

Universidad Mayor de San Andrés
Facultad de Ciencias Puras y Naturales
Carrera de Informática



PROYECTO DE GRADO

**“Sistema de Gestión Académica
para el Instituto Normal Superior Simón Bolívar”**

**PARA OPTAR AL TITULO DE LICENCIATURA EN INFORMATICA
MENCION INGENIERIA DE SISTEMAS**

Autor: Univ. Diego Omar Chambi Lima
Tutor: Lic. Mario Loayza Molina MSc.
Revisora: Lic. Celia Elena Tarquino Peralta
Asesora: Lic. Maribel Tania Ibáñez Reaza

LA PAZ – BOLIVIA
2007

INDICE GENERAL

CAPITULO I INTRODUCCION		Pag.
1.1.	INTRODUCCION	1
1.2.	ANTECEDENTES	3
1.3.	ANALISIS DE LA SITUACION ACTUAL	7
1.4.	PLANTEAMIENTO DEL PROBLEMA	8
1.5.	OBJETIVOS	9
1.6.	JUSTIFICACION	10
1.7.	LÍMITES Y ALCANCES	12
 CAPITULO II MARCO TEORICO		 Pag.
2.1.	INTRODUCCIÓN.....	16
2.2.	INGENIERÍA DEL SOFTWARE.....	17
2.3.	INGENIERÍA WEB.....	18
2.4.	ANALISIS Y DISEÑO ORIENTADO A OBJETOS.....	19
2.5.	MODELO DE DESARROLLO ITERATIVO E INCREMENTAL.....	21
2.6.	LENGUAJE DE MODELADO UNIFICADO, UML.....	24
2.7.	METODOLOGIA DE DESARROLLO.....	25
2.8.	ARQUITECTURA CLIENTE / SERVIDOR.....	59
2.9.	CALIDAD DEL SOFTWARE.....	60
2.10.	ESTIMACION DEL COSTO DEL PROYECTO.....	77
2.11.	HERRAMIENTAS DE CONSTRUCCIÓN DE SOFTWARE.....	81
 CAPITULO III DESARROLLO DEL SISTEMA		 Pag.
ANALISIS Y DISEÑO		
3.1.	INTRODUCCION.....	88
3.2.	PROCESO DE DESARROLLO.....	88
3.3.	FASE DE PLANEACIÓN Y ELABORACIÓN.....	94
3.4.	FASE DE CONSTRUCCIÓN.....	101
3.5.	IMPLEMENTACION.....	141
 CAPITULO IV CALIDAD DEL SISTEMA		 Pag.
4.1.	INTRODUCCION.....	145
4.2.	METODOLOGÍA DE EVALUACIÓN DE CALIDAD DE SITIOS WEB....	146
4.3.	BENEFICIO/COSTO	157

CAPITULO V CONCLUSIONES Y RECOMENDACIONES Pag.

5.1. CONCLUSIONES..... 168
5.2. RECOMENDACIONES..... 170

BIBLIOGRAFIA..... 171

ANEXOS

- A ARBOL DE PROBLEMAS Y OBJETIVOS**
- B CASOS DE USO**
- C DOCUMENTOS**



INDICE ESPECÍFICO

CAPITULO I INTRODUCCION

	Pag.
1.1. INTRODUCCION	1
1.2. ANTECEDENTES	3
1.2.1. ENTORNO INSTITUCIONAL	3
1.2.2. ANTECEDENTES DE PROYECTOS SIMILARES	4
1.3. ANALISIS DE LA SITUACION ACTUAL	7
1.4. PLANTEAMIENTO DEL PROBLEMA	8
1.4.1. FORMULACION DEL PROBLEMA	8
1.5. OBJETIVOS	9
1.5.1. OBJETIVO GENERAL	9
1.5.2. OBJETIVOS ESPECIFICOS	9
1.6. JUSTIFICACION	10
1.6.1. JUSTIFICACION TECNICA	10
1.6.2. JUSTIFICACION ECONÓMICA	11
1.6.3. JUSTIFICACION SOCIAL	11
1.7. LÍMITES Y ALCANCES	12

CAPITULO II MARCO TEORICO

Pag.

2.1.	INTRODUCCIÓN.....	16
2.2.	INGENIERÍA DEL SOFTWARE.....	1
2.3.	INGENIERÍA WEB.....	1
2.4.	ANÁLISIS Y DISEÑO ORIENTADO A OBJETOS.....	1
2.4.1.	ANÁLISIS ORIENTADO A OBJETOS.....	1
2.4.2	DISEÑO ORIENTADO A OBJETOS.....	2
2.4.3.	CONSTRUCCION.....	2
2.5.	MODELO DE DESARROLLO ITERATIVO E INCREMENTAL.....	2
2.5.1.	MODELO ITERATIVO.....	2
2.5.2.	MODELO INCREMENTAL.....	2
2.6.	LENGUAJE DE MODELADO UNIFICADO, UML.....	2
2.7.	METODOLOGIA DE DESARROLLO.....	2
2.7.1.	PASOS DE MACRONIVEL.....	2
2.7.2.	FASE DE PLANEACIÓN Y ELABORACIÓN	2
2.7.2.1.	ESPECIFICACIÓN DE REQUERIMIENTOS.....	2
2.7.2.2.	CASOS DE USO: DESCRIPCIÓN DE PROCESOS.....	3
2.7.2.2.1.	CASOS DE USO.....	3
2.7.2.2.2.	ACTORES.....	3
2.7.2.2.3.	FORMATO DE LOS CASOS DE USO.....	3
2.7.2.2.3.1.	Casos de uso de alto nivel.	3
2.7.2.2.3.2.	Casos de uso con formato expandido.....	3
2.7.2.3.	DIAGRAMAS DE LOS CASOS DE USO.....	3
2.7.3.	FASE DE CONSTRUCCIÓN.....	3
2.7.3.1.	ETAPA DE ANÁLISIS.....	3
2.7.3.1.1.	MODELO CONCEPTUAL.....	3
2.7.3.1.2.	REGISTRO DE LOS TÉRMINOS DEL GLOSARIO.....	4
2.7.3.1.3.	DIAGRAMAS DE SECUENCIA DEL SISTEMA.....	4
2.7.3.1.4.	CONTRATOS DE OPERACIONES.....	4
2.7.3.2.	ETAPA DE DISEÑO.....	4
2.7.3.2.1.	DESCRIPCIÓN DE LOS CASOS DE USO REALES.....	4
2.7.3.2.2.	DEFINIR DIAGRAMAS DE INTERACCIÓN.....	4
2.7.3.2.2.1.	DIAGRAMAS DE COLABORACION.....	4
2.7.3.2.3.	DIAGRAMAS DE CLASES.....	5
2.7.3.2.3.1.	CLASE.....	5
2.7.3.2.3.1.1	ATRIBUTOS.....	5
2.7.3.2.3.1.2	OPERACIONES	5
2.7.3.2.3.2.	ASOCIACION DE CLASE	5
2.7.3.2.3.2.1	GENERALIZACION	5
2.7.3.2.3.2.2	ASOCIACIONES	5
2.7.3.2.3.2.3	MULTIPLICIDAD	5
2.7.3.2.4.	DIAGRAMAS DE ESTADO.....	5
2.7.3.2.4.1.	GENERALIZACION DE ESTADOS.....	5
2.7.4.	FASE DE APLICACIÓN.....	5
2.7.4.1.	PRUEBAS.....	5
2.8.	ARQUITECTURA CLIENTE / SERVIDOR.....	5
2.9.	CALIDAD DEL SOFTWARE.....	6
2.9.1.	METODOLOGÍA WEB-SITE QEM.....	6
2.9.1.1.	DEFINIENDO EL DOMINIO Y ENTE PARA LA EVALUACION DE LA CALIDAD.....	6

2.9.1.2.	DEFINIENDO METAS DE EVALUACION Y SELECCIONANDO EL PERFIL DE USUARIO.....	64
2.9.1.3.	ESPECIFICANDO REQUERIMIENTOS DE CALIDAD PARA ARTEFACTOS WEB.....	65
2.9.1.3.1	ENFOQUES DE MODELO DE CALIDAD.....	66
2.9.1.4.	DEFINIENDO CRITERIOS ELEMENTALES E IMPLEMENTANDO PROCEDIMIENTOS DE MEDICION.....	66
2.9.1.4.1.	REPRESENTACION NOTACIONAL DE LOS CRITERIOS.....	67
2.9.1.4.2.	TIPOS DE CRITERIOS DE PREFERENCIA DE CALIDAD ELEMENTAL.....	68
2.9.1.5.	DEFINIENDO LAS ESTRUCTURAS DE AGREGACION E IMPLEMENTANDO LA EVALUACION DE CALIDAD GLOBAL.....	71
2.9.1.5.1.	GRADOS DE CONJUNCION Y DISYUNCION.....	72
2.9.1.5.2.	EMPLEO DE LOS OPERADORES DE LSPPARA MODELAR RELACIONES LOGICAS ENTRE CARACTERISTICAS Y ATRIBUTOS.....	73
2.9.1.5.3.	TIPOS DE FUNCIONES DE AGREGACION.....	75
2.9.1.5.4.	CONSIDERACIONES SOBRE LAS ACTIVIDADES EN EL PROCESO DE AGREGACION.....	76
2.9.1.6.	ANALIZANDO Y COMPARANDO LOS RESULTADOS PARCIALES Y GLOBALES.....	76
2.10.	ESTIMACION DEL COSTO DEL PROYECTO.....	77
2.10.1.	EL MÉTODO COCOMO II.....	78
2.11.	HERRAMIENTAS DE CONSTRUCCIÓN DE SOFTWARE.....	81
2.11.1.	PHP.....	81
2.11.2.	MYSQL.....	83
2.11.3.	SERVIDOR HTTP APACHE.....	85
2.11.4.	INSTALACION DE PHP5 CON WAMP5.....	86



CAPITULO III DESARROLLO DEL SISTEMA ANÁLISIS Y DISEÑO

	Pag.	
3.1.	INTRODUCCION.....	88
3.2.	PROCESO DE DESARROLLO.....	88
3.2.1.	ANÁLISIS ACTUAL DEL INSSB.....	89
3.2.1.1.	ESTRUCTURA ORGANIZACIONAL.....	90
3.2.1.2.	PROCESOS ACADEMICOS GENERALES.....	92
3.3.	FASE DE PLANEACIÓN Y ELABORACIÓN.....	94
3.3.1.	INFORME PRELIMINAR.....	94
3.3.2.	ESPECIFICACIÓN DE REQUERIMIENTOS.....	94
3.3.2.1.	PANORAMA GENERAL.....	94
3.3.2.2.	CLIENTES.....	95
3.3.2.3.	METAS.....	98
3.3.2.4.	FUNCIONES DEL SISTEMA.....	99
3.3.2.4.1.	FUNCIONES BÁSICAS.....	99
3.3.2.5.	ATRIBUTOS DEL SISTEMA.....	100
3.4.	FASE DE CONSTRUCCIÓN.....	101
3.4.1.	ETAPA DE ANÁLISIS DEL SISTEMA	101
3.4.1.1.	CASOS DE USO.....	101
3.4.1.2.	DIAGRAMA DE CASO DE USO	102
3.4.1.3.	MODELO CONCEPTUAL	108
3.4.1.4.	TERMINOS EN EL GLOSARIO	109
3.4.1.5.	DIAGRAMAS DE SECUENCIA	111
3.4.1.6.	CONTRATOS DE OPERACIONES	115
3.4.2.	ETAPA DE DISEÑO Y CONSTRUCCIÓN.....	120
3.4.2.1.	CASOS REALES DE USO.....	120
3.4.2.2.	DIAGRAMAS DE INTERACCION.....	131
3.4.2.2.1.	DIAGRAMAS DE COLABORACION.....	131
3.4.2.3.	DIAGRAMA DE CLASES.....	134
3.4.2.4.	DIAGRAMA DE ESTADO.....	135
3.4.3.	DISEÑO DE LA INTERFAZ DE USUARIO.....	137
3.5.	IMPLEMENTACION.....	141
3.5.1.	HERRAMIENTAS PARA EL DESARROLLO.....	141
3.5.1.1.	PLATAFORMA DEL SISTEMA.....	141
3.5.1.2.	SISTEMA GESTOR DE BASE DE DATOS.....	141
3.5.1.3.	HERRAMIENTA DE PROGRAMACION.....	141
3.5.1.4.	HERRAMIENTA DE DISEÑO.....	141
3.5.2.	INSTALACION.....	142
3.5.3.	PORTABILIDAD.....	143
3.5.3.1.	SERVIDOR.....	143
3.5.3.1.	CLIENTE.....	143
3.5.4.	ESTRUCTURA OPERACIONAL DEL SISTEMA.....	144

CAPITULO IV CALIDAD DEL SISTEMA

	Pag.
4.1. INTRODUCCION.....	145
4.2. METODOLOGÍA DE EVALUACIÓN DE CALIDAD DE SITIOS WEB.....	146
4.2.1. DEFINIENDO EL DOMINIO Y ENTE PARA LA EVALUACIÓN DE LA CALIDAD	147
4.2.2. DEFINIENDO METAS DE EVALUACIÓN Y SELECCIONANDO EL PERFIL DE USUARIO.....	147
4.2.2.1. METAS DE EVALUACION.....	147
4.2.2.2. PERFILES DE USUARIO CONSIDERANDO LAS METAS Y EL DOMINIO DE APLICACIÓN EN LA WEB.....	148
4.2.3. ESPECIFICANDO REQUERIMIENTOS DE CALIDAD PARA ARTEFACTOS WEB.....	149
4.2.3.1. MODELOS DE CALIDAD DE PRODUCTO PRESCRITOS EN LOS ESTÁNDARES ISO 9126 E IEEE 1061.....	149
4.2.3.2. SELECCIONANDO EL ENFOQUE DE MODELO DE CALIDAD.....	149
4.2.3.3. ÁRBOL DE REQUERIMIENTOS DE CALIDAD GENERAL	150
4.2.4. DEFINIENDO LOS CRITERIOS DE EVALUACIÓN ELEMENTALES E IMPLEMENTANDO PROCEDIMIENTOS DE MEDICIÓN.	152
4.2.5. DEFINIENDO LAS ESTRUCTURAS DE AGREGACIÓN E IMPLEMENTANDO LA EVALUACIÓN DE CALIDAD GLOBAL.....	153
4.2.5.1. EMPLEO DE LOS OPERADORES DE LSP PARA MODELAR RELACIONES LOGICAS ENTRE CARACTERÍSTICAS Y ATRIBUTOS..	153
4.2.5.2. COMPUTO DE LAS PREFERENCIAS PARCIALES Y GLOBALES	154
4.2.6. ANÁLISIS Y COMPARACIÓN DE RESULTADOS PARCIALES Y GLOBALES.....	156
4.3. BENEFICIO/COSTO	157
4.3.1 ANÁLISIS DE COSTOS.....	157
4.3.1.1 COSTO DEL SOFTWARE DESARROLLADO.....	157
4.3.1.1.1 ESTIMACIÓN DE PUNTOS DE FUNCIÓN	158
4.3.1.2 COSTO DE IMPLEMENTACIÓN DEL SISTEMA	161
4.3.1.3 COSTO DE ELABORACIÓN DEL PROYECTO	163
4.3.1.4 COSTO TOTAL	163
4.3.2 ANÁLISIS DE BENEFICIOS	164
4.3.2.1 INCREMENTO DE VELOCIDAD EN LOS PROCESOS	164
4.3.2.2 CAPACIDAD EN EL VOLUMEN DE INFORMACIÓN	164
4.3.2.3 CONTROL DE PROCESOS.....	165
4.3.2.4 INTEGRACIÓN DE LA INFORMACIÓN.....	165
4.3.2.5 INFORMACIÓN PARA LA TOMA DE DECISIONES.....	165

