checkbox"/> 3	tiempo_ejecucion	int(11)			Sí	NULL	
<input type="checkbox"/> 4	fecha_ini	date			Sí	NULL	
<input type="checkbox"/> 5	fecha_fin	date			Sí	NULL	

Tabla 3.31: Diccionario de datos realiza

Fuente: [Elaboración propia]

Tipo_material

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
<input type="checkbox"/> 1	cod_tipmat	int(11)			No	Ninguna	AUTO_INCREMENT
<input type="checkbox"/> 2	nombre	varchar(20)	utf8_general_ci		No	Ninguna	
<input type="checkbox"/> 3	pais_fab	varchar(20)	utf8_general_ci		Sí	NULL	
<input type="checkbox"/> 4	descripcion	varchar(200)	utf8_general_ci		No	Ninguna	

Tabla 3.32: Diccionario de datos tipo_material

Fuente: [Elaboración propia]

Trabajador (Personal Técnico)

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
1	cod_trab	int(11)			No	Ninguna	AUTO_INCREMENT
2	cargo	varchar(20)	utf8_general_ci		Sí	NULL	
3	sueldo	float			Sí	NULL	

Tabla 3.33: Diccionario de datos trabajador

Fuente: [Elaboración propia]

Usuario

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
1	cod_usuario	int(11)			No	Ninguna	AUTO_INCREMENT
2	nombre	varchar(20)	utf8_general_ci		No	Ninguna	
3	nit_ci	int(11)			No	Ninguna	
4	paterno	varchar(20)	utf8_general_ci		No	Ninguna	
5	materno	varchar(20)	utf8_general_ci		No	Ninguna	
6	fechanac	date			Sí	NULL	
7	telefono	int(11)			Sí	NULL	
8	direccion	varchar(200)	utf8_general_ci		Sí	NULL	
9	usuario	varchar(20)	utf8_general_ci		No	Ninguna	
10	contraseña	varchar(20)	utf8_general_ci		No	Ninguna	
11	cod_trab	int(10)			No	Ninguna	

Tabla 3.34: Diccionario de datos usuario

Fuente: [Elaboración propia]

Utiliza

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra
1	cod_trab	int(11)			Sí	NULL	
2	cod_her	int(11)			Sí	NULL	
3	cantidad	int(11)			No	Ninguna	
4	fecha_dotacion	date			Sí	NULL	
5	fecha_devolucion	date			Sí	NULL	

Tabla 3.35: Diccionario de datos utiliza

Fuente: [Elaboración propia]

3.7. Modelo Físico

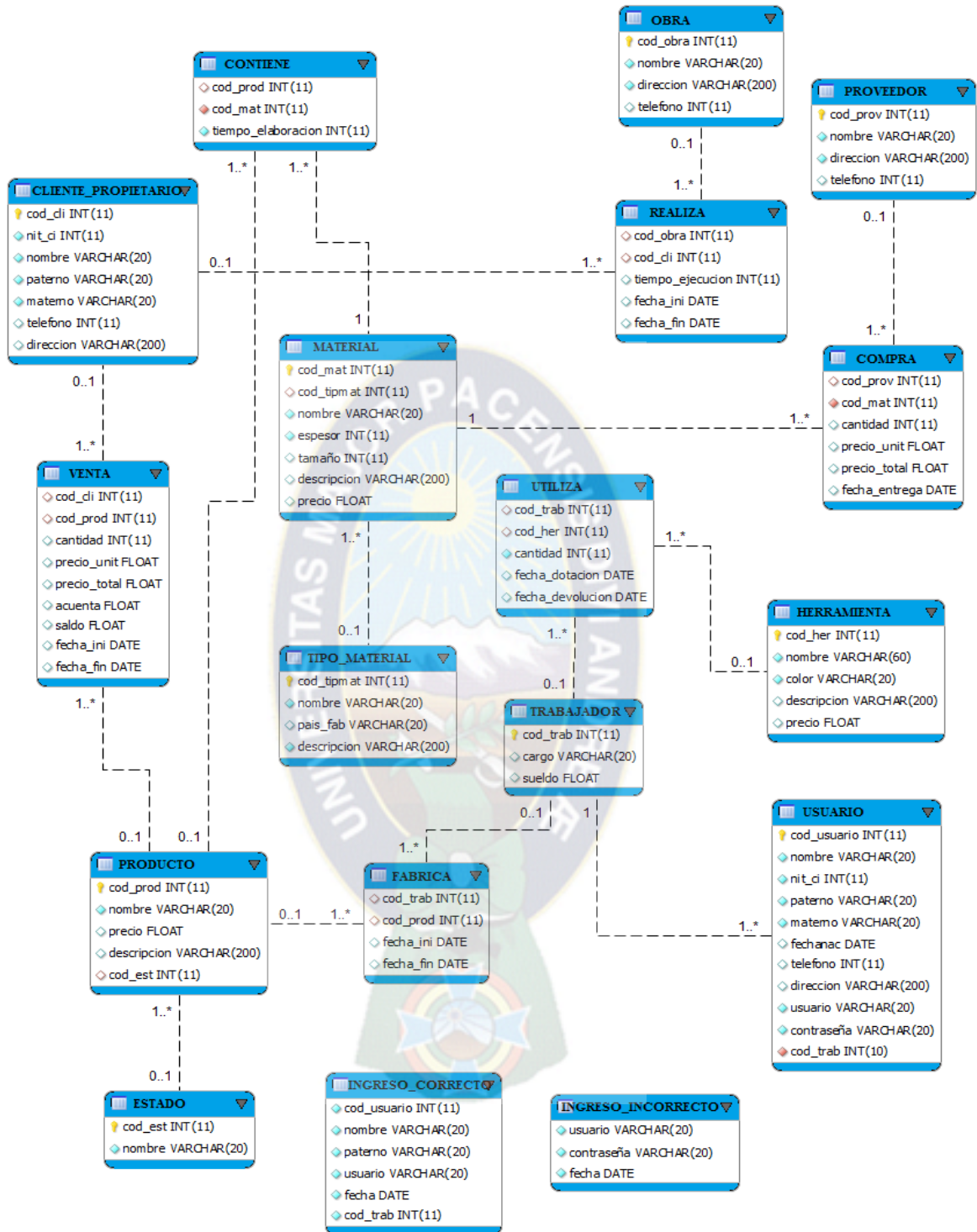


Figura 3.17: Modelo Físico

Fuente: [Elaboración propia]

3.8. Modelo Navegacional

El diseño navegacional del sistema, muestra las opciones de navegación y proceso, partiendo del proceso básico del ingreso al sistema.

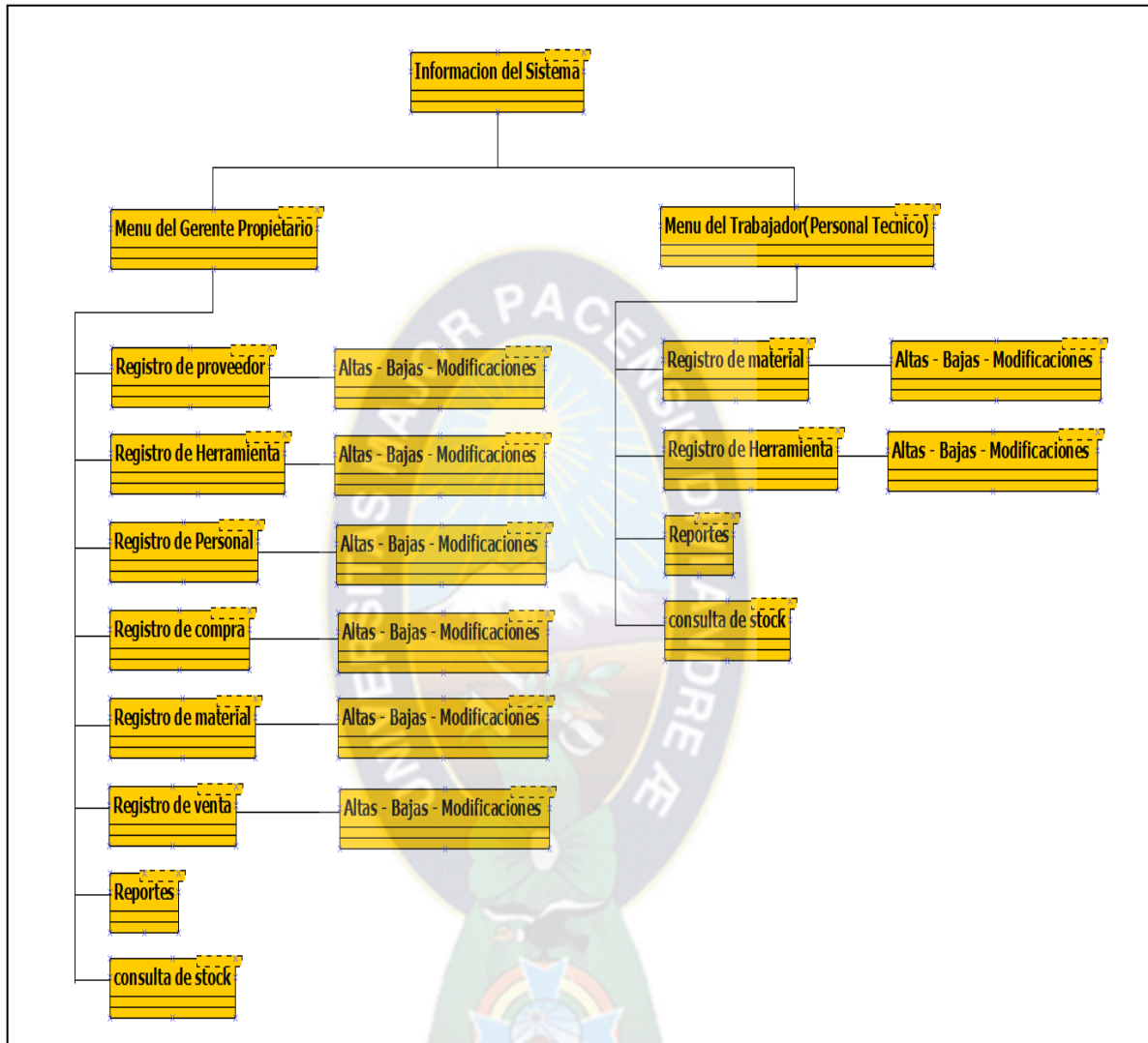


Figura 3.18: Diseño navegacional – Ingreso al sistema

Fuente: [Elaboración propia]

3.8.1. Modulo de Inventarios

El modulo de Inventarios, contiene los datos necesarios para poder ser registrados las entradas y salidas de los materiales y herramientas, con la cual podrá consultarse el stock que existe en los almacenes y sus movimientos.