CAPITULO V CONCLUSIONES Y RECOMENDACIONES	Pag.
5.1. CONCLUSIONES.....	168
5.2. RECOMENDACIONES.....	170
BIBLIOGRAFIA.....	171

ANEXOS

- A ARBOL DE PROBLEMAS Y OBJETIVOS**
- A1. ARBOL DE PROBLEMAS**
- A2. ARBOL DE OBJETIVOS**
- A3. MATRIZ DE PLANIFICACIÓN DEL PROYECTO**

- B CASOS DE USO**

- C DOCUMENTOS**



INDICE DE FIGURAS

Figura		Pag.
2.1.	Estratos de la ingeniería de software.....	17
2.2.	Significado de las actividades de desarrollo.....	19
2.3.	Modelo Iterativo.....	21
2.4.	Ciclos de desarrollo iterativo.....	22
2.5.	Pasos de macronivel en el desarrollo.....	25
2.6.	Actividades de la Fase de Planeación y Elaboración.....	26
2.7.	Actividades de dependencia de la Fase de Planeación y Elaboración	28
2.8.	Icono del UML que representa un Caso de Uso.....	31
2.9.	Representación de un Actor.....	32
2.10.	Diagrama de casos de uso.....	34
2.11.	Ciclo de Desarrollo.....	35
2.12.	Actividades de la etapa de Análisis dentro un ciclo de desarrollo.....	36
2.13.	Dependencias de los artefactos durante la Fase de Construcción.....	37
2.14.	Modelo conceptual parcial, muestra conceptos del mundo real.....	38
2.15.	Multiplicidad en una asociación.....	39
2.16.	Valores de multiplicidad.....	40
2.17.	Formato de Diagrama de Secuencia.....	42
2.18.	Actividades de la Etapa de Diseño dentro un ciclo de desarrollo.....	44
2.19.	Ventana.....	46
2.20.	Diagrama de Colaboración.....	48
2.21.	Diagrama de Colaboración: Mensajes Condicionales.....	49
2.22.	Relación entre los artefactos.....	50
2.23.	Diagrama de clases del diseño.....	51
2.24.	Representación visual de una clase en UML.....	52
2.25.	Representación visual de una generalización en UML.....	53
2.26.	Representación visual de una asociación en UML.....	54
2.27.	Diagrama de estado para un teléfono.....	55
2.28.	Actividades de la Fase de Aplicación.....	57
2.29.	Árbol de Requerimientos de Calidad.....	62
2.30.	Un panorama de los principales módulos intervinientes en el proceso de evaluación y comparación usando Web-site QEM.....	63
2.31.	Notación de Escala de Preferencia.....	68
2.32.	Taxonomía de tipos de criterio elementales.....	68
2.33.	Panorama del proceso de determinación de la preferencia de calidad global a partir de preferencias elementales.....	71
2.34.	Operadores lógicos Conjuntivos y Disyuntivos de LSP y niveles de Polarización.....	74
2.35.	Tres funciones simples de agregación de preferencias.....	75
3.1.	Especialidades por carrera que ofrece el INSSB.....	89
3.2.	Organigrama del Instituto Normal Superior Simón Bolívar.....	90
3.3.	Diagrama de Casos de Uso del Sistema.....	102

3.4.	Diagrama de Caso de uso: Registro de estudiantes nuevos.....	103
3.5.	Diagrama de Caso de uso: Matriculación.....	104
3.6.	Diagrama de Caso de uso: Inscripción.....	105
3.7.	Diagrama de Caso de uso: Registro de Notas.....	106
3.8.	Diagrama de Caso de uso: Administración del Sistema.....	107
3.9.	Diagrama Modelo Conceptual Preliminar.....	108
3.10.	Secuencia correspondiente al control de acceso del usuario.....	111
3.11.	Secuencia correspondiente al registro de estudiantes.....	112
3.12.	Secuencia correspondiente a la Matriculación.....	113
3.13.	Secuencia correspondiente a la Inscripción.....	113
3.14.	Secuencia correspondiente al Registro de Notas.....	114
3.15.	Ventana Formulario de registro de estudiantes nuevos.....	121
3.16.	Ventana Formulario de Matriculación.....	123
3.17.	Ventana Formulario de Inscripción.....	124
3.18.	Ventana Materias a registrar notas.....	126
3.19.	Ventana Registro de notas.....	126
3.20.	Ventana Formulario Administración de Usuarios.....	128
3.21.	Ventana administración de paralelos.....	129
3.22.	Relación entre los artefactos.....	131
3.23.	Diagrama de colaboración Registro de Estudiantes.....	131
3.24.	Diagrama de colaboración Matriculación de Estudiantes.....	132
3.25.	Diagrama de colaboración Inscripción de Estudiantes.....	132
3.26.	Diagrama de colaboración Adición de Usuarios.....	133
3.27.	Diagrama de colaboración para la Autenticación.....	133
3.28.	Diagrama de Clases del Sistema.....	134
3.29.	Diagrama de Estado correspondiente a la autenticación e ingreso al sistema.....	135
3.30.	Diagrama de Estado para el registro de estudiantes nuevos.....	135
3.31.	Diagrama de Estado para la matriculación de estudiantes.....	136
3.32.	Diagrama de Estado correspondiente a la Inscripción.....	136
3.33.	Ventana principal de ingreso al sistema.....	137
3.34.	Ventana de autenticación incorrecta.....	138
3.35.	Ventana con menú para el proceso de registro.....	139
3.36.	Diagrama de Estado para el registro de estudiantes nuevos.....	139
3.37.	Ventana de Inscripción de estudiante.....	140
3.38.	Ventana salir del sistema (finalización de Sesión).....	140
3.39.	Estructura Operacional del Sistema.....	144
4.1.	Operadores de LSP.....	153

INDICE DE TABLAS

Tabla	Pag.
2.1. Categoría de las funciones.....	29
2.2. Atributos de un sistema.....	30
2.3. Estructura de casos de alto nivel.....	32
2.4. Estructura de casos de uso expandido.....	33
2.5. Formato del glosario.....	41
2.6. Estructura de un Contrato.....	43
2.7. Casos de Uso Reales.....	45
2.8. Curso normal de los eventos.....	46
2.9. Plantilla para especificar a una característica de más alto nivel	69
2.10. Plantilla para especificar a un atributo.....	70
2.11. Plantilla para especificar a una subcaracterística.....	70
2.12. Categoría de un proyecto en función de sus líneas de código.....	77
2.13. Coeficientes COCOMO	79
2.14. Conversión Líneas de código a Puntos de función.....	80
3.1. Funciones Básicas.....	99
3.2. Tabla de Atributos del Sistema.....	100
3.3. Actores Identificados.....	101
3.4. Caso de uso: Registro de estudiantes nuevos.....	103
3.5. Caso de uso: Matriculación.....	104
3.6. Caso de uso: Inscripción.....	105
3.7. Caso de uso: Registro de Notas.....	106
3.8. Caso de uso: Administración del Sistema.....	107
3.9. Glosario.....	110
3.10. Contrato correspondiente al caso de uso: Verificaron de usuario....	115
3.11. Contrato correspondiente al caso de uso: Registro de estudiantes nuevos.....	116
3.12. Contrato correspondiente al caso de uso: Matriculación.....	117
3.13. Contrato correspondiente al caso de uso: Inscripción.....	118
3.14. Contrato correspondiente al caso de uso: Registro de Notas.....	119
3.15. Casos de Uso Reales: Registro de Estudiantes.....	120
3.16. Casos de Uso Reales: Matriculación de Estudiantes.....	122
3.17. Casos de Uso Reales: Inscripción de Estudiantes.....	124
3.18. Casos de Uso Reales: Registro de notas.....	125
3.19. Casos de Uso Reales: Administración de Usuarios.....	127
3.20. Casos de Uso Reales: Administración de Paralelos.....	129
4.1. Computo de las preferencias parciales.....	155
4.2. Computo de las preferencias globales.....	156
4.3. Cálculo de Puntos de Función no ajustados.....	158
4.4. Determinación de Complejidad.....	158
4.5. Conversión de Puntos de Función a KLDC.....	159
4.6. Coeficientes a_b y c_b y los exponentes b_b y d_b	160
4.7. Costo de Software Propietario.....	161
4.8. Costo de Hardware requerido.....	162

4.9.	Costo de implementación del Sistema.....	162
4.10.	Costo de elaboración del proyecto.....	163
4.11.	Costo Total.....	163
4.12	Cuadro de Beneficios.....	167



RESUMEN

En esta nueva era de revolución de las tecnologías de información, los sistemas que realizan el tratamiento de información se van haciendo más necesarios e imprescindibles para toda Institución, es así que Instituciones dedicadas al área de la educación superior, manejan cada vez más grandes volúmenes de información, tal es el caso del Instituto Normal Superior Simón Bolívar, encargado de la formación de Maestros, que debe adecuarse a estas nuevas tecnologías para mejorar sus servicios.

En las últimas gestiones la cantidad de estudiantes del INSSB ha tenido un crecimiento considerable, esto debido a la gran demanda que esta Institución tiene, actualmente el INSSB no cuenta con un Sistema de Información que ayude a realizar procesos fundamentales del área académica como ser: Registro de Estudiantes Nuevos, Matriculación, Inscripción, Registro y Control de notas, Seguimiento Académico, Emisión de Documentos de estudiantes, etc., estos procesos son realizados en forma manual produciendo demora en la realización de las diferentes labores que realiza el personal administrativo del Área Académica, sin embargo en el presente proyecto se pretende afrontar los inconvenientes anteriormente mencionados implementando el Sistema de Gestión Académica para el Instituto Normal Superior Simón Bolívar, que coadyuve en los procesos del área académica, para proporcionar información confiable, segura y oportuna, disminuyendo el trabajo excesivo del personal administrativo, beneficiando a la Institución, realzando su prestigio y proporcionando credibilidad en el manejo de la información de sus estudiantes, también beneficiará al personal administrativo, ayudando en los procesos del área académica, reduciendo el trabajo excesivo y moroso, para una mejor atención a la comunidad normalista y usuarios en general.

CAPÍTULO I



INTRODUCCIÓN

CAPÍTULO I

INTRODUCCIÓN

1.1. INTRODUCCIÓN

La World Wide Web y la Internet que la alimentan, son posiblemente los desarrollos más importantes en la historia de la computación. Estas tecnologías han llevado a todos (con cientos de millones mas que eventualmente seguirán) a la era de la informática; además, se han convertido en parte integral de la vida diaria en la primera década del siglo XXI. [PRESSMAN, 2005]

En esta nueva era de revolución de las tecnologías de información, los sistemas que realizan el tratamiento de información se van haciendo más necesarios e imprescindibles, más aún si estos están interconectados vía intranet o Internet, ya que facilitan la comunicación y la integración de los usuarios, brindando la capacidad de compartir dinámicamente los recursos, disminuyendo el trabajo redundante y optimizando el flujo de la información.

Hoy en día la información es un recurso muy importante para toda Institución que trabaja con cantidades de datos enormes; es así que Instituciones dedicadas al área de la enseñanza, manejan cada vez más grandes volúmenes de información, ya que la demanda de alumnos que ingresan a estas instituciones va creciendo continuamente, tal es el caso del Instituto Normal Superior Simón Bolívar, encargado de la formación de Maestros, que actualmente cuenta con aproximadamente 3000 estudiantes, en los Niveles de Inicial, Primario, Secundario y Primario Secundario, este ultimo se instituyó en la presente gestión;

la información de todas estas personas es manipulada y procesada en forma manual, tanto en el registro de estudiantes nuevos, matriculación e inscripción de estudiantes regulares, registro y control de notas y la emisión de documentos (hoja de ingreso personal, matricula, boleta de inscripción, certificado de notas, historial de notas, globales, etc.) que son realizadas al inicio y culminación de cada gestión. Al ser un Instituto de Educación Superior, los procesos del área académica, deben funcionar eficientemente, para asegurar el prestigio y el éxito de dicha entidad, cabe resaltar que las inscripciones y matriculaciones son semestrales, por lo que cada gestión se tropieza con procesos manuales que retardan el proceso de inscripción y matriculación de estudiantes, ocasionando grandes filas y mucho trabajo para el personal administrativo. Por otra parte los documentos que se exigen producen una gran cantidad de papeles que sencillamente se podrían suprimir sistematizando y almacenando esta información en una Base de Datos centralizada, que proporcione información eficiente, precisa y en el menor tiempo posible.

En el presente proyecto se pretende afrontar los inconvenientes anteriormente mencionados implementando el "Sistema de Gestión Académica para el Instituto Normal Superior Simón Bolívar", que coadyuve en los procesos del área académica, para proporcionar información eficiente, segura y oportuna, disminuyendo así el trabajo excesivo del personal administrativo del área académica.

1.2. ANTECEDENTES

1.2.1. ENTORNO INSTITUCIONAL

El Instituto Normal Superior Simón Bolívar (INSSB), fundado en fecha 24 de mayo de 1917, es una persona jurídica de derecho público y una institución nacional de estudios superiores para la educación dependiente del Ministerio de Educación, Cultura y Deportes, [LOZADA, 2003], según la entrevistas realizadas en un principio se creó como Facultad de Pedagogía bajo la dependencia de la Universidad Mayor de San Andrés, en ese entonces el Rector fue Don Claudio Sanjinez; en ese entonces contaba con poco alumnado, hasta hace poco la población estudiantil se ha ido incrementando, en la actualidad cuenta con aproximadamente 3000 estudiantes, en las diferentes Especialidades de sus cuatro Niveles.

La Unidad de Archivo y Kardex, según el Manual de Funciones, esta encargada de organizar y controlar el Archivo y Kardex a fin de dar celeridad y fluidez a los diferentes requerimientos de parte de los estudiantes, registra y archiva los datos de las actas oficiales de cada gestión académica, planifica las inscripciones de alumnos nuevos y antiguos, atiende a los estudiantes en la recepción de consultas académicas. [LOZADA, 2003]. Sin embargo, actualmente el INSSB no cuenta con un Sistema de Información que ayude a realizar procesos fundamentales como ser: Registro de Estudiantes Nuevos, Matriculación, Inscripción, Registro y Control de notas, Emisión de Documentos de estudiantes, etc. todas estas tareas son realizadas en forma manual por el personal de esta Unidad.