La siguiente figura hace referencia al registro o modificación de las entradas y salidas de los materiales y herramientas de la empresa para la generación de reportes.

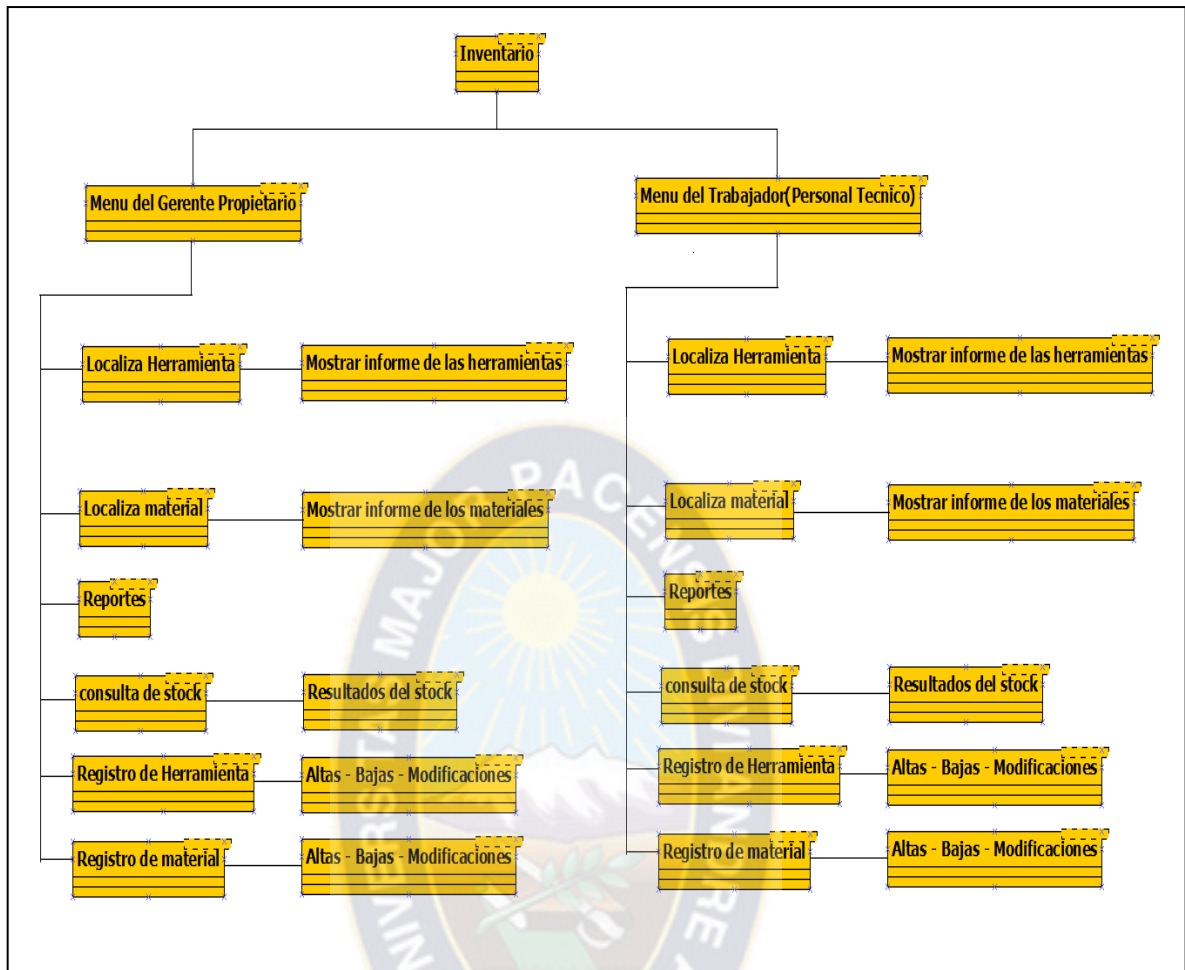


Figura 3.19: Diseño navegacional – Módulo de inventario

Fuente: [Elaboración propia]

3.8.2. Módulo de Compras

El módulo de compras, contiene los datos necesarios para poder ser registrados las compras de los materiales y herramientas, con la cual podrá ser ingresadas al inventario del sistema.

La siguiente figura hace referencia al registro de la compra de los materiales y herramientas para la elaboración de productos.

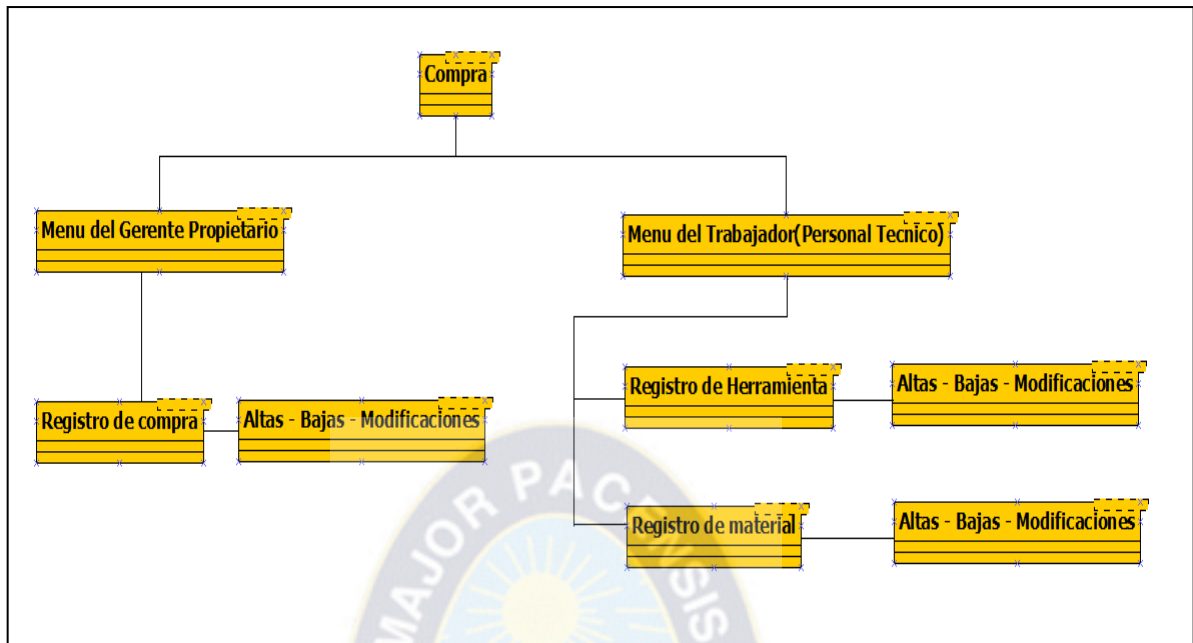


Figura 3.20: Diseño navegacional – Módulo de compras

Fuente: [Elaboración propia]

3.8.3. Módulo de Ventas

El módulo de ventas, contiene los datos necesarios para poder ser registrados las ventas de los productos en el sistema.

La siguiente figura hace referencia al registro de venta de productos.

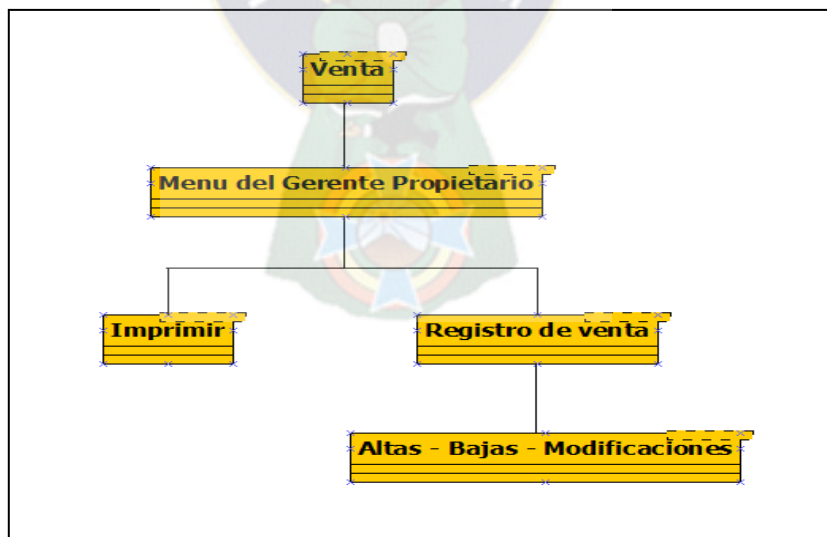


Figura 3.21: Diseño navegacional – Módulo de venta

Fuente: [Elaboración propia]

3.9. Diagramas de presentación

Los diagramas de presentación, que se muestran a continuación, permiten visualizar el resultado final de las interfaces que tendrá el sistema cumpliendo con los requerimientos planteados.

3.9.1. Modulo de Información de Ventas

En el diagrama de presentación para el modulo de la información, muestra las interfaces de listado de las ventas realizadas así como el registro de los mismos.

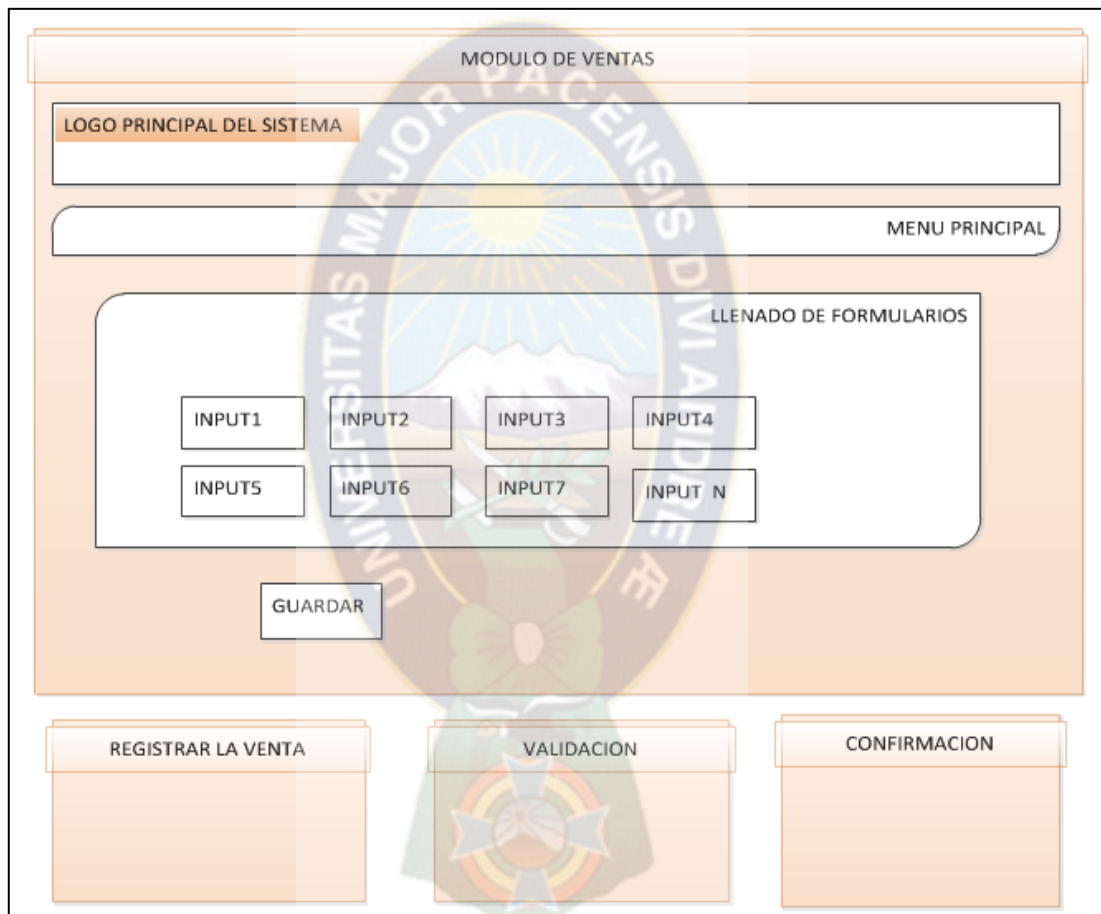


Figura 3.22: Diagrama de presentación – Información de ventas

Fuente: [Elaboración propia]

3.9.2. Modulo de Información de Materiales y Herramientas

En el diagrama de presentación para el modulo de la información de materiales y de las herramientas, muestra las interfaces de listado de los materiales y herramientas así como el registro de los mismos.

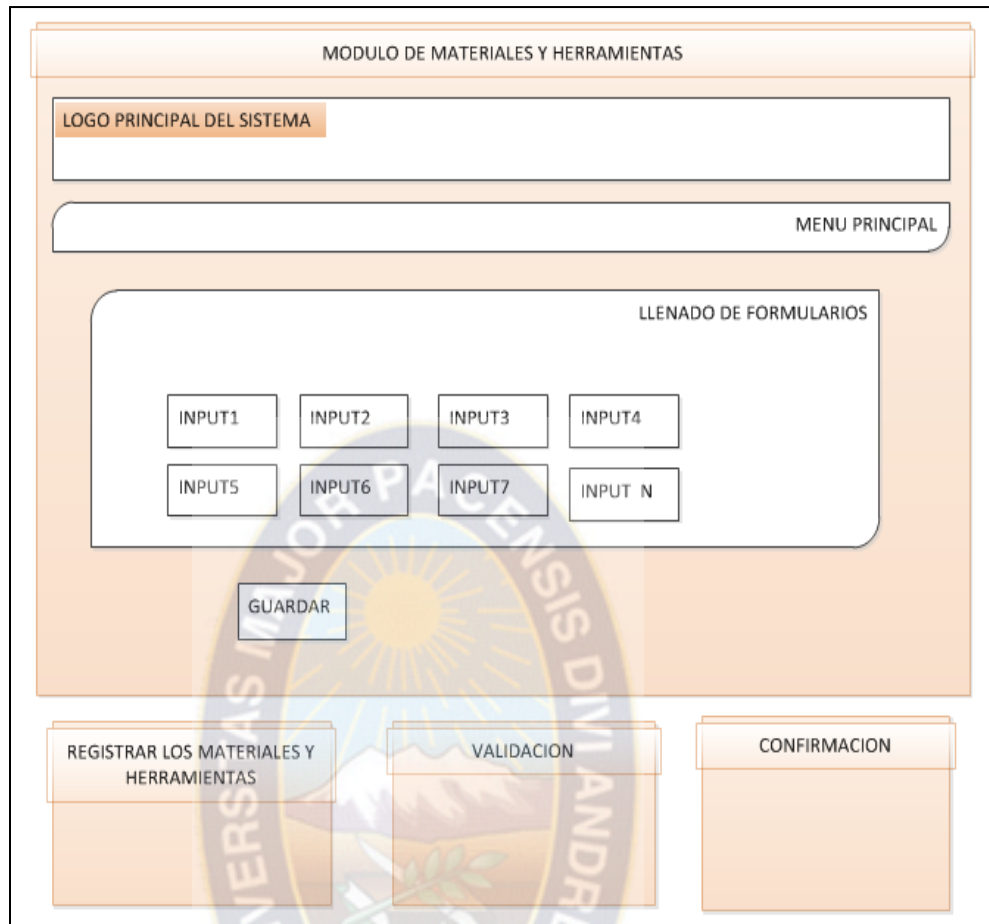


Figura 3.23: Diagrama de presentación – información de material y herramientas

Fuente: [Elaboración propia]

3.9.3. Modulo de Inventario

En el diagrama de presentación para el modulo de inventario, muestra las interfaces de listado de productos existentes en almacén así como el registro de los mismos.



Figura 3.24: Diagrama de presentación – Inventarios

Fuente: [Elaboración propia]

3.9.4. Modulo de Información de compras

En el diagrama de presentación para el modulo de compras, muestra las interfaces de listado de materiales y herramientas que se adquirieron así como el registro de los mismos.



Figura 3.25: Diagrama de presentación – Informe de compras

Fuente: [Elaboración propia]

3.10. Post-Juego

Durante esta última etapa se realizaron las actividades de prueba de la aplicación web, para esto se propusieron las políticas de seguridad para el sistema. Se obtuvieron métricas se realizo el diseño de ayuda para los usuarios.

3.11. Modelo e Implementación del programa

3.11.1. Fase de Transición

Esta fase es en la que se realizan los test de aceptación, para asegurar al funcionamiento final de un determinado caso de uso.

3.11.2. Pruebas de aceptación

Las pruebas de aceptación nos muestran las acciones o peticiones que realiza el usuario y las respuestas de parte del sistema web. La siguiente tabla muestra el caso de uso de prueba para el primer caso de uso.

CASO DE PRUEBA	Gestión de información de ventas
ROLES	Administrador
PRE - CONDICIONES	La información de ventas debe estar almacenado en la base de datos, en caso de una nueva venta se realizara el registro.
DESCRIPCION	
PETICIONES	RESPUESTAS
<ol style="list-style-type: none">1. El usuario accede al sistema web2. El usuario selecciona la opción de venta de productos.4. El usuario elige la opción de agregar una nueva venta.6. El usuario ingresa los datos correspondientes a la venta.	<ol style="list-style-type: none">3. El sistema despliega el formulario de acuerdo a la sección.5. El sistema web realiza la validación de la información, si existe un campo vacio un mensaje de error.7. El sistema web guarda las ventas.

8. El usuario guarda la información.	
POST – CONDICIONES	Los datos son guardados en la base de datos con un código único de venta.

Tabla 3.36: Caso de prueba – Gestión de información de ventas

Fuente: [Elaboración propia]

CASO DE PRUEBA	Gestión de información de inventarios
ROLES	Administrador – Personal técnico
PRE - CONDICIONES	La información de inventarios debe estar almacenado en la base de datos, en caso de una nuevos ingresos ya sea de materiales y herramientas se realizaran el registro respectivo.
DESCRIPCION	
PETICIONES	RESPUESTAS
3. El usuario accede al sistema web 4. El usuario selecciona la opción de ingreso de material o herramientas. 6. El usuario elige la opción de agregar al almacén. 8. El usuario ingresa los datos correspondientes al inventario. 9. El usuario guarda la información.	4. El sistema despliega el formulario de acuerdo a la sección. 7. El sistema web realiza la validación de la información, si existe un campo vacio un mensaje de error. 9. El sistema web guarda los datos ingresados.
POST – CONDICIONES	Los datos son guardados en la base de datos.

Tabla 3.37: Caso de prueba – Gestión de información de inventarios

Fuente: [Elaboración propia]

CASO DE PRUEBA	Gestión de información de compras
ROLES	Administrador
PRE - CONDICIONES	La información de compras debe estar almacenado en la base de datos, en caso de una nuevos ingresos se realizaran el registro respectivo.
DESCRIPCION	
PETICIONES	RESPUESTAS
5. El usuario accede al sistema web 6. El usuario selecciona la opción de ingreso de compras. 8. El usuario elige la opción de agregar la compra de materiales. 10. El usuario ingresa los datos correspondientes a las compras. 10. El usuario guarda la información.	5. El sistema despliega el formulario de acuerdo a la sección. 9. El sistema web realiza la validación de la información, si existe un campo vacio un mensaje de error. 11. El sistema web guarda los datos ingresados.
POST – CONDICIONES	Los datos son guardados en la base de datos con un código único de compras.