1.2.2. ANTECEDENTES DE PROYECTOS SIMILARES

Muchos postulantes realizaron Proyectos de Grado similares al que se esta desarrollando, sin embargo cada una presenta características particulares y tienen una aplicación específica ya que fueron desarrollados según los requerimientos de una Institución particular. A continuación se describen los siguientes proyectos:

A. SISTEMAS DESARROLLADOS PARA UNIVERSIDADES BOLIVIANAS

"SISTEMA DE SEGUIMIENTO ACADÉMICO UNIVERSIDAD TECNOLÓGICA BOLIVIANA (U T B)", del Postulante: Nelson Egberto Tarqui Carpio, cuyo objetivo fue analizar, diseñar, desarrollar e implementar un Sistema de Seguimiento Académico para la U.T.B. que facilite el manejo eficiente y oportuno de la información, para esto desarrollo los módulos de Registro de alumnos nuevos, Inscripciones, Registro y control de notas, Emisión de Historial Académico, Emisión de Certificado de Notas, Registro de Egresados y Reportes. La metodología que planteó para el análisis y desarrollo del Sistema es el Análisis y Diseño Estructurado Moderno de Yourdon, para la programación hizo uso de herramientas visuales.[TARQUI, 2003]

B. SISTEMAS DESARROLLADOS PARA FACULTADES Y CARRERAS DE UNIVERSIDADES

"SISTEMA DE GESTION ACADÉMICA UNIVERSITARIA CASO: FACULTAD DE INGENIERIA", de la Postulante: Rocio de la Azucena Sirpa Caceres, cuyo objetivo fue mejorar la gestión de la información en los procesos académicos de la Facultad de Ingeniería, adaptando, desarrollando e implementando el Sistema de Gestión Académica Universitaria en las carreras de Ingeniería Eléctrica, Electromecánica mención Eléctrica, Metalurgia y de Materiales, Civil, Petrolera y en el departamento de Cursos Básicos. Para tal efecto desarrollo los módulos de *Admisión y Matriculación, Planificación de la Gestión Académica, Inscripción, Seguimiento Académico y el acceso seguro, confidencial y restringido a cada usuario*. Adoptó la Metodología del Proceso Unificado de Desarrollo de Software, utilizando el Lenguaje Unificado de Modelado para preparar todos los esquemas

de un sistema software. Esta desarrollado en el lenguaje de programación JAVA, su Gestor de Base de Datos es PostgreSQL, y opera en servidores LINUX. [SIRPA, 2005].

C. SISTEMA DESARROLLADO PARA LA CARRERA DE INFORMÁTICA

"SISTEMA DE INFORMACIÓN Y GESTIÓN ACADÉMICA SIGA", del Lic. Mg.Sc.Franz Cuevas Quiroz, desarrollado en el Instituto de Investigación de Informática, el cual realiza el registro de estudiantes nuevos, inscripciones con el control de prerequisites, registro de notas. Emite reportes de: historial académico, materias a cursar, materias inscritas y número aleatorio para inscripciones. Cabe mencionar que este sistema estuvo en funcionamiento en la carrera de Informática.

D. PROYECTOS SIMILARES REALIZADOS EN LA CARRERA DE INFORMÁTICA

"CONTROL Y SEGUIMIENTO ACADÉMICO WEB PARA LA EMPRESA EDUCOMSER SRL" del Postulante: Freddy Flores Condori, cuyo objetivo fue diseñar y desarrollar un Sistema Académico para mejorar el proceso de inscripciones y control académico de participantes y docentes para así brindar información oportuna y confiable a la parte Académica, para tal efecto desarrolló los módulos de *Inscripciones, Calificaciones, Cronograma, Seguridad, Emisión de Informes*. La Metodología para el Análisis y Diseño que empleó fue el lenguaje UML (Unified Modeling Language). El método de desarrollo que se empleó en el presente proyecto fue el Método Empírico. Utilizó Linux como plataforma, PHP como lenguaje de Programación, Mysql como gestor de Base de Datos, Apache como servidor de paginas Web. [FLORES, 2003]

"PORTAL DE GESTIÓN ACADÉMICA ESTUDIANTIL VIA WEB POGAW" del Postulante: Jesús Lalo Flores Rojas, cuyo objetivo fue apoyar a la educación generando un espacio en Internet utilizando nuevas tecnologías, que permita involucrar a padres, docentes y estudiantes en el proceso educativo, ofreciendo información rápida y coherente a cada uno de ellos, para tal efecto desarrolló los módulos de *Gestión Académica, Servicios Web, Control y Seguimiento de*

Alumnos, control de Pensiones, Admisión de Alumnos, Reportes. Para el Análisis y Diseño empleó la metodología planteada por Craig Larman, utilizó Linux como plataforma, PHP como lenguaje de Programación, Mysql como gestor de Base de Datos y Apache como servidor de paginas Web. [FLORES, 2005]

E. PROYECTOS REALIZADOS PARA EL INSTITUTO NORMAL SUPERIOR SIMÓN BOLIVAR

"SISTEMA DE SEGUIMIENTO ACADÉMICO INSTITUTO NORMAL SUPERIOR SIMÓN BOLIVAR", de las Postulantes: Pérez Elizabeth y Laura Wendy, cuyo objetivo fue Analizar, diseñar e implementar un sistema de información para seguimiento del área académica del INSSB, para de esta manera resolver la problemática suscitada al interior de la Institución por ausencia de un control automático de estos aspectos. Comprende los módulos de *Inscripciones, Registro de alumnos, Registro y control de notas, Seguimiento académico, Matriculación, Reportes, Emisión de certificados de notas y Mantenimiento.* Para el diseño del Sistema utilizaron la Metodología, del "Análisis Estructurado Moderno" de Edward Yourdon. Para el desarrollo e implementación utilizaron Delphi 5.0 como lenguaje de programación, SQL Server 7.0 como gestor de base de datos. [PEREZ Y LAURA, 2001].

1.3. ANALISIS DE LA SITUACION ACTUAL

De acuerdo a la observación que se realizó en las faenas de inscripción, matriculación y emisión de documentos, además de entrevistas con el personal administrativo y estudiantil, se vió que actualmente las Áreas de Contabilidad y Archivo Kardex del Instituto Normal Superior Simón Bolívar, encargadas de efectuar los procesos de matriculación e inscripción respectivamente, realizan estas tareas de forma manual, produciendo demora considerable y trabajo excesivo, la implementación de un Sistema de Información en esta área, agilizaría y optimizaría el manejo de la información. Hubo una ocasión en que dos postulantes desarrollaron un sistema para el área académica del INSSB, sin embargo la institución no cuenta ni ha contado con ningún sistema de información en el Área Académica, toda la información de los estudiantes son almacenadas en hojas de Microsoft Excel.

En los últimos años se ha visto una creciente cantidad de postulantes que eligen la profesión de Docencia y quieren ingresar a esta casa de Estudio Superior, en consecuencia la población estudiantil del Instituto Normal Superior Simón Bolívar va en aumento y se presume que este comportamiento continúe. De mantenerse el manejo de la información de forma manual, como hasta el momento se lo ha estado haciendo, llegará un momento en que esta cantidad de información no podrá ser controlada por la Institución y se verá en una situación problemática difícil de remediar.

1.4. PLANTEAMIENTO DEL PROBLEMA

De acuerdo a la descripción del punto anterior, el incremento de la población estudiantil del INSSB hace que la información sea excesiva y difícil de manejarla adecuadamente, además de que este arduo trabajo se realiza de forma manual y esta concentrado en pocas personas, haciendo dificultosa y morosa estas tareas.

1.4.1 FORMULACIÓN DEL PROBLEMA

¿El Sistema de Gestión Académica para el Instituto Normal Superior Simón Bolívar, coadyuvará en los procesos de Registro de estudiantes nuevos, matriculación, inscripción, registro y control de notas, seguimiento académico, emisión de documentos y reportes, proporcionando información eficiente y oportuna?

1.5. OBJETIVOS

1.5.1. OBJETIVO GENERAL

Desarrollar e Implementar el Sistema de Gestión Académica para el Instituto Normal Superior Simón Bolívar, que coadyuvará en los procesos del área académica, para proporcionar información eficiente y oportuna.

1.5.2 OBJETIVOS ESPECÍFICOS

- Gestionar en forma automática los procesos académicos que permitan agilizar las tareas realizadas en el Área Académica del Instituto Normal Superior Simón Bolívar.
- Realizar el registro de estudiantes nuevos a la base de datos y emitir la Hoja de Ingreso Personal.
- Realizar la matriculación y emisión de la Matrícula de estudiantes regulares.
- Realizar la inscripción e impresión de la Boleta de Inscripción de los estudiantes.
- Implementar el módulo Registro y Control de notas que permita realizar el registro de Notas a docentes de manera rápida y confiable.
- Implementar el módulo de Consulta de estudiantes para que realicen su seguimiento académico y puedan acceder a información como ser: materias inscritas, notas del último periodo, historial de notas, pensum y registro personal.
- Emitir documentos: certificado de notas, historial de notas, globales, listas oficiales y boletines de notas de forma automática, rápida y confiable.
- Emitir reportes y cuadros comparativos (detalle de estudiantes por especialidad, sexo, edad, idioma, tipo de colegio, etc.) para ayudar a la toma de decisiones a nivel ejecutivo.
- Capacitar al personal encargado que trabajará directamente con el Sistema de Información.

1.6. JUSTIFICACION

1.6.1. JUSTIFICACION TECNICA

El continuo desarrollo tecnológico de la ciencia de las computadoras ha permitido masificar el uso de esta herramienta, incrementando cada vez más el manejo de grandes cantidades de información, dando lugar a que Instituciones dedicadas al área de la educación superior, tal es el caso del INSSB, deban adecuarse a estas nuevas tecnologías para mejorar sus servicios, eliminando los procesos manuales que conducen a la demora en la realización de las diferentes labores que realiza el personal administrativo del Área Académica.

En cuanto a recursos de Hardware y Software, la institución cuenta con equipos suficientes y una red local instalada donde se puede implementar el sistema de información, cuenta con aproximadamente 30 computadoras en los diferentes departamentos (Dirección General, Dirección Académica, Centro Informático, Archivo Kardex, Contabilidad, Jefatura de Secundaria, Jefatura de Primaria, etc.) interconectadas mediante una red de área local a un servidor ubicado en el Centro informático, también cuenta con un Laboratorio de Computación con aproximadamente 25 computadoras, la mayoría de estos equipos son Pentium III y Pentium IV; en cuanto a Software cuentan con la Licencia del sistema operativo Windows Milenium, y utilizan Microsoft Excel para almacenar los registros de los estudiantes.

Para el desarrollo del Sistema de Información se utilizará como lenguaje de programación PHP 5, MySQL como Gestor de Base de Datos y Apache como Servidor Web, debido a los muchos beneficios que ofrecen además que son de uso gratuito y no habrá algún costo extra en cuanto a licencias se refiere.

La tecnología Cliente/Servidor utilizada es la topología estrella, para compartir o distribuir la información de la Base de Datos que esté disponible en cualquier computador conectado a la red, utilizando medidas de seguridad para que esta información pueda ser accedida solo por usuarios autorizados, además de usar una interfaz adecuada y fácil de operar por cualquier usuario.

1.6.2. JUSTIFICACION ECONÓMICA

Actualmente la información es un recurso muy importante para toda Institución que maneja grandes volúmenes de datos, creemos que todas estas entidades deben automatizar su información, ya que esto implica un ahorro de tiempo considerable, eliminando trabajo excesivo, minimizando el uso de material de escritorio, mejorando la productividad y reduciendo los costos relacionados con la recopilación y distribución de información.

Por otra parte el Sistema de Información Integrado se desarrollará utilizando Software Libre, lo que implica total independencia en cuestión a Licencias, es decir el lenguaje de programación (PHP), el gestor de base de datos (MySQL) y el servidor (Apache) son de uso gratuito, no requieren de licencias y no tienen costo alguno. A esto podemos añadir además la plataforma de Windows Me, la cual cuenta con la Licencia correspondiente, adquirida por el Instituto Normal Superior Simón Bolívar. Además el INSSB cuenta con la instalación de una red local lo cual implica la facilidad en cuanto a la instalación y configuración del Sistema en la Institución.

1.6.3. JUSTIFICACION SOCIAL

En las ultimas gestiones la cantidad de estudiantes del INSSB ha tenido un crecimiento considerable, esto debido a la gran demanda que esta Institución tiene, si embargo la información de estas personas es manipulada de forma manual, pero con la implementación del Sistema de Información se pretende beneficiar a la Institución, realzando su prestigio y proporcionando credibilidad en el manejo de la información de sus estudiantes, también beneficiará al personal administrativo, ayudando en los procesos del área académica, reduciendo el trabajo excesivo y moroso, para una mejor atención a la comunidad normalista y usuarios en general. Además beneficiará directamente al estudiante, ya que podrá realizar los diferentes procesos académicos de manera rápida y confiable, tendrá la posibilidad de realizar su seguimiento académico mediante consultas a través de Internet, evitando así las aglomeraciones y largas colas existentes en el área de Archivo Kardex.

1.7. LÍMITES Y ALCANCES

El alcance del Sistema de Gestión Académica comprende los módulos que se especifican a continuación:

- ***Módulo Registro de Estudiantes Nuevos.***
 - Registra semestralmente toda la información de los estudiantes nuevos a la base de datos.
 - Emite la Hoja de Ingreso Personal, que es archivado en el file del estudiante.
 - Realiza el listado de estudiantes registrados (por gestión, carrera y especialidad).
 - Incluye una búsqueda de estudiantes registrados por C.I.
- ***Módulo Matriculación.***
 - Realiza el registro de la matriculación.
 - Emite la impresión de la matricula correspondiente a cada semestre con los datos propios del estudiante.
 - Realiza un listado de estudiantes matriculados por gestión, carrera y especialidad.
 - Realiza una búsqueda de estudiantes matriculados por gestión y C.I.
- ***Módulo Inscripción.***
 - Registra la inscripción semestral de los estudiantes en las materias y paralelos correspondientes.
 - Emite la Boleta de Inscripción.
 - Realiza un listado por gestión, carrera, especialidad y paralelo.
 - Realiza una búsqueda de estudiantes inscritos por gestión y C.I.

- ***Módulo Registro y control de notas.***

Módulo utilizado por los docentes quienes:

- Registran las notas de los estudiantes una vez culminada la gestión.
- Emiten el boletín de notas, una vez registrado las notas.
- Consultan materias a dictar con sus respectivos horarios.
- Acceden a listas oficiales de estudiantes inscritos en sus materias.

- ***Módulo Seguimiento Académico de los estudiantes.***

Este módulo será para el uso de los estudiantes, quienes podrán realizar consultas y la impresión de algunos documentos, como ser:

- Materias inscritas.
- Notas del último periodo.
- Historial de notas.
- Hoja de ingreso personal.
- Pensum de su especialidad.

- ***Módulo Emisión de documentos***

A este módulo tendrán acceso el personal de Archivo y Kardex, quienes deben realizar la emisión de documentos certificados, que frecuentemente se solicita, como ser:

- Hoja de Ingreso Personal.
- Boleta de Inscripción.
- Pensum por especialidad.
- Certificado de Notas.
- Historial de Notas.
- Emisión de Globales.
- Listas oficiales de estudiantes inscritos por paralelo.

- ***Módulo de Consultas y Reportes.***

Este módulo se encargará de emitir los diferentes documentos y reportes requeridos por la Dirección General y Académica como ser:

- Cuadros comparativos de la cantidad de estudiantes matriculados por carrera, especialidad y gestión.
- Cuadros comparativos de la cantidad de estudiantes inscritos por carrera, especialidad y gestión.
- Cuadros comparativos de la cantidad de estudiantes registrados por carrera, especialidad y gestión.
- Detalle de la cantidad de estudiantes inscritos por paralelo.
- Cantidad de mujeres y varones registrados en carrera y especialidad, por gestión.
- Estado civil (solteros/as, casados/as), de estudiantes registrados por gestión.
- Edad mínima, máxima y promedio de estudiantes registrados por gestión.
- Registro de estudiantes que hablan algún idioma nativo (aymará, quechua, guaraní, otros), por gestión.
- Registro del tipo de colegio (fiscal, particular, convenio) de egreso de los estudiantes, por gestión.
- Ciudad de egreso de los estudiantes registrados (La Paz, El Alto, otros), por gestión.

- ***Módulo de Administración del Sistema.***

Este módulo estará encargado de:

- Administrar el diseño académico
 - Registro de materias a la base de datos.
 - Elaboración del pensum.
 - Asignación de docente-materia.
 - Habilitación de paralelos.

- Administración de usuarios
 - Registro de usuarios.
 - Asignación de niveles de usuario a personal encargado, (docentes y administrativos)
- Administración de procesos académicos.
 - Registro de estudiantes nuevos.
 - Matriculación.
 - Inscripción.
 - Registro y control de notas.
 - Seguimiento Académico.



CAPÍTULO II



MARCO TEÓRICO

CAPÍTULO II

MARCO TEÓRICO

2.1. INTRODUCCIÓN

El mundo de la informática no para de hablar de procesos de desarrollo, el modo de trabajar eficientemente para evitar catástrofes que llevan a que un gran porcentaje de proyectos se terminen sin éxito.

El objetivo de un proceso de desarrollo es subir la calidad del software (en todas las fases por las que pasa) a través de una mayor transparencia y control sobre el proceso. Da igual si es algo casero o para un cliente, hay que producir lo esperado en el tiempo esperado y con el costo esperado. Es labor del proceso de desarrollo hacer que esas medidas para aumentar la calidad sean reproducibles en cada desarrollo. [MOLPECERES, 2003]

Muchas veces se realiza el diseño del Software de una manera rígida, con los requerimientos que el cliente solicita, de tal manera que cuando el cliente en la etapa final (etapa de pruebas) solicita un cambio se hace muy difícil realizarlo, pues si se hace una modificación en el diseño del producto, en la arquitectura del sistema o en la infraestructura de software y hardware puede llevar a un retraso o al fracaso completo del proyecto.

La ingeniería del software es una disciplina que integra al proceso, los métodos y las herramientas para el desarrollo del software de computadora. [PRESSMAN, 2005]

2.2. INGENIERÍA DEL SOFTWARE

La Ingeniería del Software, según Pressman, es una tecnología estratificada, ilustrada en la figura 2.1, que debe estar sustentada en un compromiso de calidad.



Figura 2.1. Estratos de la ingeniería de software
[PRESSMAN, 2005]

Pressman define los estratos como:

El proceso es la base de la ingeniería del software, elemento que mantiene unidos los estratos de la tecnología y que permite un desarrollo racional y a tiempo del software de computadora. También forma la base para el control de la gestión de los proyectos de software y establece el contexto en el cual se aplican los métodos técnicos, se generan los productos del trabajo como son los modelos, documentos, datos reportes, formatos, etc., se establecen los fundamentos, se asegura la calidad, y el cambio se maneja de manera apropiada.

Los métodos son los que proporcionan la manera técnica para construir software. Abarcan un amplio espectro de tareas que incluyen la comunicación, el análisis de requisitos, el modelado del diseño, la construcción del programa, la realización de pruebas y el soporte.

Las herramientas proporcionan el soporte automatizado o semiautomatizado para el proceso y los métodos.

2.3. INGENIERÍA WEB

La ingeniería Web (IWeb) es el proceso bajo el cual se crean sistemas y aplicaciones basados en Web (WebApps) de alta calidad. La ingeniería Web no es un clon perfecto de la ingeniería del software, pero toma prestados muchos de los conceptos y principios fundamentales de ella. El proceso de la IWeb acentúa actividades técnicas y administrativas similares, existiendo diferencias sutiles en la forma como se dirigen dichas actividades, pero el método esencial señala que el enfoque debe ser disciplinado para el desarrollo de un sistema basado en computadora. [PRESSMAN, 2005].

La ingeniería Web aplica principios científicos, de ingeniería y de administración, y enfoques disciplinados y sistemáticos para el desarrollo, despliegue y mantenimiento exitosos de sistemas y aplicaciones basadas en Web de alta calidad.

En este capítulo se presenta una breve exposición de los conceptos y métodos relacionados con el proyecto basados en la Metodología de Craig Larman con la herramienta **UML**, para procesos de desarrollo de sistemas orientado a objetos (**análisis y diseño orientado a objetos**) que se funden en un enfoque **iterativo e incremental** de los **casos de uso**. [LARMAN, 1999]

2.4. ANALISIS Y DISEÑO ORIENTADO A OBJETOS

Para crear una aplicación de software hay que describir el problema y las necesidades o requerimientos: en que consiste el conflicto y que debe hacerse. El Análisis se centra en una investigación del problema, no en la manera de definir una solución. El Diseño pone de relieve una solución lógica: como el sistema cumple con los requerimientos, contando con descripciones detalladas y de alto nivel de la solución lógica y saber como satisface los requerimientos y las restricciones.

La esencia del análisis y el diseño orientado a objetos consiste en situar el dominio de un problema y su solución lógica dentro de la perspectiva de los objetos (cosas, conceptos o entidades), como se advierte en la figura 2.2.



Figura 2.2. Significado de las actividades de desarrollo.
[LARMAN, 1999]

2.4.1. ANÁLISIS ORIENTADO A OBJETOS.

Durante el análisis orientado a objetos se procura ante todo identificar y describir los objetos – o conceptos – dentro del dominio del problema. El propósito del análisis orientado a objetos es definir todas las clases que son relevantes al problema que se va a resolver, las operaciones y atributos asociados, las relaciones y comportamiento asociadas con ellas. Para cumplirlo se deben ejecutar las siguientes tareas:

Los requisitos básicos del usuario, el cual debe comunicarse entre el cliente y el ingeniero del software.

- Identificar clases, donde se deben definir atributos y métodos
- Se debe especificar una jerarquía de clases.
- Representan las relaciones objeto a objeto
- Modelar el comportamiento del objeto
- Repetir iterativamente estas tareas hasta completar el modelo

Este análisis, como los métodos de análisis convencional forman un análisis multiparte para satisfacer este objetivo [PRESSMAN, 2005].

2.4.2 DISEÑO ORIENTADO A OBJETOS.

La naturaleza del diseño orientado a objetos, reside en su capacidad de construir, procura definir los objetos lógicos del software que finalmente serán implementado, los objetos tienen atributos y métodos.

El diseño es el primer paso en la fase de desarrollo de cualquier producto o sistema de ingeniería. El diseño orientado a objetos transforma el modelo de análisis creado usando el análisis orientado a objetos en un modelo de diseño que sirve como un anteproyecto para la construcción del software.

Durante el diseño orientado a objetos, se procura definir los objetos lógicos del software que finalmente serán implementados en el lenguaje de programación orientado a objetos. Los objetos tienen atributos y métodos.

2.4.3. CONSTRUCCION.

Finalmente, durante la construcción o programación orientada a objetos, se implementan los componentes del diseño, como una clase *Libro* en C++, Java, Smalltalk o Visual Basic.

2.5. MODELO DE DESARROLLO ITERATIVO E INCREMENTAL

El modelo de desarrollo iterativo e incremental se utiliza para la construcción de sistemas que evolucionan a lo largo del tiempo, ya que existen problemas en el transcurso del desarrollo que llegan a alcanzar grados elevados de complejidad, siendo lo más práctico dividir el trabajo en partes pequeñas o miniproyectos. Donde cada miniproyecto llega a ser una iteración que resulta en un incremento. Las iteraciones hacen referencia a pasos de flujo de trabajo, y los incrementos, al crecimiento del producto.

El proceso iterativo e incremental consta de una secuencia de iteraciones, cada iteración aborda una parte de la funcionalidad total, pasando por todos los flujos de trabajo relevantes y refinando la arquitectura. Cada iteración se analiza cuando termina. Se puede determinar si han aparecido nuevos requisitos o han cambiado los existentes, afectando a las iteraciones siguientes. Durante la planificación de los detalles de la siguiente iteración, el equipo también examina cómo afectarán los riesgos que aún quedan al trabajo en curso. Toda la retroalimentación de la iteración pasada permite reajustar los objetivos para las siguientes iteraciones. Se continúa con esta dinámica hasta que se haya finalizado por completo con la versión actual del producto.

2.5.1. MODELO ITERATIVO

El ciclo de vida iterativo se basa en la evolución de prototipos ejecutables que se muestran a los usuarios y clientes,



Figura 2.3. Modelo Iterativo [PRESSMAN, 2005]

Para que sea efectivo, las iteraciones deberían estar controladas; esto es que se deberían seleccionar y ejecutar de una forma planificada por el mismo hecho de que son miniproyectos.

En cada iteración, los desarrolladores identifican y especifican los casos de uso relevantes, crean un diseño utilizando la arquitectura seleccionada. Si una iteración cumple con sus objetivos, el desarrollo continúa con la siguiente iteración; pero en cambio si una iteración no cumple sus objetivos, el desarrollador debe revisar sus decisiones previas y probar un nuevo enfoque.

2.5.2. MODELO INCREMENTAL

Combina elementos del modelo lineal secuencial y de construcción de prototipos ver Figura 2.4, puesto que permite la entrega del avance del producto operacional al cliente, en cada presentación. Este modelo al igual que otros modelos evolutivos permite al desarrollador construir versiones mas completas del software, y la construcción de prototipos presenta un enfoque iterativo por naturaleza [PRESSMAN, 2005].

Aplica secuencias lineales de forma escalonada mientras progresa en el calendario, cada secuencia lineal produce un incremento del software [PRESSMAN, 2005].



Figura 2.4. Ciclos de desarrollo iterativo [PRESSMAN, 2005]

En los ciclos iterativos se produce una retroalimentación que permite ajustar nuestros objetivos para el siguiente ciclo, de esta manera se lleva a cabo

el tiempo programando hasta tener el producto desarrollado listo para la implementación.

Entre las características que presenta este modelo están:

- Cada iteración es una etapa dentro de un proyecto.
- Descomposición de un proyecto grande en mini proyectos.
- Las iteraciones deben estar controladas.
- Cada iteración trata un conjunto de casos de uso.
- Cada miniproyecto es una iteración.



2.6. LENGUAJE DE MODELADO UNIFICADO, UML

El UML (Lenguaje Unificado para la construcción de Modelos) se define como un "lenguaje que permite especificar, visualizar y construir los artefactos de los sistemas de software...". Es un sistema notacional destinado a los sistemas de modelado que utilizan concepto orientado a objetos.

Se ha convertido en el estándar de facto de la industria, debido a que ha sido concebido por los autores de los tres métodos más usados de orientación a objetos: Grady Booch, Ivar Jacobson y Jim Rumbaugh. Estos autores fueron contratados por la empresa Rational Software Co. para crear una notación unificada en la que basar la construcción de sus herramientas CASE.

El UML (Unified Modeling Language) es el resultado de varios intentos de estandarización de las diversas notaciones de métodos de análisis y diseños orientados a objetos. Es así que en 1994 por iniciativa de Grady Booch y Jim Rumbaugh se unieron para combinar sus dos famosos métodos: el de Booch y el OMT (Object Modeling Technique, Técnica de Modelado de Objetos). Más tarde se les unió Ivar Jacobson, creador del método OOSE (Object-Oriented Software Engineering, Ingeniería de Software Orientada a Objetos). En respuesta a una petición de OMG (Object Management Group, asociación para fijar los estándares de la industria) para definir un lenguaje y una notación estándar del lenguaje de construcción de modelos, en 1997 propusieron el UML como candidato [LARMAN, 1999].

Podría UML ser utilizado con cualquier proceso o modelo de desarrollo de software. Sin embargo se recomienda que para un uso óptimo sea usado por un modelo que fuese dirigido por los casos de uso, centrado en la arquitectura iterativo e incremental.

2.7. METODOLOGIA DE DESARROLLO

La metodología usada será la planteada por Larman Craig, [LARMAN, 1999], el enfoque que toma es el de un ciclo de vida iterativo incremental, el cual permite una gran flexibilidad a la hora de adaptarlo a un proyecto. El ciclo de vida está dirigido por casos de uso, es decir, por la funcionalidad que ofrece el sistema, con el cual se puede resolver la necesidad del usuario.

No se trata de un método nuevo, sino la descripción de modelos y procesos recomendados (MPR) que utilizan los profesionales y que, con diversos nombres y ligeras modificaciones, se incluyen en otros métodos del análisis y diseño orientado a objetos [Rumbaugh97].

2.7.1. PASOS DE MACRONIVEL

La metodología de Craig Larman esta constituida por 3 fases fundamentales (Ver Fig. 2.5):



Figura 2.5. Pasos de macronivel en el desarrollo [LARMAN, 1999]

Donde la fase de Planeación y Elaboración, es donde se desarrollará la planificación del proyecto, se definen los requisitos del sistema, la construcción de prototipos, etc.

En la fase de Construcción es donde esencialmente se crea el sistema, teniendo diferentes etapas en ella.

Finalmente se encuentra la parte de Aplicación, la cual es la puesta en marcha del sistema en el entorno previsto de uso.

De ellas, la fase de Construcción es la que consume la mayor parte del esfuerzo y del tiempo en un proyecto de desarrollo. Para llevarla a cabo se va adoptar un enfoque iterativo a través de múltiples ciclos de desarrollo de análisis, diseño, implementación y pruebas. En cada ciclo se aborda un conjunto relativamente pequeño de requerimientos (agrupados según casos de uso) y llevándolo a través del análisis y diseño hasta la implementación y pruebas. El sistema va creciendo incrementalmente con cada ciclo que concluye.

Con esta aproximación se consigue disminuir el grado de complejidad que se trata en cada ciclo, y se tiene pronto en el proceso una parte del sistema funcionando el cual se va retroalimentando gradualmente.

2.7.2. FASE DE PLANEACIÓN Y ELABORACIÓN

En esta fase del proyecto se establece un proceso a seguir, principalmente es la captura de requisitos del sistema, se ve el alcance que tendrá el mismo, en sí es una abstracción que se hace de la problemática, es donde se desarrolla en líneas generales una investigación de alternativas sobre la planificación que se hará del proyecto. El proceso está formado por una serie de actividades y subactividades que se tiene, cuya realización se va repitiendo en el tiempo aplicadas a distintos elementos, observar Figura 2.6.