Tabla 3.38: Caso de prueba – Gestión de información de compras

Fuente: [Elaboración propia]

3.12. Diseño e interfaz

3.13. Fase de construcción

A continuación se muestra, las interfaces resultantes de la fase de construcción o desarrollo del sistema web.

Cada usuario con diferentes roles, inician sesión para tener acceso a la información que contiene.



Figura 3.26: Inicio se sesión del sistema

Fuente: [Elaboración propia]

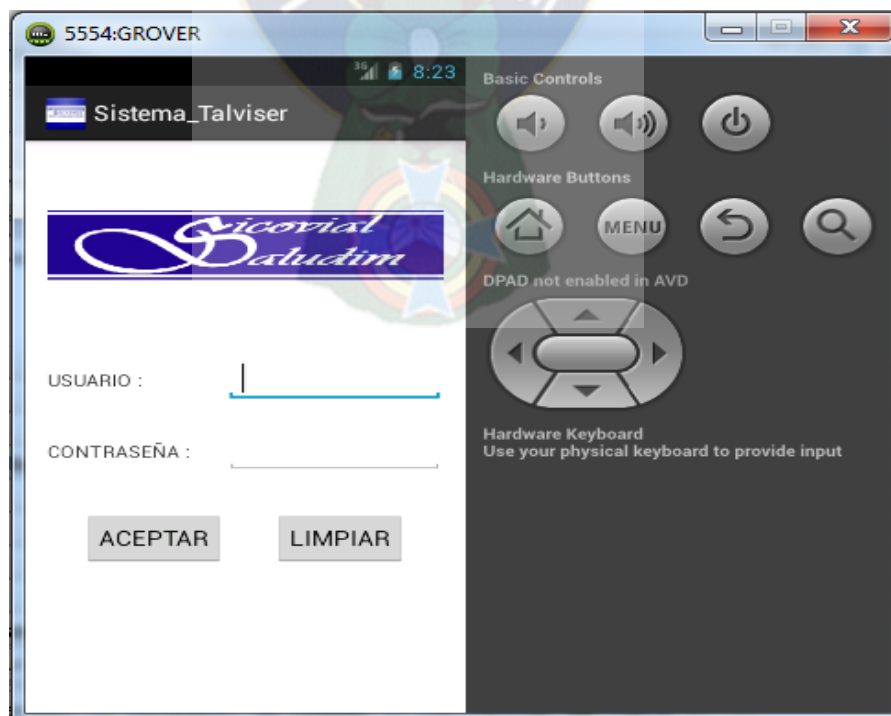


Figura 3.27: Inicio se sesión del sistema en un dispositivo móvil

Fuente: [Elaboración propia]

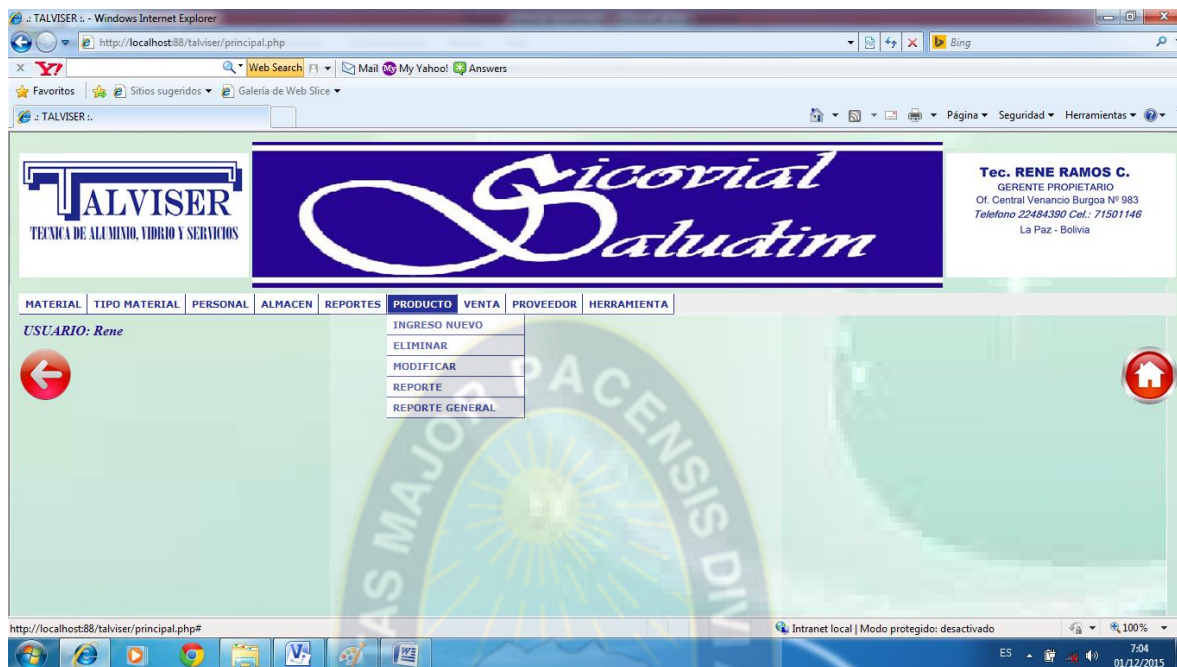


Figura 3.28: Menú principal del Administrador del sistema

Fuente: [Elaboración propia]



Figura 3.29: Menú principal del Personal técnico del sistema

Fuente: [Elaboración propia]



Figura 3.30: Ingreso de materiales o herramientas a almacenes

Fuente: [Elaboración propia]

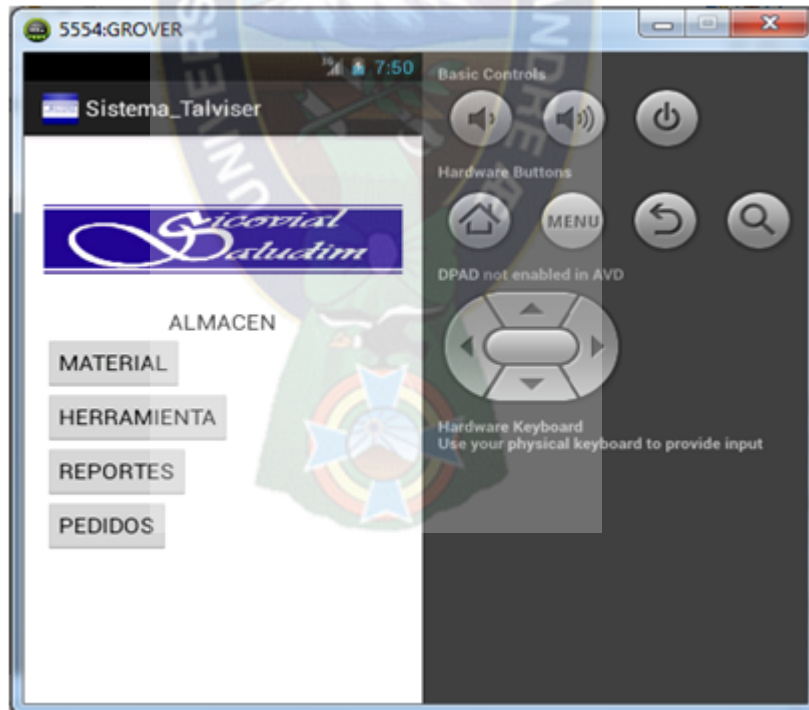


Figura 3.31: Ingreso de materiales o herramientas a almacenes en un dispositivo móvil

Fuente: [Elaboración propia]

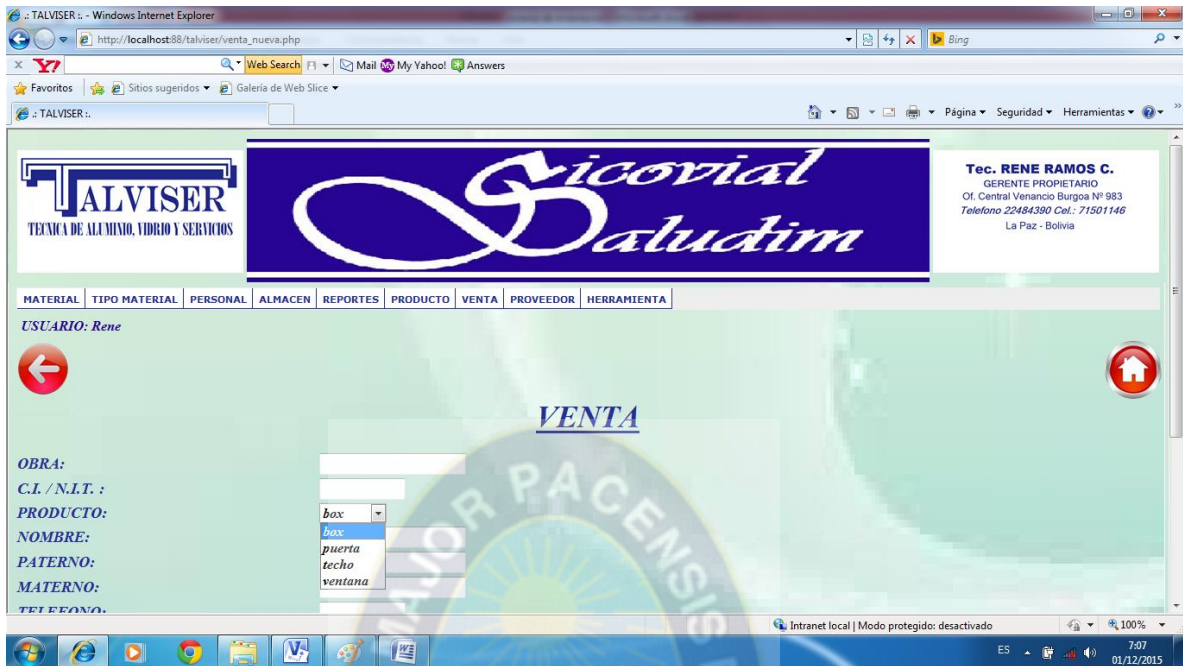


Figura 3.32: Ingreso de ventas al sistema

Fuente: [Elaboración propia]