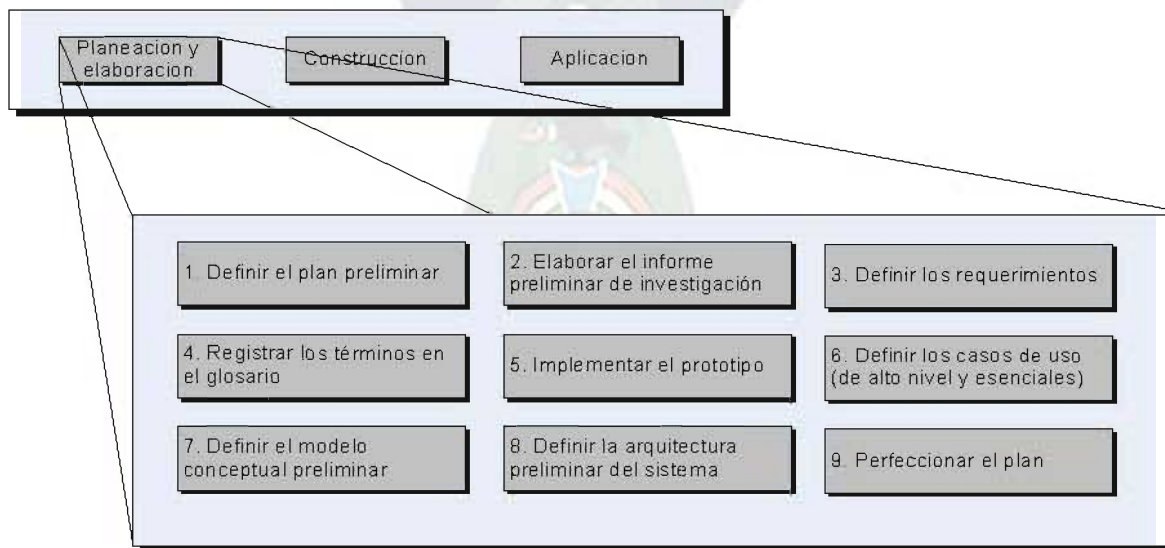


Figura 2.6 Actividades de la Fase de Planeación y Elaboración [LARMAN, 1999]

En la Figura 2.6. se observa las actividades que debe seguirse en esta fase:

1. *Definir el Plan Preliminar.* Se realiza un borrador de lo que se hará, se definen programas, asignación de recursos, se estiman presupuestos en función del proyecto que se pretende desarrollar.
2. *Crear el Informe de Investigación Preliminar.* Se realiza un análisis con los motivos del porque se realizará, alternativas, necesidades que tiene la institución o empresa por lo que se opto desarrollar una solución.
3. *Definir los Requisitos.* Se declaran los requerimientos que tiene la institución, las cuales serán resueltas a partir de la solución que se proponga.
4. *Registrar Términos en el Glosario.* Se desarrolla un diccionario con los nombres, algunos conceptos a mas detalle, además de toda la información concerniente al tema de estudio.
5. *Implementar un Prototipo.* Se define una primera versión de la solución cuyo fin es facilitar la comprensión del problema.
6. *Definir Casos de Uso (de alto nivel y esenciales).* Descripciones narrativas de los casos de uso y sus relaciones.
7. *Definir el Modelo Conceptual Preliminar:* Cuya finalidad es facilitar el conocimiento del vocabulario del dominio, especialmente en su relación con los casos de uso y con las especificaciones de los requerimientos.
8. *Definir la Arquitectura del Sistema-Borrador.* (Puede retrasarse hasta una fase posterior). Se desarrolla un bosquejo de la estructura física - lógica del sistema en desarrollo.
9. *Perfeccionar el Plan:* Una vez analizado todo lo anterior, se puede ya realizar un refinamiento del plan.

Una de las actividades principales de esta fase es la especificación de requisitos, lo que nos lleva a preguntar:

La captura de requisitos es complicada, ya que los desarrolladores de software al crear sistemas para usuarios, uno al principio no puede saber que es lo que necesita el usuario, que es lo que espera que el sistema realice, para ello los usuarios deben especificar lo más claramente posible lo que desean. Sin embargo, se ha visto que en la mayoría al obtener los requisitos del usuario es una fuente imperfecta de recolección de información, ya que ellos no siempre pueden ver como puede hacerse mas eficiente la operación en su conjunto. La mayoría de ellos no sabe qué partes de su trabajo pueden transformarse en software.

En esta fase se observan actividades, las cuales tienen dependencia entre sí (Ver Fig. 2.7).

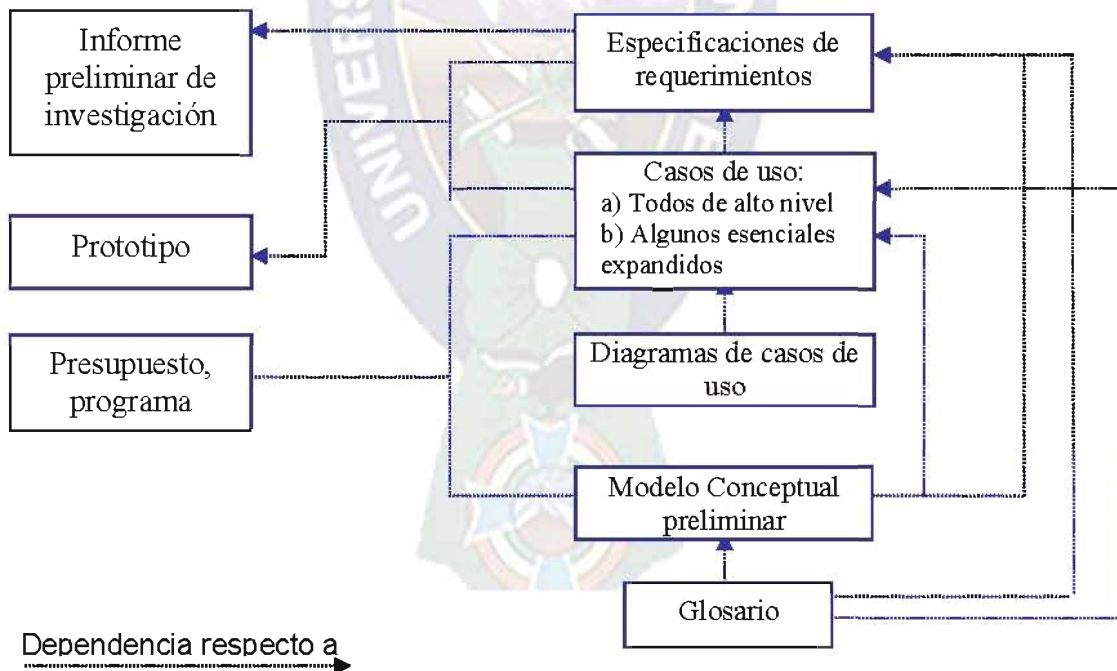


Figura 2.7. Actividades de dependencia de la Fase de Planeación y Elaboración [LARMAN, 1999]

2.7.2.1. ESPECIFICACIÓN DE REQUERIMIENTOS

En el desarrollo de un proyecto de software, las necesidades o deseos del usuario son muy importantes, Craig Larman en su metodología recomienda que se empleen los siguientes artefactos:

- **Presentación general.** Se desarrolla una narrativa correspondiente a la problemática que se desea resolver.
- **Clientes.** Son aquellas personas o instituciones donde se llevará acabo el proyecto.
- **Metas.** Se considera metas a los objetivos que se desea alcanzar con el desarrollo de la solución.
- **Funciones del sistema.** Las funciones que presenta un sistema significan lo que éste habrá de realizar, pudiéndose dividir entre las siguientes categorías, como se muestra en la tabla 2.1.

Categoría de la función	Significado
Evidente	Debe realizarse, y el usuario debería saber que se ha realizado.
Oculto	Debe realizarse, aunque no es visible para los usuarios.
Superflua	Opcionales, su inclusión no produce ninguna clase de fallos.

Tabla 2.1. Categoría de las funciones [LARMAN, 1999]

- **Evidente:** Como su nombre lo indica es una función que el cliente puede hasta visualizarlo observando el sistema de manera tal que se pueda verificar que lo que se a pedido se ha hecho.
- **Oculto:** Son operaciones que realiza el sistema pero de manera lógica, por ejemplo el usuario no podrá ver como es que se realiza la búsqueda de un alumno en la base de datos, es decir son procesos que no están a la vista del usuario.
- **Superflua:** Significa operaciones opcionales que puede o no presentar el sistema.

- **Atributos del sistema.** Los atributos del sistema son las características o cualidades que tiene el sistema como se puede apreciar en la tabla 2.2, por ejemplo un atributo sería la facilidad de uso que presenta el sistema; no son las funciones que realiza el mismo.

ATRIBUTO	DETALLES Y RESTRICCIONES DE FRONTERA
Tiempo de respuesta	Es el tiempo que demorará el sistema en responder a la petición del usuario
Metáfora de interfaz	Es el detalle del sistema, el gráfico, el colorido, es decir la presentación del sistema, de cuan amigable es el mismo para el usuario
Tolerancia a fallas	Son las posibles fallas que pueda ocurrir para que el sistema pueda tener fallas
Plataforma del sistema operativo	Que plataformas se usarán

Tabla 2.2. Atributos de un sistema [LARMAN, 1999]

- **Tiempo de respuesta:** El tiempo de respuesta se lo puede calcular en una unidad de tiempo como ser los segundos, esto se lo puede realizar evaluando cuanto tardará el sistema en por ejemplo en verificar algunos datos o en la solicitud de impresión, etc.
- **Metáfora de interfaz:** Al hablar de metáfora de interfaz, nos referimos a la interfaz hombre – máquina, de cómo vera el usuario el sistema si será de fácil manejo para el usuario.
- **Tolerancia a fallas:** En esta parte se puede describir de cómo el sistema puede responder a fallas que pueden ocurrir, por ejemplo una falla sería el corte eléctrico de cuanta información se puede llegar a perder.
- **Plataformas del sistema operativo:** Se refiere a que plataformas son las indicadas para que el sistema este en perfecto funcionamiento, a que lenguajes se utilizarán para el desarrollo del mismo, por ejemplo un sistema en MSDOS no será tan amigable como uno echo en un lenguaje visual.

2.7.2.2. CASOS DE USO: DESCRIPCIÓN DE PROCESOS

En este caso se emplea la notación UML para poder desarrollar una descripción del problema, los casos de uso permiten mejorar la comprensión de los requerimientos.

2.7.2.2.1. CASOS DE USO

El caso de uso es un documento narrativo que describe la secuencia de eventos de un actor, que utiliza un sistema para completar un proceso. Un caso de uso es una descripción de la secuencia de interacciones que se producen entre un actor y el sistema, cuando el actor usa el sistema para llevar a cabo una tarea específica.

El nombre del caso de uso debe reflejar la tarea específica que el actor desea llevar a cabo usando el sistema, debería ser un verbo para enfatizar que se trata de un proceso. (Ver Fig. 2.8)

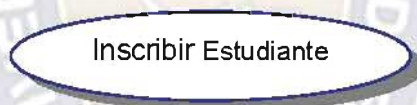


Figura 2.8. Icono del UML que representa un Caso de Uso [LARMAN, 1999]

Existen pasos para la identificación de los casos de uso esto requiere una lluvia de ideas y revisar los documentos actuales sobre la especificación de los requerimientos.

Un método con que se identifican los casos de uso se basa en los actores:

- Se identifican los actores relacionados con un sistema.
- En cada actor, se identifican los procesos que inician o en que participan

2.7.2.2.2. ACTORES

Un actor es una entidad externa al sistema que realiza algún tipo de interacción con el mismo. Se representa mediante una figura humana dibujada con palotes (Ver Fig. 2.9). Esta representación sirve tanto para actores que son personas como para otro tipo de actores (otros sistemas, sensores, etc.).



Figura 2.9. Representación de un Actor [LARMAN, 1999]

2.7.2.2.3. FORMATO DE LOS CASOS DE USO

Los casos de uso pueden ser expresados en dos formatos:

2.7.2.2.3.1. CASOS DE USO DE ALTO NIVEL. Permite entender los principales procesos globales, Larman sugiere la estructura de la tabla 2.3.

Casos de uso:	Nombre del caso de uso
Actores:	Lista de actores (Agentes externos), en la cual se indica quien inicia el caso de uso.
Tipo:	Primario, secundario u opcional (a explicar) Esencial o real (a explicar)
Descripción:	Resumen del proceso concerniente al Caso de Uso.

Tabla 2.3. Estructura de casos de alto nivel [LARMAN, 1999]

2.7.2.2.3.2. CASOS DE USO CON FORMATO EXPANDIDO. Muestran más detalle que uno en formato de alto nivel, este tipo de casos permite alcanzar un conocimiento mas profundo del proceso y requerimientos que se esta tratando, Craig Larman propone el formato que se advierte en la tabla 2.4.

Casos de uso:	Nombre del caso de uso
Actores:	Lista de actores (Agentes externos), en la cual se indica quien inicia el caso de uso.
Propósito:	Intención del caso de uso
Resumen:	Repetición del caso de uso de alto nivel o alguna síntesis similar
Tipo:	1. Primario, secundario u opcional 2. Esencial o Real
Referencia cruzada:	Casos relacionados de uso y funciones también relacionadas del sistema.

Curso normal de los eventos

Acción del actor	Respuesta del sistema
Acciones numeradas de los actores.	Descripciones numeradas de las respuestas del sistema.

Cursos alternos

Alternativas que pueden ocurrir en e número de línea. Descripción de excepciones
--

Tabla 2.4. Estructura de casos de uso expandido [LARMAN, 1999]

Los casos de uso pueden ser:

- **Casos de uso Primarios.** Representan los procesos más importantes.
- **Casos de uso Secundarios.** Corresponden a los procesos menores o raros.
- **Casos de uso Opcionales.** Estos casos de uso corresponden a los procesos que pueden no abordarse.

ó

- **Casos de uso Esenciales.** Se expresan en forma teórica y son muy abstractas.
- **Casos de uso Reales.** Describen concretamente el proceso a partir de su diseño actual.

2.7.2.3. DIAGRAMAS DE LOS CASOS DE USO

Un diagrama de casos de uso explica gráficamente un conjunto de casos de uso de sistema, los actores y la relación entre estos y los casos de uso (Ver Fig. 2.10). Hay líneas de comunicaciones entre los casos y los actores; las flechas indican el flujo de la información o el estímulo.

El diagrama tiene por objeto ofrecer una clase de diagrama contextual que nos permite conocer rápidamente los actores externos de un sistema y las formas básicas en que lo utilizan.

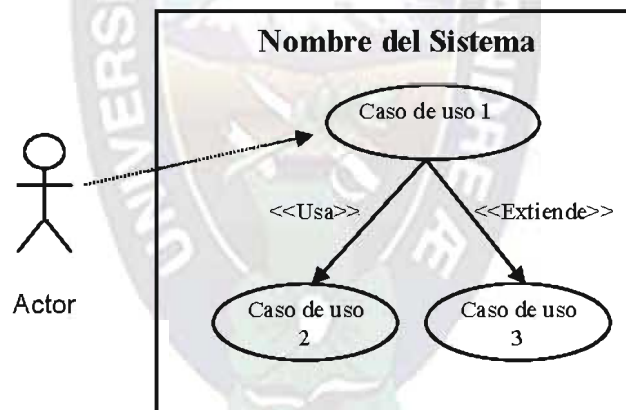


Figura 2.10. Diagrama de casos de uso [LARMAN, 1999]

2.7.3. FASE DE CONSTRUCCIÓN

La fase de construcción requiere de varios ciclos de desarrollo (probablemente con plazos fijos) a lo largo de los cuales se extiende el sistema. Un ciclo individual de desarrollo esta conformado por dos etapas: Análisis y Diseño. (Ver Fig. 2.11), para mejorar el prototipo cuando llegue la hora de la entrega del producto. El objetivo es llegar a obtener un sistema de software que llene las expectativas del usuario. El análisis y el diseño consiste en definir el dominio de un problema y su solución lógica.