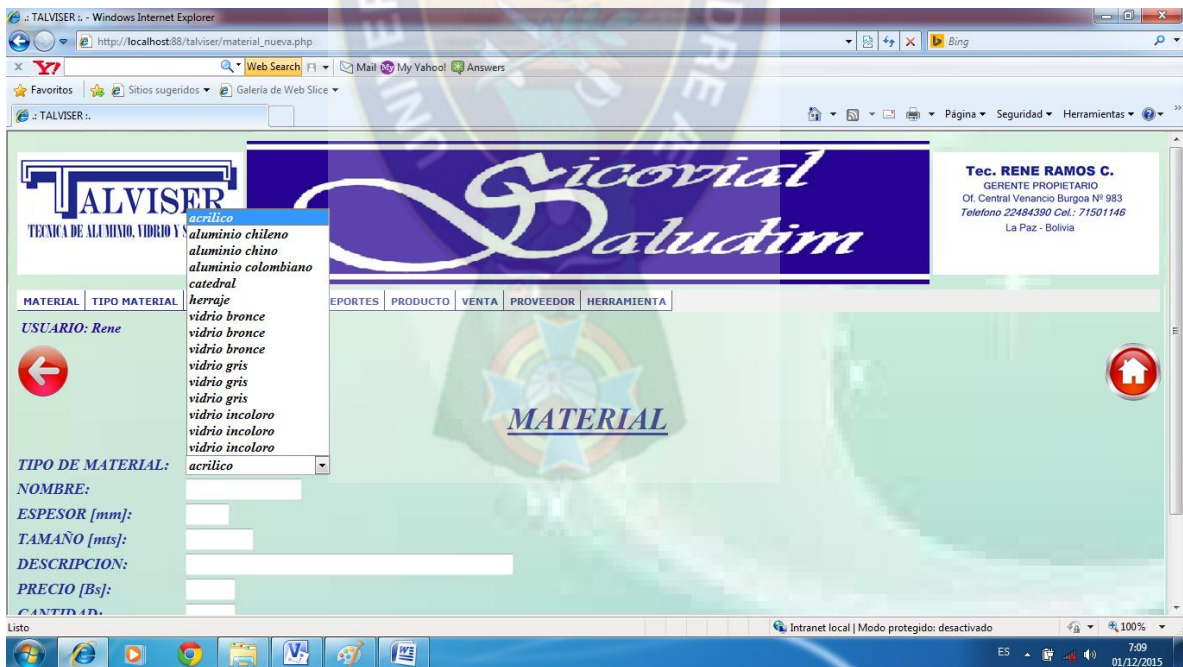


Figura 3.33: Ingreso de compras al sistema

Fuente: [Elaboración propia]

CAPITULO IV

METRICAS DE CALIDAD Y SEGURIDAD

4.1. Calidad

4.1.1. Tecnica Web Site ISO 9126

La calidad del sistema web será definida en base a un conjunto de propiedades inherentes a un producto, que le confieren capacidad para satisfacer necesidades implícitas o explícitas.

El objetivo principal de esta técnica es alcanzar la calidad necesaria para satisfacer las necesidades del cliente. Se evalúan dos ámbitos: el producto final y los procesos. Estos ámbitos son conocidos como: modelos de referencia. La norma ISO 9126 que hace referencia al modelo de calidad del producto del software y la ISO 14594 hace referencia a la calidad de evaluación del producto de software.

La calidad según esta norma web site ISO 9126 puede ser pedida de acuerdo a los factores:

- Usabilidad
- Eficacia
- Flexibilidad
- Facilidad de uso
- Portabilidad

4.1.1.1. Usabilidad

La usabilidad consiste en la evaluación del esfuerzo necesario que el usuario invertirá para usar el sistema, en base a su comprensión y estructura lógica que el sistema tiene. Esta comprensión por parte de los usuarios con relación al sistema evalúa los siguientes pasos:

- Comprensibilidad
- Facilidad de aprender
- Operabilidad

Se realizan encuestas a los usuarios finales sobre el manejo, la comprensión y la facilidad de aprender el sistema para medir la usabilidad según la siguiente tabla:

PREGUNTAS	RESPUESTAS		PORCENTAJE
	SI	NO	
¿El acceso al sistema es complicado?	0	10	100 %
¿Son comprensibles las respuestas del sistema?	1	9	90%
¿Son complicadas los procesos que realiza el sistema?	10	0	100%
¿El sistema tiene interfaces entendibles?	9	1	90%
¿La interfaz del sistema es agradable a la vista?	10	0	100%
¿Son satisfactorias las respuestas que el sistema devuelve?	8	2	80%
¿El sistema reduce su tiempo de trabajo?	9	1	90%
¿Es difícil aprender a manejar el sistema?	1	9	90%
¿El sistema satisface las necesidades que usted requiere?	9	1	90%
¿Utiliza el sistema con facilidad?	9	1	90%
PROMEDIO			92%

Tabla 4.1: Encuesta de usabilidad del sistema

Fuente: [Elaboración propia]

4.1.1.2. Mantenibilidad

Para hallar la mantenibilidad del sistema se utiliza el índice de madurez de software (IMS), que proporciona una indicación de la estabilidad de un producto de software (basado en los cambios que ocurren con cada versión del producto).

Se determina la siguiente fórmula para hallar el (IMS).

$$IMS = [Mt - (Fc + Fa + Fe)] / Mt$$

Donde:

Mt: Numero de módulos total de la versión actual.

Fc: Numero de módulos de la versión actual que se cambiaron.

Fa: Numero de módulos de la versión actual que se añadieron.

Fe: Número de módulos de la versión anterior que se eliminaron en la versión actual.

Por lo tanto se aplica al sistema y da el siguiente resultado de índice de madurez.

$$IMS = [6 - (1 + 0 + 0)] / 6$$

$$IMS = 0.83 * 100\% = 83\%$$

Es decir el sistema empieza a estabilizarse en un 83%.

4.1.1.3. Funcionalidad

4.1.1.3.1 Punto función

Los puntos función miden el software desde una perspectiva del usuario, dejando de lado los detalles de la codificación.

Es una técnica totalmente independiente de todas las consideraciones del lenguaje y ha sido aplicada en más de 240 lenguajes diferentes. Se supone que FPA evalúa con fiabilidad.

El proceso consta de dos etapas:

Paso 1.- Se identifican las funciones disponibles para el usuario y se organizan en cinco grupos.

Paso 2.- Se ajusta este total de acuerdo con unas características del entorno.

Haremos uso de cinco características de dominios de información y se proporcionan las cuentas en la posición apropiada de la tabla. Los valores de los dominios de la información, y se definen de la siguiente manera.

Numero de entradas de usuario.- Se cuenta cada entrada de usuario que proporciona diferentes datos orientados a la aplicación. Las entradas de deberían diferenciar de las peticiones, las cuales se encuentran de forma separada.

Numero de salidas de usuario.- En este contexto la salida de usuario se refiere a informes, pantallas, mensajes de error, etc. Los elementos de datos particulares dentro de un informe no se cuentan de forma separada.

Numero de peticiones de usuario.- Una petición se define como una entrada interactiva que produce la generación de alguna respuesta del software inmediata en forma de salida interactiva. Se cuenta cada petición por separado.

Numero de archivos.- Se cuenta cada archivo maestro lógico (se refiere a un grupo lógico de datos que puede ser parte de una gran base de datos o un archivo independiente).

Numero de interfaces externos.- Se cuentan todas las interfaces legibles por la maquina (por ejemplo: archivos de dato de disco) que se utilizan para transmitir información a otro sistema (Pressman, 2005).

De acuerdo a la información y comportamiento del sistema se obtuvieron los siguientes datos:

PARAMETROS DE ENTRADA	CUENTA
Nro. de entradas de usuario	80
Nro. de salidas de usuario	50
Nro. de peticiones de usuario	30
Nro. de archivos	60
Nro. de interfaces externos	20

Tabla 4.2: Entradas para el cálculo de funcionalidad según punto función

Fuente: [Elaboración propia]

Los puntos función se calculan con la ayuda de la anterior tabla considerando los factores de ponderación medio.

PARAMETROS DE MEDICION	CUENTA		FACTOR DE PONDERACION MEDIO	TOTALES
Nro. de entradas de usuario	80	x	5	400
Nro. de salidas de usuario	50	x	7	350
Nro. de peticiones de usuario	30	x	4	120

Nro. de archivos	60	x	10	600
Nro. de interfaces externos	20	x	6	120
CUENTA TOTAL				1590

Tabla 4.3: Cuenta total con factor de ponderación medio

Fuente: [Elaboración propia]

Calcular la relación para calcular el punto función.

$$PF = CUENTA\ TOTAL * (Grado\ de\ confiabilidad + Tasa\ de\ error * \sum F).$$

Donde:

PF = Medida de confiabilidad.

CUENTA TOTAL = Es la suma del valor de las entradas, salidas, peticiones, interfaces externas y archivos.

Grado de confiabilidad = Confiabilidad estimada del sistema.

Tasa de error = Probabilidad subjetiva estimada del dominio de la información este error es del 1%.

$\sum F$ = Son los valores de ajuste de complejidad que toman los valores de la tabla anterior que dan respuesta en la siguiente tabla que se muestra a continuación.