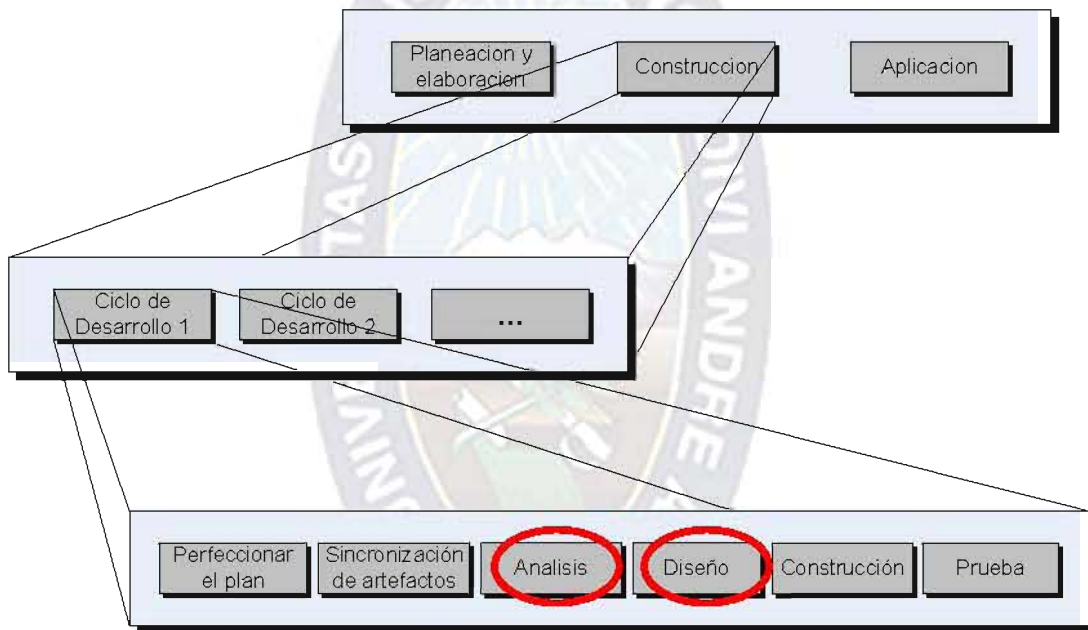


Figura 2.11. Ciclo de Desarrollo [LARMAN, 1999]

Durante la fase de Análisis lo que se desea ante todo es identificar y describir los objetos (conceptos) dentro del dominio del problema.

En la fase de Diseño, el objetivo es definir los objetos lógicos del software que finalmente serán implementados en un lenguaje de programación. A continuación se presenta una explicación etapa por etapa que corresponden a esta fase.

2.7.3.1. ETAPA DE ANÁLISIS

Una vez concluida la fase de planeación y elaboración, donde se han identificado ya los requerimientos y los casos de uso, comienza la etapa de análisis de la fase de construcción donde se siguen actividades, en la cual se cumplen los ciclos del desarrollo iterativo e incremental.

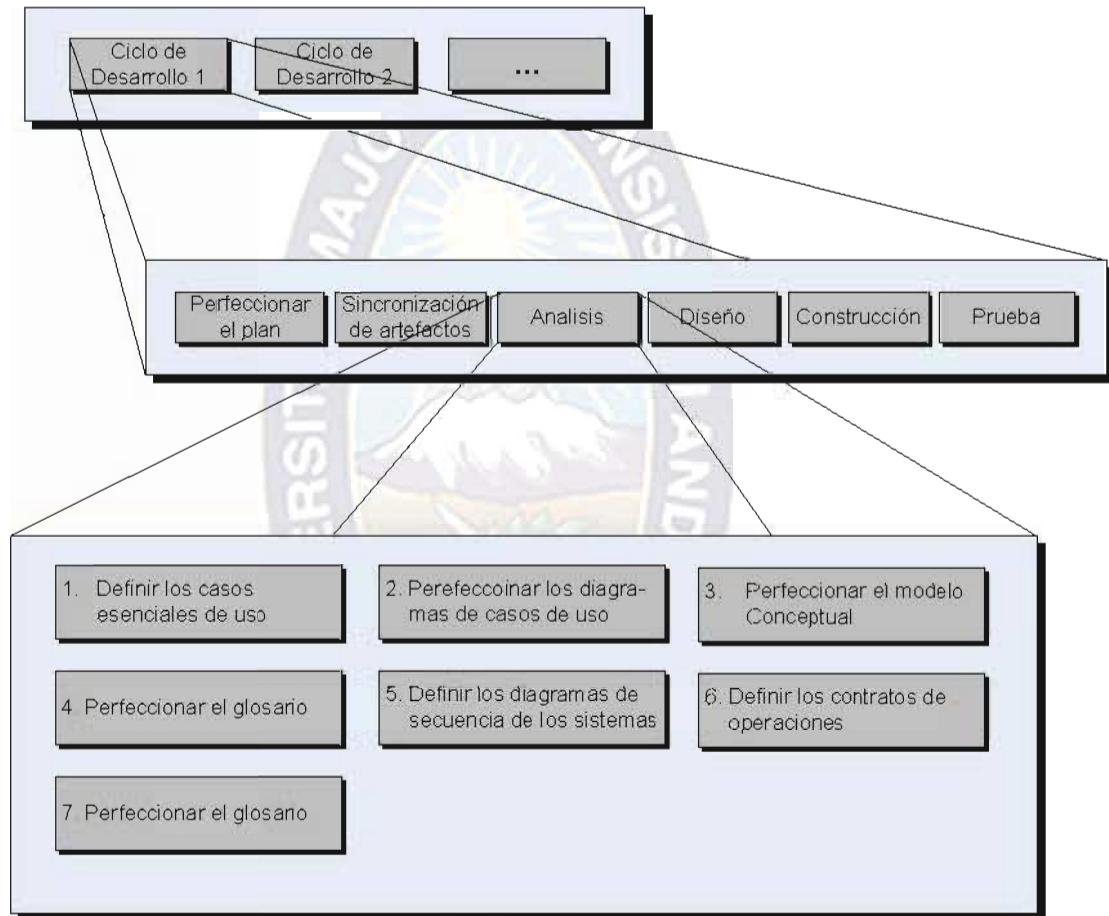


Figura 2.12. Actividades de la etapa de Análisis dentro un ciclo de desarrollo [LARMAN, 1999]

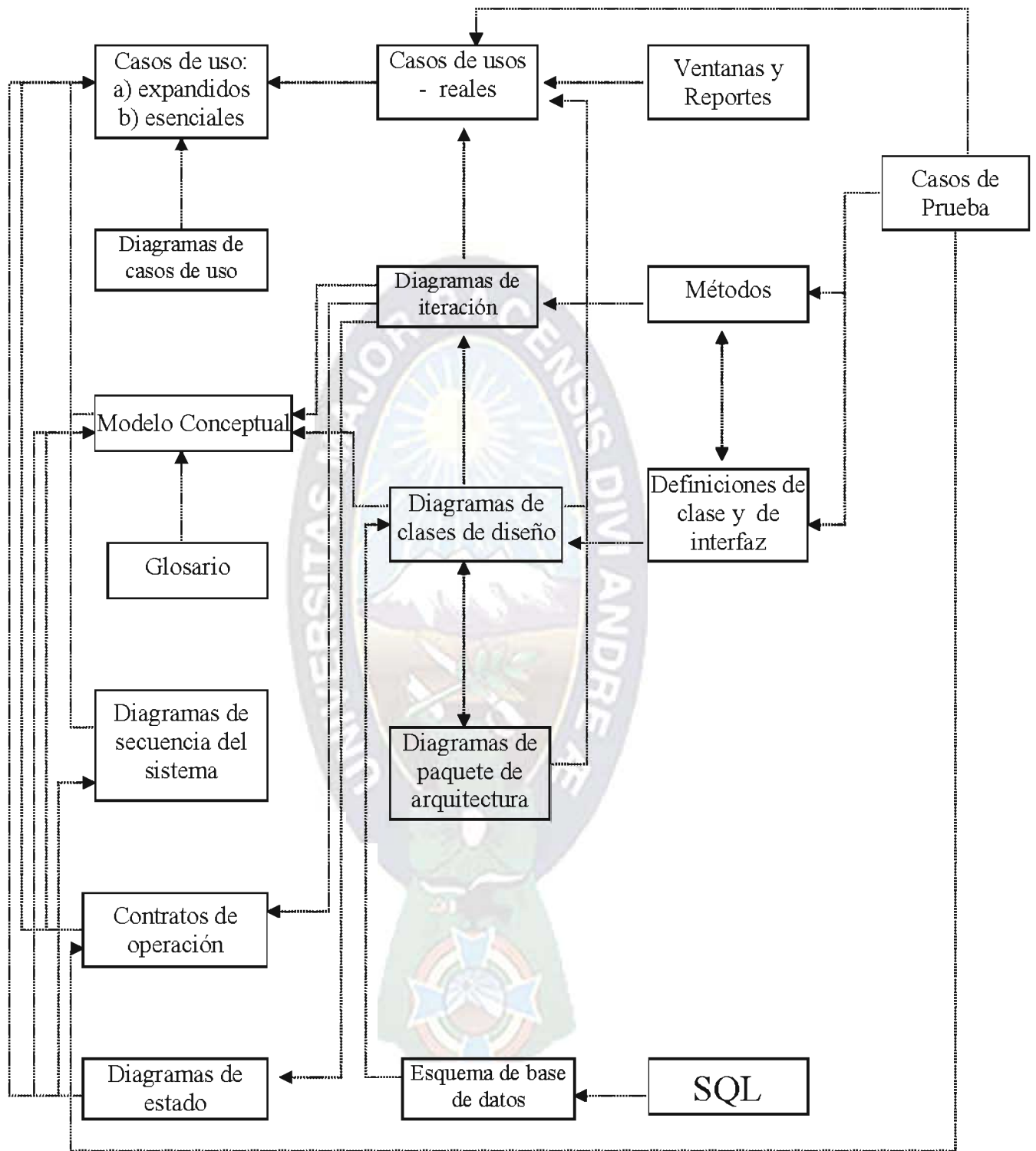


Figura 2.13. Dependencias de los artefactos durante la Fase de Construcción [LARMAN, 1999]

2.7.3.1.1. MODELO CONCEPTUAL

Un modelo conceptual explica (a sus creadores) los conceptos significativos en un dominio del problema; es el artefacto más importante a crear durante el análisis orientado a objetos, representa esencialmente cosas del mundo real – los casos de uso son un importante artefacto del análisis de requerimientos.

En lo referente a este modelo, no se debe tomar en cuenta lo que es el software, como las interfaces de usuario (ventana), ni la base de datos, los procesos.

El paso esencial de un análisis o investigación orientados a objetos es descomponer el problema en conceptos y objetos individuales. Un modelo conceptual es una representación de conceptos en un dominio del problema. En el UML, los ilustramos con un grupo de diagramas de estructura estática donde no se define ninguna operación.

Cuando nos referimos al modelo conceptual, este nos puede mostrar:

- Conceptos
- Asociaciones entre conceptos
- Atributos de conceptos



Figura 2.14. Modelo conceptual parcial, muestra conceptos del mundo real [LARMAN, 1999]

Craig Larman, para desarrollar el modelo conceptual recomienda los siguientes pasos:

- 1) Liste los conceptos idóneos usando la Lista de categorías de conceptos y la identificación de la frase nominal relacionada con los requerimientos en cuestión.
- 2) Dibújelos en un modelo conceptual.
- 3) Incorpore las asociaciones necesarias para registrar las relaciones para las cuales debe reservar un espacio en la memoria.
- 4) Agregue los atributos necesarios para cumplir con las necesidades de información.

Es necesario identificar las asociaciones de los conceptos, lo que contribuirá a entender el modelo conceptual. Una asociación es una relación entre dos conceptos, lo que indica alguna conexión entre ellos.

Al generarse el modelo conceptual, se realiza la asociación luego es necesario identificar los atributos de los conceptos que se necesitan para satisfacer los requerimientos de información de los casos de uso. Un atributo es un valor lógico de un dato de un objeto.

Los extremos de una asociación pueden contener una expresión de multiplicidad que indique la relación numérica entre las instancias de los conceptos.

La multiplicidad define cuantas instancias de un tipo A pueden asociarse a una instancia del tipo B en determinado momento

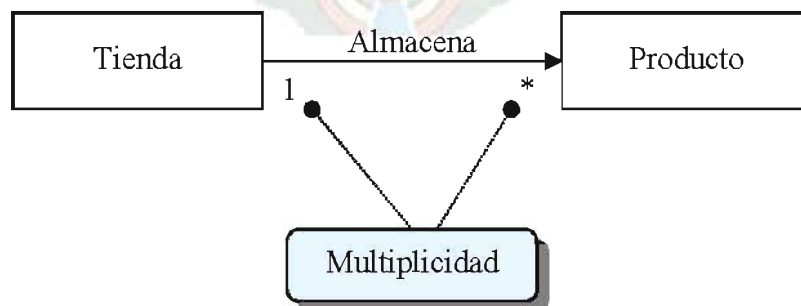


Figura 2.15 Multiplicidad en una asociación [LARMAN 1999]

Por ejemplo, una instancia individual de una *Tienda* puede asociarse a “muchas” instancias (cero o mas marcadas con *) de *Producto*.

En la figura 2.16 se ofrecen algunas expresiones de multiplicidad.

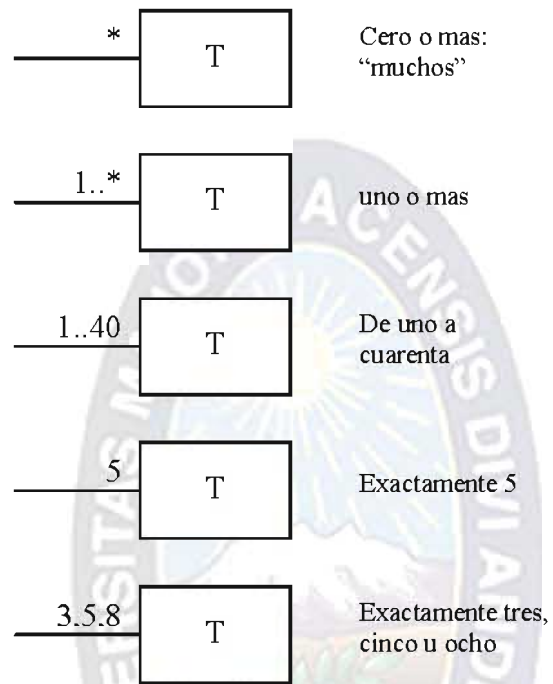


Figura 2.16 Valores de multiplicidad [LARMAN 1999]

2.7.3.1.2. REGISTRO DE LOS TÉRMINOS DEL GLOSARIO

El glosario o diccionario de datos, es semejante a un diccionario de datos el cual incluye y define todos los términos que requieren explicación para mejorar la comunicación entre el usuario y el analista, para que de esta manera se tenga una misma comprensión entre ambos. Por lo general se realiza junto a la especificación de requerimientos desde la primera fase, pero se lo va perfeccionando en cada ciclo de desarrollo que pasa.