FACTORES	Sin Importancia	Incremental	Moderado	Medio	Significativo	Esencial
	0	1	2	3	4	5
FACTOR						
¿Requiere el sistema copias de seguridad y de información?						x
¿Se requiere comunicación de datos?					x	

¿Existen funciones de procesos distribuidos?		x				
¿Es crítico el rendimiento?			x			
¿El sistema web será ejecutado en el S. O. actual?					x	
¿Se requiere una entrada interactiva para el sistema?				x		
¿Se requiere que el sistema tenga entrada de datos con múltiples ventanas?					x	
¿Son complejas las entradas, salidas, los archivos o las peticiones?				x		
¿Es complejo el procesamiento interno del sistema?				x		
¿Se ha diseñado el código para ser reutilizable?					x	
¿Se ha diseñado el sistema para facilitar a los usuarios el trabajo y ayudarlos a encontrar la información?						x
TOTAL $\sum Fi$						38

Tabla 4.4: Ajuste de complejidad del punto función

Figura: [Elaboración propia]

Sin importancia	0
Incremental	1
Moderado	2
Medio	3
Significativo	4
Esencial	5

Tabla 4.5: Valores de ajuste de complejidad

Fuente: [Elaboración propia]

Con la obtención de los datos anteriores se considera un grado de confiabilidad del 83% calculemos el grado de punto función.

$$PF = CUENTA\ TOTAL * (Grado\ de\ confiabilidad + Tasa\ de\ error * \sum F).$$

$$PF = 1590 * (0.83 + 0.01 * 38)$$

$$PF = 1924$$

Hallamos el punto de valor máximo para comparar los valores del sistema.

$$PF_{max} = CUENTA\ TOTAL * (Grado\ de\ confiabilidad + Tasa\ de\ error * \sum F).$$

$$PF_{max} = 1590 * (0.83 + 0.01 * 55)$$

$$PF_{max} = 2194$$

Calcularemos el valor de ajuste da la complicación del punto función. Por lo tanto la funcionalidad real es:

$$FUNCIONALIDAD = 1590 * 100\% / 2194$$

$$FUNCIONALIDAD = 73\%$$

La funcionalidad estimada es de 73% tomando en cuenta el punto función máximo.

4.1.1.4. Confiabilidad

Aquí se agrupan un conjunto de atributos que se refieren a la capacidad del software de mantener su nivel de ejecución bajo condiciones normales en un periodo de tiempo establecido (Valle, 2009).

Se observa el trabajo hasta que se observa un fallo en un instante t, la función es la siguiente.

$$\text{Probabilidad de hallar una falla: } P(T \leq t) = F(t)$$

$$\text{Probabilidad de no hallar una falla: } P(T > t) = 1 - F(t) \lambda$$

$$\text{Conf}(t) = FC * e^{\lambda/6 * 12}$$

Donde:

FC = 0.73; Funcionalidad del sistema.

$\lambda = 1$; Tasa de fallos en 8 ejecuciones dentro de un mes.

Hallamos la confiabilidad del sistema:

$$F(t) = FC * e^{\lambda/6 * 12}$$

$$F(t) = 0.73 * e^{1/6 * 12}$$

$$F(t) = 0.103$$

La probabilidad de hallar una falla es de un 10% durante los próximos 12 meses.

$$P(T > t) = 1 - F(t)$$

$$P(T > t) = 1 - 0.103$$

$$P(T > t) = 0.897$$

Por lo tanto se determina que la probabilidad de no hallar una falla es del 90% durante los próximos 12 meses, por lo tanto es una aceptación confiable y aceptable de parte del sistema.

4.1.1.5. Portabilidad

La portabilidad se refiere a la habilidad del software de ser transferido de un ambiente a otro, se considera los siguientes aspectos.

- **Adaptabilidad.-** Aquí se evalúa la capacidad de adaptar el software a diferentes ambientes sin la necesidad de aplicarle modificaciones.
- **Facilidad de instalación.-** Es el esfuerzo necesario para instalar el software en un ambiente determinado.
- **Conformidad.-** Permite evaluar si el sistema se adhiere a estándares o convenciones relativas a portabilidad.
- **Capacidad de reemplazo.-** Hace referencia a la oportunidad y el esfuerzo usado en sustituir el software por otro producto con funciones similares.

Se puede medir en los siguientes niveles:

- a) **Nivel del software.-** El instalador del sistema puede ser distribuido en medios magnéticos, el software PHP, MySQL pueden ser instalados en los sistemas operativos Windows o Linux, por lo tanto el sistema es fácilmente portable, y se puede migrar la base de datos e información fácilmente o Oracle, XML, Acces, SQL Server.
- b) **Nivel del hardware.-** El sistema es portable y adaptable, si se cumple con las siguientes características:
 - CPU Pentium IV o superior.
 - RAM de 128 MB.
 - Disco Duro de 2 GB o superior.
 - Monitor, Lector CD-ROM, Teclado y Mouse.

4.1.2. Resultados

El factor de calidad total está directamente relacionado con el grado de satisfacción con el usuario que ingresa al sistema web de inventario, ventas y compras.

A continuación se muestra los resultados de la evaluación de calidad.

CARACTERÍSTICAS	RESULTADO
Usabilidad	0.92
Funcionalidad	0.73
Confiabilidad	0.90
Mantenibilidad	0.83
Evaluación de Calidad total	0.85

Tabla 4.6: Resultado de la evaluación de calidad

Fuente: [Elaboración propia]

El nivel de aceptabilidad satisfactorio, indica que los valores de preferencia se encuentran en el rango 60 – 100.

El nivel de aceptabilidad marginal, indica que los valores de preferencia se encuentran en el rango de 0 – 40.

Por lo tanto el factor de calidad total está directamente relacionado con el grado de satisfacción del usuario que ingresa al sistema web, la calidad total es **85%**.

4.2. Seguridad

Debido a que el sistema WEB desarrollado para la empresa contiene información representada por datos almacenados, estos son susceptibles a diferentes tipos a diferentes tipos de amenaza, como un mal uso de las contraseñas por parte de los usuarios, la corrupción de los datos por un sujeto externo e incluso fallas en el sistema eléctrico.

Es por tal motivo que en el sistema web desarrollado se implemento la parte de la seguridad en sus datos de las siguientes formas:

4.2.1. Autenticación

La seguridad en cuanto a la autenticación, se refiere al control de sesiones o verificación de la identificación, nombre de usuario y contraseña, previamente establecida en el sistema.

El framework CakePHP, utilizado en el desarrollo del sistema web, tiene la capacidad de encriptar de manera automática las contraseñas, permitiendo así un mejor control de la seguridad en cuanto a la autenticación de usuarios.

4.2.2. Logs o Registros

Los logs son registros de los eventos realizados en un sistema, permitiendo el seguimiento de estos eventos, que se clasifican principalmente de las acciones de un usuario como ser: ingresos al sistema con respecto al nombre y fecha, además se los intentos fallidos de ingreso al sistema como se muestran a continuación.

Ingreso_Correcto

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra	Acción
1	cod_usuario	int(11)			No	Ninguna		Cambiar Eliminar Primaria Único Índice Espacial Más
2	nombre	varchar(20)	utf8_general_ci		No	Ninguna		Cambiar Eliminar Primaria Único Índice Espacial Más
3	paterno	varchar(20)	utf8_general_ci		No	Ninguna		Cambiar Eliminar Primaria Único Índice Espacial Más
4	usuario	varchar(20)	utf8_general_ci		No	Ninguna		Cambiar Eliminar Primaria Único Índice Espacial Más
5	fecha	date			No	Ninguna		Cambiar Eliminar Primaria Único Índice Espacial Más
6	cod_trab	int(11)			No	Ninguna		Cambiar Eliminar Primaria Único Índice Espacial Más

Tabla 4.7: Registro de ingresos correctos

Fuente: [Elaboración propia]

Ingreso_Incorrecto

#	Nombre	Tipo	Cotejamiento	Atributos	Nulo	Predeterminado	Extra	Acción
1	usuario	varchar(20)	utf8_general_ci		No	Ninguna		Cambiar Eliminar Primaria Único Índice Espacial Más
2	contraseña	varchar(20)	utf8_general_ci		No	Ninguna		Cambiar Eliminar Primaria Único Índice Espacial Más
3	fecha	date			No	Ninguna		Cambiar Eliminar Primaria Único Índice Espacial Más

Tabla 4.8: Registro de ingresos incorrectos

Fuente: [Elaboración propia]

4.2.3. Base de Datos

Los datos son los activos más valiosos dentro de una empresa, por lo mismo es importante resguardar la seguridad de los mismos.

En cuanto a las formas de resguardar la seguridad de base de datos se puede nombrar: la conexión que se realiza a la misma al momento de obtener o recuperar datos. Esto contempla las conexiones a la base de datos y el cierre o finalización de la conexión de forma automática.

En cuanto a la amenaza SQL Injection que es una de las más comunes en las conexiones con las bases de datos, se implemento medidas de seguridad como la restricción de caracteres especiales en los campos de ingreso de texto. Los Backups o resguardos de la base de datos, que permiten mayor seguridad, se generan de manera automática teniendo así una copia de la información contenida en la base de datos para ser restaurada, la misma será realizada al finalizar el día.



CAPITULO V

ANALISIS DE COSTOS Y BENEFICIOS

5.1. COCOMO II

La planificación o estimación de costos es muy importante en todo proyecto, no solo de los requerimientos de hardware sino también de costos de tiempo y esfuerzo; COCOMO II, es un método de estimación de costos y esfuerzo de únicamente proyectos de software, que permite la estimación por medio de los módulos planificados en el software.

5.1.1. Punto función

Para obtener la estimación con los puntos función, COCOMO II, requiere la selección de lenguaje, en este caso “Orientado a Objetos”, ya que el framework CakePHP pertenece a este lenguaje y los datos que se muestran a continuación:

- Archivos lógicos internos, que corresponde al grupo de entradas o tablas.
- Archivos de interfaz externos, caratulas o fronts utilizados en el sistema.
- Insumos externos, datos de entrada al sistema, esos datos de entrada corresponden a grupos de entrada, no así a datos sueltos como nombre y edad.
- Salidas externas o reportes generados.
- Consultas externas.

A continuación se muestran los resultados que se obtienen de COCOMO II:

<i>ESTIMACION</i>	<i>ESFUERZO</i>	<i>SEMANAS</i>	<i>PRODUCTO</i>	<i>COSTO (\$)</i>	<i>PERSONAS</i>
<i>Optima</i>	35	11	331	21174	3
<i>Mas</i>	45	12	265	26428	4
<i>Aproximado</i>	55	13	221	33397	4

Tabla 5.1: Resultados de la estimación con COCOMO II

Fuente: [Elaboración propia]

La tabla que se muestra a continuación refleja los años de mantenimiento con respecto al costo de egreso e ingresos; el costo de egreso del año 1, corresponde al costo más aproximado obtenido en COCOMO II.