El formato que sigue un glosario se lo puede apreciar en la tabla 2.5:

TÉRMINO	CATEGORÍA	COMENTARIO
Nombre del término	Atributo	Descripción detallada del término.
	Tipo	
	Caso de uso	

Tabla 2.5. Formato del glosario [LARMAN, 1999]

2.7.3.1.3. DIAGRAMAS DE SECUENCIA DEL SISTEMA

El diagrama de secuencia de un sistema muestra gráficamente los eventos que fluyen de los actores al sistema. Es una representación que muestra, en un determinado escenario de un caso de uso, los eventos generados por actores externos, su orden y los eventos internos del sistema, donde a todos los sistemas se les trata como una caja negra (Ver Fig. 2.17).

El UML ofrece una notación con los diagramas de secuencia. Su creación comienza a partir de los casos de uso identificados en la primera fase, es decir que se origina a partir de la formulación previa de los casos de uso.

Para elaborar diagramas de secuencia de un sistema, de tal manera que describa el curso normal de los eventos en un caso de uso, se puede seguir los siguientes pasos:

- 1) Trace una línea que represente el sistema como un caja negra.
- 2) Identifique los actores que operan directamente sobre el sistema. Trace una línea para cada uno de ellos.
- 3) A partir del curso normal de los eventos del caso de uso identifique los eventos ("externos") del sistema que son generados por los actores. Muéstreselos gráficamente en el diagrama.
- 4) A la izquierda del diagrama puede incluir o no el texto del caso de uso.

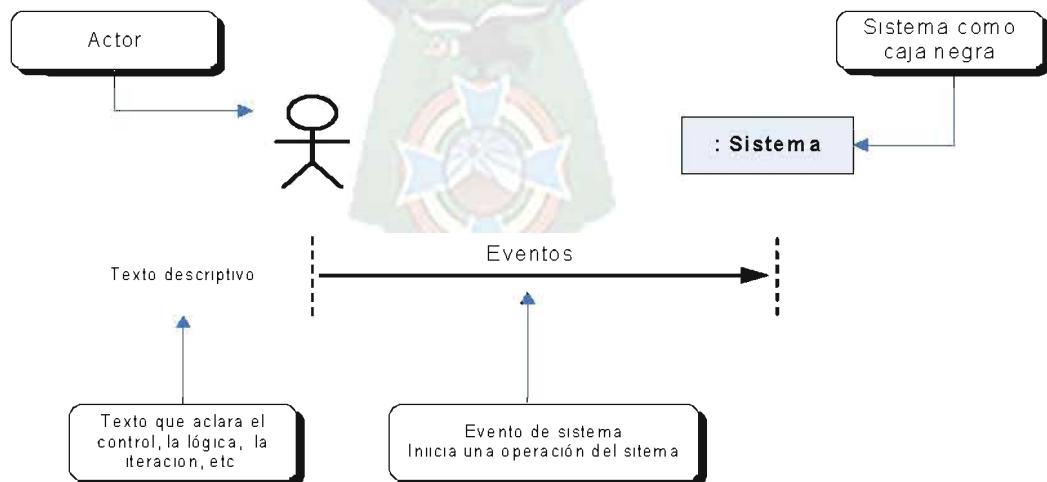


Figura 2.17. Formato de Diagrama de Secuencia [LARMAN, 1999]

2.7.3.1.4. CONTRATOS DE OPERACIONES

Los contratos contribuyen a definir el comportamiento de un sistema a partir de como cambia el estado de un sistema cuando se llama una operación suya, es decir que describen el efecto que sobre él tiene las operaciones. La preparación de los contratos de operación depende del desarrollo previo del modelo conceptual, de los diagramas de la secuencia del sistema.

A continuación presentamos en la tabla 2.6 la estructura de un contrato de operaciones:

CONTRATO	
Nombre	Nombre de la operación y parámetros
Responsabilidades	Descripción informal de la responsabilidad que debe cumplir la operación
Tipo	Nombre del tipo (concepto, clase de software, interfaz)
Referencias cruzadas	Número de referencia de las funciones del sistema, casos de uso, etc.
Notas	Notas de diseño, algoritmos e información afín.
Excepciones	Casos excepcionales.
Salida	No salida de la interfaz del usuario (mensajes o registros que se envían fuera)
Precondiciones	Suposición acerca del estado del sistema antes de ejecutar la operación.
Poscondiciones	El estado del sistema después de la operación.

Tabla 2.6. Estructura de un Contrato [LARMAN, 1999]

2.7.3.2. ETAPA DE DISEÑO

En la fase de análisis del desarrollo se da prioridad al conocimiento de los requerimientos, los conceptos y las operaciones relacionadas con el sistema. A menudo la investigación y el análisis se caracterizan por estar centradas en cuestiones concernientes al qué, es decir: cuáles son los procesos, que son los conceptos, etc.

Durante el ciclo de desarrollo iterativo es posible pasar a la fase de diseño, una vez terminados estos documentos del análisis. En la etapa de diseño se logra una solución lógica, su esencia es la elaboración de diagramas de interacción, que muestran gráficamente cómo los objetos se comunicaran entre ellos a fin de cumplir con los requerimientos.

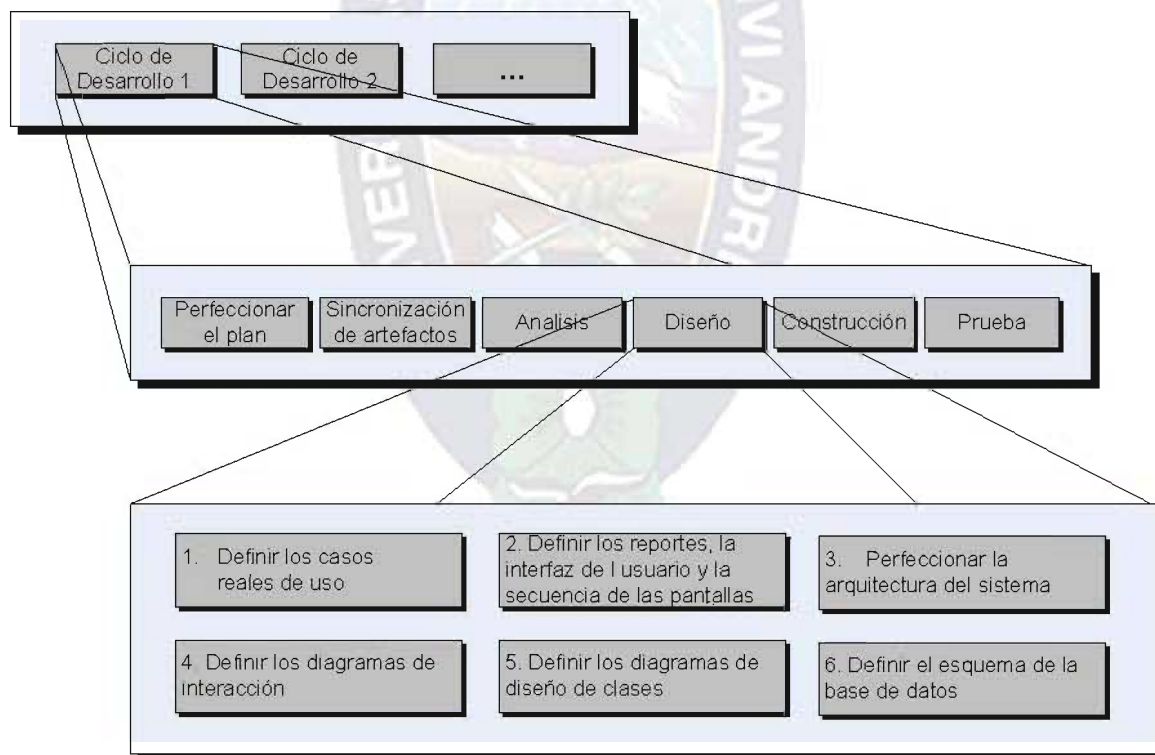


Figura 2.18. Actividades de la Etapa de Diseño dentro un ciclo de desarrollo
[LARMAN, 1999]

Ahora daremos los conceptos de las actividades que tiene esta fase:

2.7.3.2.1. DESCRIPCIÓN DE LOS CASOS DE USO REALES

La creación de los casos de uso reales depende de los casos de uso esenciales. El caso real de uso describe el diseño de forma concreta la manera en la que se realiza un caso de uso (proceso), a partir de los conceptos de entrada y salida en lo que se refiere a una tecnología en particular, así como de la implementación de la misma.

En los casos de uso reales se realiza el modelado de la interfaz (ventanas), una alternativa sería realizar una secuencia de las pantallas de la interfaz general de manera secuencial los cuales se presentará al usuario para que las revise, así después se irá incorporando los detalles durante la implementación.

En el siguiente ejemplo y en la figura 2.19, observe la utilización de un esquema de codificación con los artefactos de ventana para obtener una descripción fluida.

Caso de uso:	Comprar productos
Actores	Cliente (iniciador), Cajero.
Propósito:	Capturar una venta y su pago en efectivo.
Resumen:	Un cliente llega a la caja con productos que desea comprar. El Cajero registra los productos de la compra y recibe el pago en efectivo. Al terminar la transacción, el Cliente se marcha con los productos comprados.
Tipo:	Primario y real
Referencias cruzadas:	Funciones: R1.1, R1.2, R1.3, R1.7, R2.1.

Tabla 2.7. Casos de Uso Reales [LARMAN, 1999]



Figura 2.19. Ventana [LARMAN, 1999]

Curso normal de los eventos

Acción de los actores

Respuesta del sistema

1. Este caso comienza cuando un Cliente llega a la caja TPDV con objetos que desea comprar.

2. Con cada producto, el Cajero teclea el código universal de producto (CUP) en **A** de la ventana. Si hay más de un producto, es opcional capturar la cantidad en **E**. Se oprime **H** después de capturar cada producto.

4. Al terminar de capturar los productos, el Cajero oprime el botón **I** para indicarle a la TPDV que termino de capturar los productos.

3. Agrega la información sobre el producto a la actual transacción de ventas. La descripción y el precio del producto actual se muestran en **B** y en **F** de la ventana

5. Calcula y presenta en **C** el total de la venta.

6. ...

Tabla 2.8. Curso normal de los eventos. [LARMAN, 1999]

2.7.3.2.2. DEFINIR DIAGRAMAS DE INTERACCIÓN

Un diagrama de interacción explica gráficamente las interacciones existentes entre las instancias de las clases. Los diagramas de interacción se generan una vez realizado modelo conceptual, los contratos de operaciones, y los casos de uso reales o esenciales.

En los diagramas de interacción se definen dos tipos de diagramas:

- Diagramas de colaboración
- Diagrama de secuencia

La representación de los Diagramas de secuencia ya se vieron en la fase de análisis, así que procederemos a detallar la representación de los diagramas de colaboración.



2.7.3.2.2.1. DIAGRAMAS DE COLABORACIÓN

Los Diagramas de colaboración describen las interacciones entre los objetos en un formato de grafo o red.

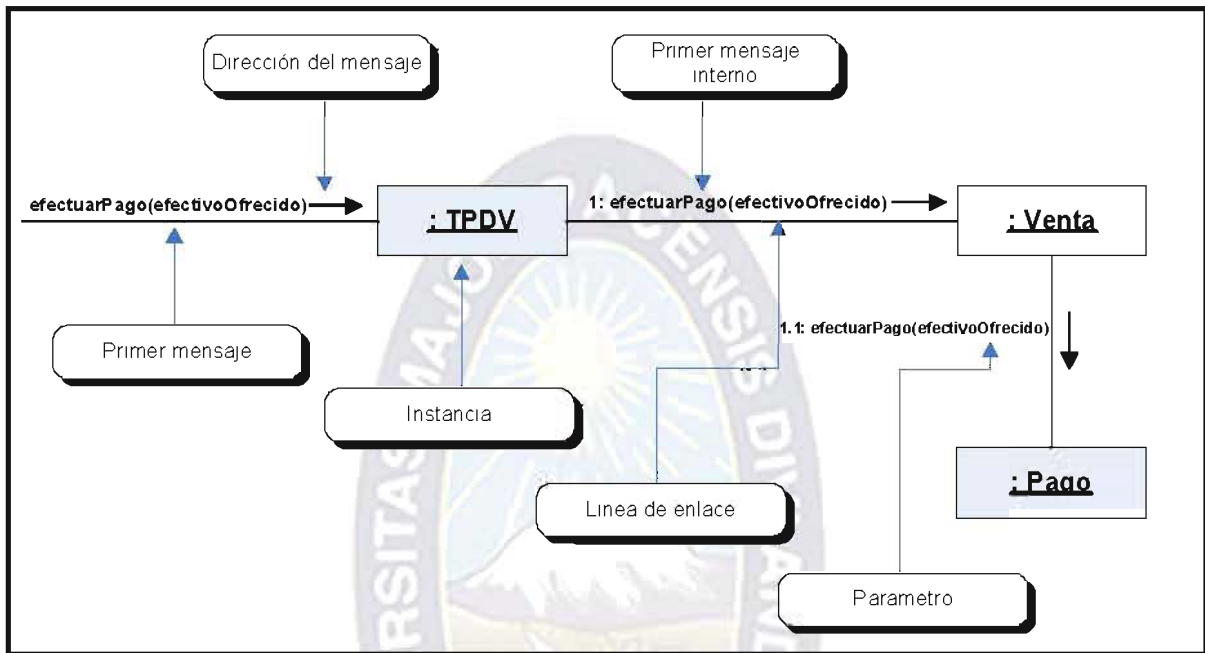


Figura 2.20. Diagrama de Colaboración [LARMAN, 1999]

El diagrama de colaboración de la figura 2.20 se lee así:

1. El mensaje *efectuarPago* se envía a una instancia de *TPDV*. La instancia corresponde al mensaje *efectuarPago* de la operación del sistema
2. El objeto *TPDV* envía mensaje *efectuarPago* a la instancia *Venta*
3. El objeto *Venta* crea una instancia de un *Pago*

Para preparar diagramas de colaboración se aplican los siguientes pasos:

1) Elaborar un diagrama por cada operación del sistema durante el ciclo actual de desarrollo.

- En cada mensaje del sistema, dibuje un diagrama incluyendo como mensaje inicial.

2) Si el diagrama se torna complejo, divídalo en diagramas más pequeños.

3) Diseñe un sistema de objetos interactivos que realicen las tareas, usando como punto de partida las responsabilidades del contrato de operación, las poscondiciones y las descripciones de casos de uso.

Es posible definir mensajes condicionales, se indica posponiendo al número de la secuencia una cláusula condicional entre corchetes. El mensaje se envía solo si la cláusula se evalúa como verdadera.