	<i>Año 1</i>	<i>Año2</i>	<i>Año3</i>	<i>Año 4</i>
<i>Egresos</i>	<i>26428</i>	<i>8005</i>	<i>5254</i>	<i>2484</i>
<i>Ingresos</i>	<i>-26428</i>	<i>6000</i>	<i>12000</i>	<i>20000</i>

Tabla 5.2: Estimación de mantenimiento

Fuente: [Elaboración propia]

El valor actual neto (VAN), se calcula por medio de los flujos de inversión, cuyo resultado refleja si la inversión en el proyecto generara beneficios si el resultado que se obtiene es positivo; este cálculo corresponde a la siguiente fórmula:

$$VAN = \sum Vt/(1+k)^t - lo$$

Donde:

Vt , representa los flujos de caja en cada periodo t.

Lo , es el valor de desembolso inicial de la inversión.

N , es el numero de periodos considerados.

k , es el interés.

Teniendo como valor del interés 15% entonces:

$$VAN = -26428 + 6000/(1-(1/15))^1 + 12000/(1-(1/5))^2 + 20000/(1-(1/5))^3$$

$$VAN = -26428 + 30072$$

$$VAN = 3644$$

Como el resultado del valor actual neto es positivo, indica que el proyecto es rentable.

A partir de los ingresos en los cuartos de mantenimiento, se obtiene el valor TIR = 16%.

Año de operación	Costos Totales (\$)	Beneficios totales	Factor de actualización 15%	Costos Actualizados (\$)	Beneficios Actualizados (\$)
1	8005	6000	0.870	6960.87	5217.39
2	5254	12000	0.756	3972.78	9073.72

3	2482	20000	0.658	1631.95	13150.32
Total	15741	38000		12565.60	27441.44

Tabla 5.3: Calculo relación de Costo Beneficio

Fuente: [Elaboración propia]

La tabla 5.3, permite obtener los ingresos y costos actualizados a partir de los cuales se desarrolla la siguiente formula, con la que se hallara la relación costo beneficio.

$$B/C = 27441.44$$

$$B/C = 12565.60$$

Por tanto se tiene como resultado de la relación costo beneficio: 2.18 es decir que por cada dos dólares invertidos se recupera 18 centavos.

5.3 Método de estimación de costo del sistema – CRI

Existen diversos métodos de estimación de costos, métodos tradicionales como ser el punto función, sin embargo la estimación de costos puede obtenerse mediante el método CRI, como describe Gomez (2013), este método toma en cuenta los siguientes elementos:

- Características de los campos de la base de datos (C).
- Relaciones entre tablas (R).
- Cantidad de vistas, consultas procedimientos almacenados, para visualizar datos en pantalla o en impresiones (I).

Para la estimación de costos por este método, se tiene la siguiente fórmula:

$$Costo = (UTc + UTr + UTi) * Valor monetario (Bs/h)$$

Donde:

UTc: Unidades de trabajo de los campos

$$UTc = \sum Complejidad del campo i$$

La complejidad de datos, se describe en la tabla 5.4.

Tipo de dato	Complejidad
VARCHAR	1
INT, DOUBLE	2
DATE, TIME	3

Tabla 5.4: Complejidad por tipo de dato

Fuente: [Método de estimación de costos del sistema, Lic Sergio Gomez]

UTr: Unidades de trabajo de las relaciones.

$$UTr = \sum Nro. \text{ de tablas de la entidad compuesta } m$$

UTi: Unidades de trabajo de los impresos.

$$UTi = \sum Nro. \text{ de tablas de reporte } r.$$

La tabla 5.5, que se muestra a continuación, refleja los resultados de la sumatoria de complejidad de datos obtenidos de las tablas utilizadas por el sistema.

Sumatoria de Complejidad de datos por tabla	
OBRA	6
REALIZA	12
CLIENTE	10
VENTA	17
PRODUCTO	6
FABRICA	10
ESTADO	3
CONTIENE	6
TRABAJADOR	5
HERRAMIENTA	7
USUARIO	19
PROVEEDOR	5
COMPRA	13
MATERIAL	10
TIPO_MATERIAL	5
UTILIZA	12

TOTAL	146
--------------	------------

Tabla 5.5: Sumatoria de complejidad de datos por tabla

Fuente: [Elaboración propia]

A continuación, la tabla 5.6, refleja los resultados de las relaciones existentes.

Relaciones	Entidades compuestas
Obra – Cliente	2
Cliente – Producto	2
Producto – Estado – Material - Cliente	4
Trabajador – Producto – Herramienta-Usuario	4
Material – Proveedor – Tipo_material	3
Total	15

Tabla 5.6: Unidades de trabajo de las relaciones

Fuente: [Elaboración propia]

A continuación, la tabla 5.7, refleja el número de reportes requeridos.

Reportes	Nro. de tablas en el reporte
Factura	3
Reporte final	4
Reporte de material	3
Reporte de herramientas	2
Reporte de E/S	4
Inventario	6

Total	22
-------	----

Tabla 5.7: Unidades de trabajo de las relaciones

Fuente: [Elaboración propia]

Concluyendo con la fórmula para la obtención de Unidades de trabajo dentro del sistema desarrollado se tiene:

$$UT = 146 + 15 + 22$$

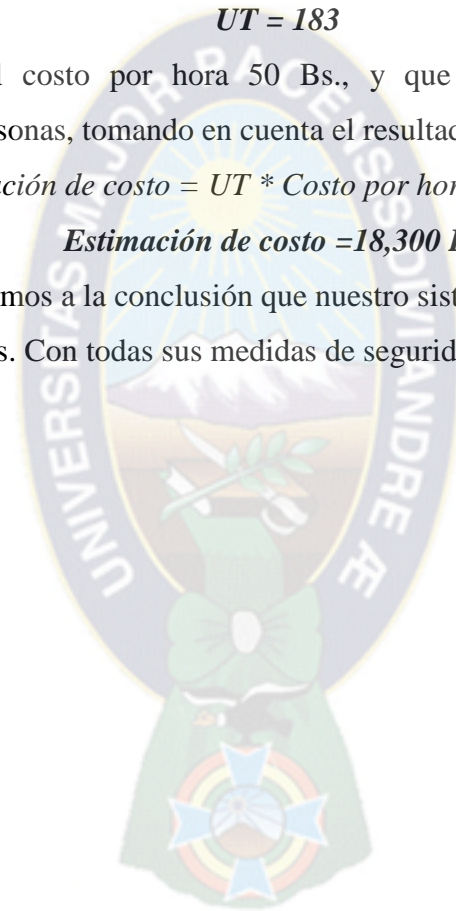
$$UT = 183$$

Considerando el costo por hora 50 Bs., y que el equipo de desarrollo este conformado por dos personas, tomando en cuenta el resultado más óptimo de COCOMO II:

$$\text{Estimación de costo} = UT * \text{Costo por hora} = 183 * 50 * 2$$

$$\text{Estimación de costo} = 18,300 \text{ Bs.}$$

Por lo cual llegamos a la conclusión que nuestro sistema tiene un costo de dieciocho mil trescientos. Con todas sus medidas de seguridad.



CAPITULO VI

CONCLUSIONES Y RECOMENDACIONES

6.1. Conclusiones

Habiendo cumplido los requerimientos establecidos por la empresa Técnica de Aluminio, Vidrio y Servicios (TALVISER), se ha logrado alcanzar al objetivo planteado por medio de la implementación de un sistema web a través de los cuatro módulos descritos como Modulo de Inventarios, Compras, Ventas y la generación de reportes.

Por tanto se llegan las siguientes conclusiones según a los Objetivos Específicos y General:

- Se logro un control de las entradas y salidas de materiales y herramientas en la empresa, haciendo este proceso automático todo esto para evitar los errores humanos en su cálculo y teniendo un historial del mismo, el administrador, puede ver el registro de entrada y salida, además de que con la aplicación móvil se puede realizar estos ingresos y salidas desde cualquier lugar de trabajo que se encuentra la empresa.
- Se logro un control de la venta de productos en la empresa hacia el cliente esto solamente dentro de la empresa.
- Se logro un control de inventarios donde ingresan los materiales y herramientas que se adquirieron de los proveedores.
- Se logro la generación de reportes de los inventarios ya que para realizar un reporte se tardaba alrededor de un día completo, ahora a través del sistema web se logra realizar el reporte en tan solo minutos.

6.2. Recomendaciones

Al haber concluido el presente proyecto de grado, se tiene las siguientes recomendaciones de proyectos similares en las que se pueden guiar para el desarrollo de sistemas similares extraídos de la biblioteca de la carrera de informática.

“Sistema Web de Comercio Electrónico y control de Inventario Caso: FIBBY’S SRL”, Israel Gabriel Bernabe Condorenz (2015). Universidad Mayor de San Andrés, Carrera de Informática, Desarrolla y diseña un sistema de control y seguimiento de inventarios y de comercio aplicando la metodología AUP y UWE.

El cual nos permite realizar una interpretación de la forma de desarrollo para realizar el diseño del sistema, así mismo se debe realizar un seguimiento al software para la mejora en distintos módulos.



BIBLIOGRAFIA

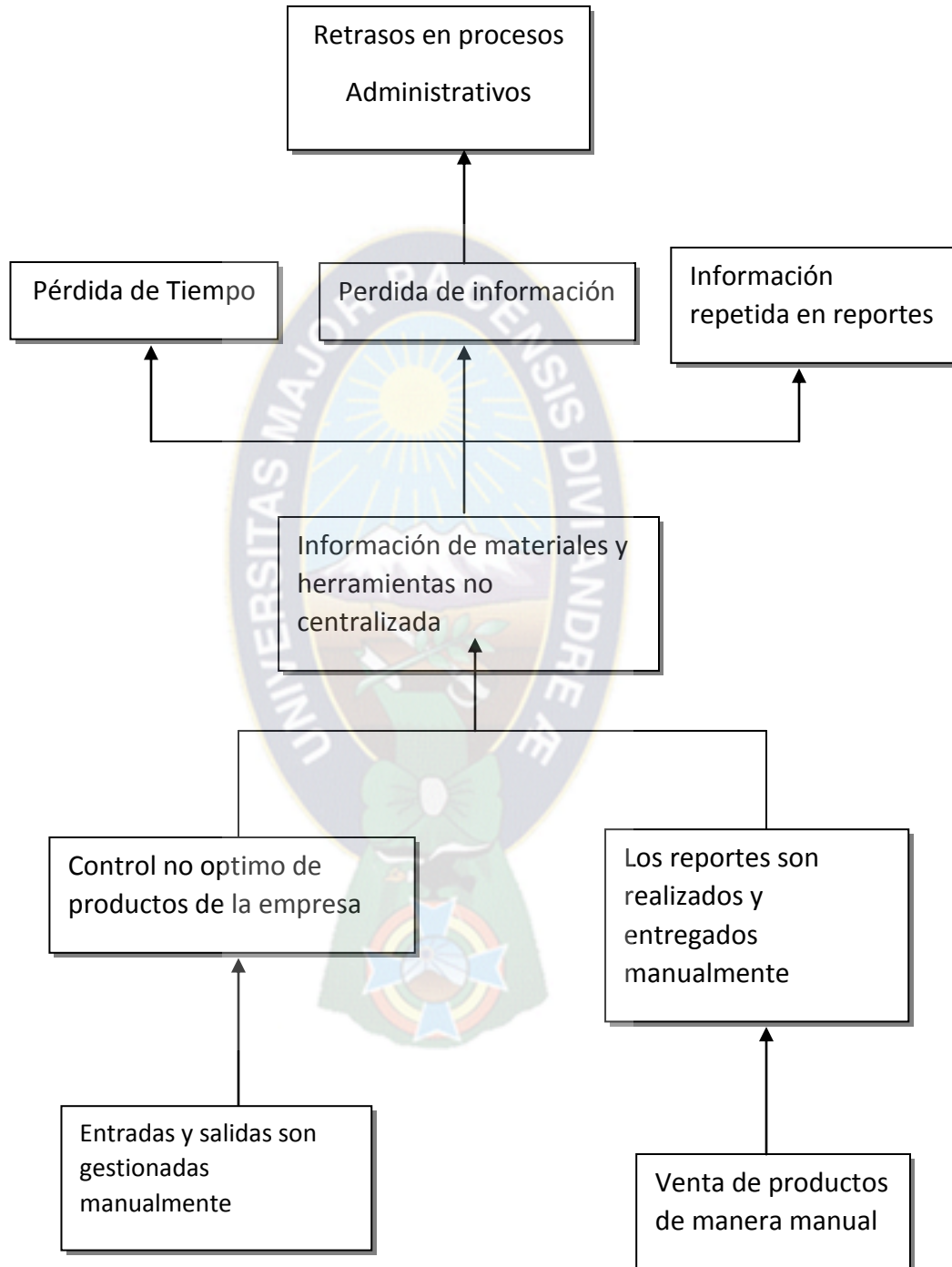
1. Condori, G.C. Sistemas de Distribución de ventas y control de inventarios. La Paz, 2007.
2. Huallpa, R.C. Sistema de control de ventas e inventarios para la red de Farmacias Virgen del Carmen. La Paz, 2009.
3. Ríos, H.R. Sistema de Información para el control de ventas e inventarios de Sucursales. La Paz, 2012.
4. Rumbaugh. Técnicas de Modelado de Objetos (OMT), 1997.
5. Ing. Joel Gonzales Estrada. Desarrollo Web con PHP y MySQL, (2005).
6. Jacobson/Booch/Rumbaugh. El lenguaje Unificado de Modelado Manual de Referencia, 2000.
7. Larman Carig. UML y patrones 1ra edición- México, 1999.
8. Roger S. Pressman. Ingeniería del Software un Enfoque Practico-Quinta Edición, 1998.
9. Platini Velthuis M. & García Rubí o F.J. Calidad en el Desarrollo y Mantenimiento del Software, 2003
10. Carla Hortensia Nadeau Inchausti. Sistema integrado de administración y gestión CREATRONIC SRL, 2007.

Referencias de Internet

1. Geoci, 2000 Disponible en:
<http://www.geocities.com/SiliconValley/2208/Insituacion.html>.
2. Isfti, 2000 Disponible en:
<http://www.isftic.mepsyd.es/w3/eos/MaterialesEducativos/mem2003/prg/gramacion/actividades/tecnicas.html>.
3. Juan C, 1995, Juan Cesar Martínez Disponible en:
<http://www.ci.ulsa.mx/~elinodocencia/ctrldesa/ISO14764.pdf>.
4. Thale,2007 Disponible en:
<http://thales.cica.es/rd/Recursos/rd98/Matematicas/29/mixto.html>.
5. PHP con MySQL. Ing. Cedric Simón Disponible en:
<http://www.solucionjava.com>.
6. Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia Disponible en:
<https://www.pid.dsic.upv.es>.
7. Norma de gestión de la calidad y garantía de la calidad, parte 3 Disponible en:
<http://www.alarcos.inf-cr.uclm.es>.

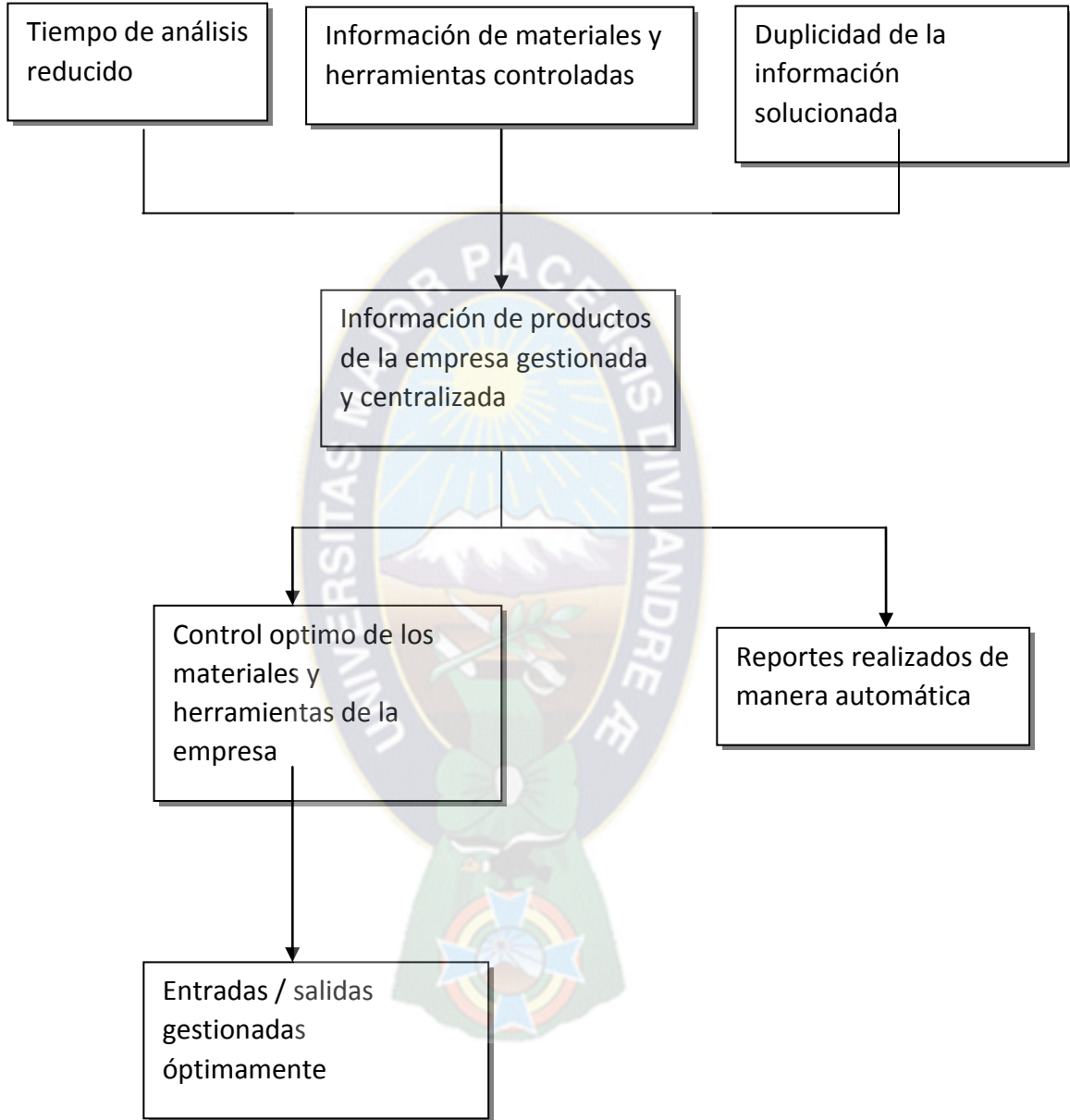
ANEXO A

ARBOL DE PROBLEMAS



ANEXO B

ARBOL DE OBJETIVOS



ANEXO C

MARCO LOGICO

JERARQUIA DE OBJETIVOS	METAS	INDICADORES	FUENTE DE VERIFICACION	SUPUESTOS
FIN	Sistema de Control de Ventas e Inventarios para Almacenes de Aluminios, mediante el uso de Dispositivos Móviles donde se maneja la información en constante actualización.	-Información centralizada en un 95%. -Reducción de errores de cálculos en un 70%.	-Reportes. -Encuestas.	
PROPOSITO	Automatizar procesos de la empresa y centralizar la información de productos y materiales.	-Reducción en la pérdida de información en un 90%.	-Tabla de resultados. -Reportes.	-Incompatibilidad con navegadores web y servidor. -Falta de recursos en la ejecución.
RESULTADOS	-Controlar las ventas de los productos. -Automatizar la entrada y salida de materiales y herramientas.	-Tiempo utilizado en los procesos de la empresa reducido en 60%. -Control de la información aumentado en un 90%.	-Diagramas. -Tabla de resultados. Reportes.	-Información de entradas y salidas y cálculos contables realizados de manera manual.
ACCIONES	-Análisis de los procesos. -Estructura de la Base de Datos. -Diseño del sistema web.	Tiempo de realización: -1semana. -1semana. -4semanas.	-Diagramas UML. -Modelo Entidad Relación, Físico y Clases	-Gestión de entradas y salidas de la información de productos. -Facturación automatizada.