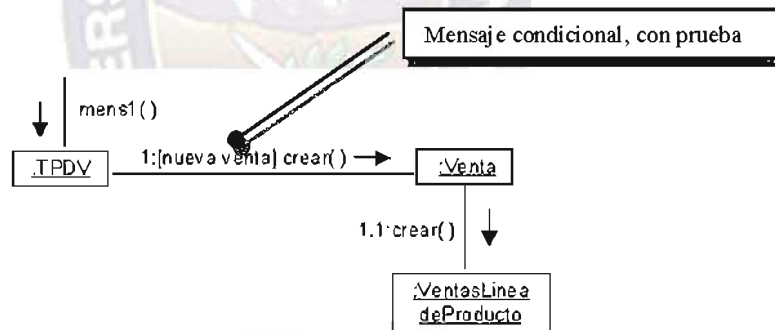


Figura 2.21. Diagrama de Colaboración: Mensajes Condicionales [LARMAN, 1999]

La relación de los diagramas de colaboración y otros artefactos se la puede observar gráficamente a continuación.

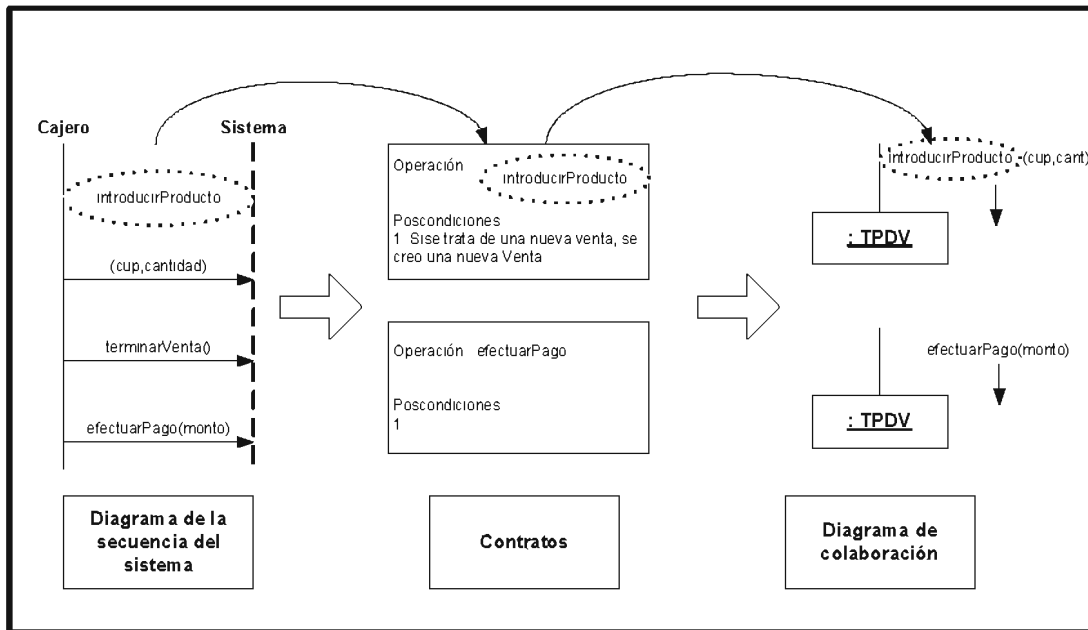


Figura 2.22. Relación entre los artefactos. [LARMAN, 1999]

Como se observa en la figura 2.22, la relación entre los artefactos incluye lo siguiente:

- Los casos de uso indican los eventos del sistema que se muestran explícitamente en los diagramas de su secuencia.
- En los contratos se describe la mejor conjetura inicial sobre las operaciones del sistema.
- Las operaciones del sistema representan mensajes y estos originan diagramas que explican gráficamente como los objetos interactúan para llevar a cabo las funciones requeridas.

2.7.3.2.3. DIAGRAMAS DE CLASES

El diagrama de clases describe gráficamente las especificaciones de las clases de software y de las interfaces en una aplicación.

A continuación se identifican los siguientes elementos en la figura. Además de que, para poder realizar el diagrama de clases se debe tener elaborado el diagrama de interacción y el modelo conceptual.

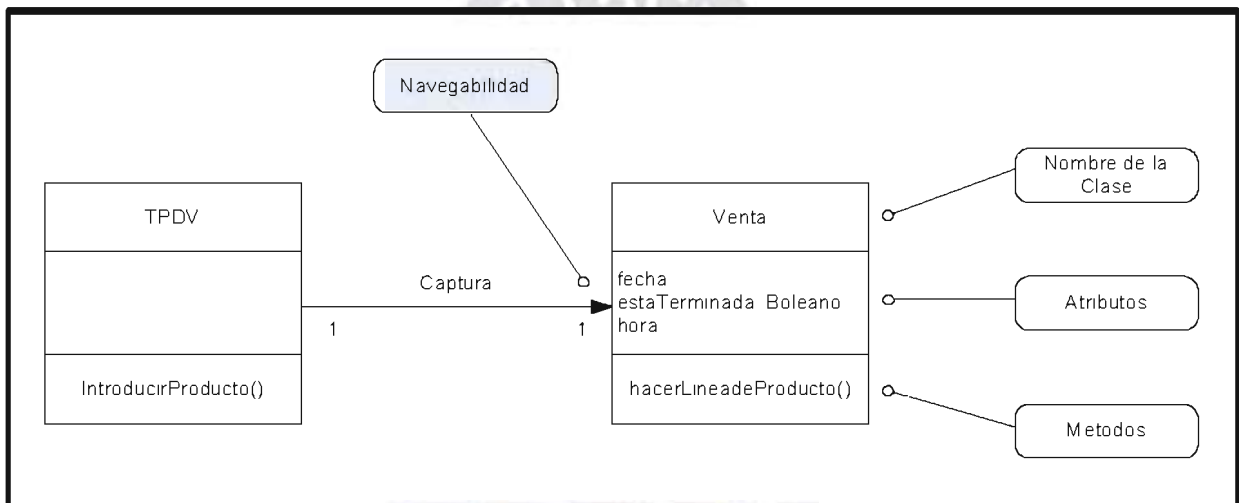


Figura 2.23. Diagrama de clases del diseño [LARMAN, 1999]

Para realizar un diagrama de clases se siguen los siguientes pasos:

1. Identifique todas las clases que participan en la solución del software. Para ello analizar los diagramas de interacción.
2. Dibújelas en un diagrama de clases.
3. Duplique los atributos provenientes de los conceptos asociados del modelo conceptual.
4. Agregue los nombres de los métodos analizados los diagramas de interacción.
5. Incorpore la información sobre lo tipos a los atributos y a los métodos.
6. Agregue las asociaciones necesarias para dar soporte a la visibilidad requerida de los atributos.
7. Agregue flechas de navegabilidad a las asociaciones para indicar la dirección de la visibilidad de los atributos.

8. Agregue las líneas de relaciones de dependencia para indicar la visibilidad no relacionada con los atributos.

2.7.3.2.3.1. CLASE

Una clase define los atributos y los métodos de una serie de objetos. Todos los objetos de esta clase (instancias de esa clase) tienen el mismo comportamiento y el mismo conjunto de atributos (cada objetos tiene el suyo propio).

Las clases están representadas por rectángulos, con el nombre de la clase, y también pueden mostrar atributos y operaciones de la clase en otros dos "compartimentos" dentro del rectángulo.

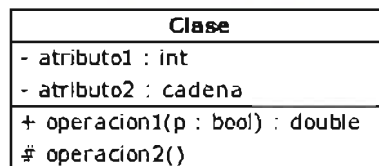


Figura 2.24. Representación visual de una clase en UML

2.7.3.2.3.1.1. ATRIBUTOS

En UML, los atributos se muestran al menos con su nombre, y también pueden mostrar su tipo, valor inicial y otras propiedades. Los atributos también pueden ser mostrados visualmente:

- + Indica atributos *públicos*
- # Indica atributos *protegidos*
- - Indica atributos *privados*

2.7.3.2.3.1.2. OPERACIONES

Las operaciones (métodos) también se muestran al menos con su nombre, y pueden mostrar sus parámetros y valores de retorno. Las operaciones, al igual que los atributos, se pueden mostrar visualmente:

- + Indica operaciones *públicas*
- # Indica operaciones *protegidas*
- - Indica operaciones *privadas*

2.7.3.2.3.2. ASOCIACIONES DE CLASES

Las clases se puede relaciones (estar asociadas) con otras de diferentes maneras:

2.7.3.2.3.2.1. GENERALIZACIÓN

La herencia es uno de los conceptos fundamentales de la programación orientada a objetos, en la que una clase "recoge" todos los atributos y operaciones de la clase de la que es heredera, y puede alterar/modificar algunos de ellos, así como añadir más atributos y operaciones propias.

En UML, una asociación de *generalización* entre dos clases, coloca a estas en una jerarquía que representa el concepto de herencia de una clase derivada de la clase base. En UML, las generalizaciones se representan por medio de una línea que conecta las dos clases, con una flecha en el lado de la clase base.

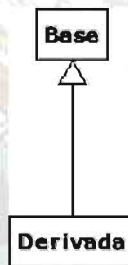


Figura 2.25. Representación visual de una generalización en UML

2.7.3.2.3.2.2. ASOCIACIONES

Una asociación representa una relación entre clases, y aporta la semántica común y la estructura de muchos tipos de "conexiones" entre objetos. Describe la conexión entre diferentes clases. Cada extremo de la asociación también tiene un valor de multiplicidad, que indica cuántos objetos de ese lado de la asociación están relacionados con un objeto del extremo contrario.

2.7.3.2.3.2.3. MULTIPLICIDAD

La multiplicidad se muestra como un rango [mín...máx] de valores no negativos, con un asterisco (*) representando el infinito en el lado máximo.



Figura 2.26. Representación visual de una asociación en UML

Puede determinarse por la especificación de multiplicidad (mínima...máxima)

- 1 Uno y sólo uno
- 0..1 Cero o uno
- M..N Desde M hasta N (enteros naturales)
- * Cero o muchos
- 0..* Cero o muchos
- 1..* Uno o muchos (al menos uno)

2.7.3.2.4. DIAGRAMAS DE ESTADOS

Antes debemos definir que es un evento, estado y transición. Un evento es un acontecimiento importante o digno de señalar, un estado es la condición de un objeto en un momento determinado, el tiempo que transcurre entre eventos, mientras que la transición es una relación entre dos estados.

Un diagrama de estado describe visualmente los estados y eventos más interesantes de un objeto, así como su comportamiento ante un evento. Las transiciones se muestran con flechas que llevan el nombre del evento respectivo. Los estados se colocan en óvalos. Se acostumbra incluir un seudoestado inicial que cumple automáticamente la transición a otro estado en el momento de crear una instancia.

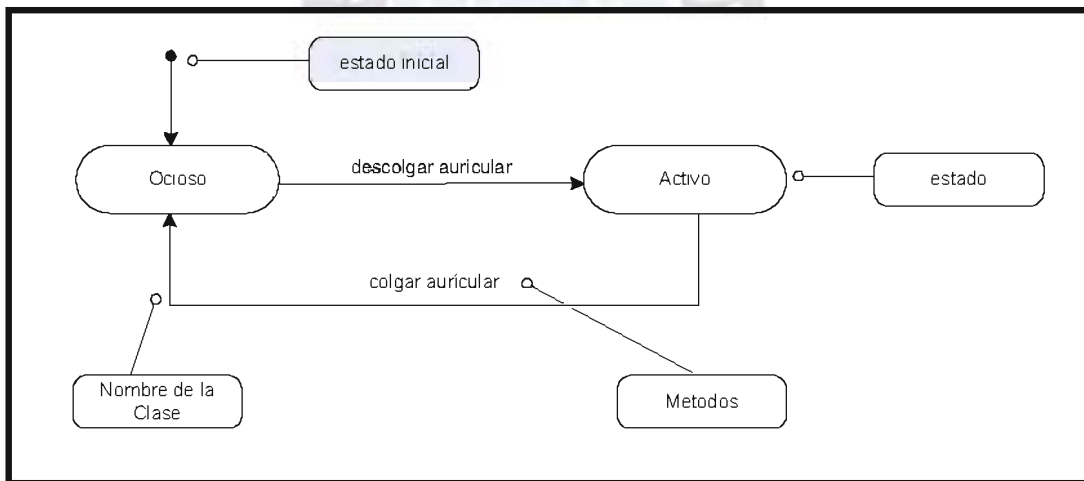


Figura 2.27. Diagrama de estado para un teléfono
[LARMAN, 1999]

Un diagrama de estado presenta el ciclo de vida de un objeto, los eventos que el ocurren, sus transiciones y los estados que median entre esos eventos. Estos diagramas pueden aplicarse a varios elementos, como ser:

- Clases de software
- Tipos (conceptos)
- Casos de uso

Además de los diagramas de estado para los casos de uso o el sistema global, se pueden crear diagramas prácticamente para cualquier tipo o clase.

2.7.3.2.4.1. GENERALIZACIÓN DE ESTADOS:

Con la generalización de estados:

- Podemos reducir la complejidad de estos diagramas usando la generalización de estados.
- Distinguimos así entre superestado y subestados.
- Un estado puede contener varios subestados disjuntos.
- Los subestados heredan las variables de estado y las transiciones externas.
- La agregación de estados es la composición de un estado a partir de varios estados independientes.



2.7.4. FASE DE APLICACIÓN

El ciclo iterativo de la segunda fase ocurre hasta que el sistema responda eficientemente a los requerimientos del usuario.

Una vez terminado las anteriores fases lo único que queda es la implementación del sistema. La fase de aplicación, es la tercera etapa y final de esta metodología. El proceso o actividades que le siguen son (Ver Fig. 2.28):



Figura 2.28. Actividades de la Fase de Aplicación [LARMAN, 1999]

2.7.4.1. PRUEBAS

Se diseñan e implementan las pruebas creando los casos de prueba. Un caso de prueba especifica una forma de probar el sistema, incluyendo la entrada o resultado con la que se ha de probar y las condiciones bajo las que ha de probarse. [Jacobson, 2000]

Se pueden especificar otros casos de prueba para probar el sistema como un todo:

- Las *pruebas de instalación* verifican que el sistema puede ser instalado en la plataforma del cliente y que el sistema funcionará correctamente cuando sea instalado.
- Las *pruebas de configuración* verifican que el sistema funciona correctamente en diferentes configuraciones.
- Las *pruebas negativas* intentan provocar que el sistema falle para poder así revelar sus debilidades.
- Las *pruebas de tensión o de estrés* identifican problemas con el sistema cuando hay recursos insuficientes o cuando hay competencia por los recursos.
- Los casos de prueba de *integración del sistema* se utilizan para verificar que los componentes interaccionan entre sí de la forma apropiada después de haber sido integrados en una construcción.



2.8. ARQUITECTURA CLIENTE / SERVIDOR

Esta arquitectura es un modelo para el desarrollo de sistemas de información, en el cual múltiples clientes distribuidos geográficamente realizan transacciones las que se dividen en procesos independientes que cooperan entre si para intercambiar información, servicio o recursos.

La arquitectura cliente / servidor describe como la forma en que se estructura la aplicación, en este sentido permite una división del trabajo en tres áreas especializadas de aplicación.

En el modelo cliente/servidor, se denomina cliente al proceso que inicia el diálogo o solicita los recursos de un determinado servidor, y servidor al proceso que responde a las solicitudes enviadas por el cliente.

Para adoptar una arquitectura Cliente/servidor se poseen las características siguientes:

- La funcionalidad de la aplicación migra hacia todas las computadoras cliente.
- Las funciones de Bases de Datos se le asignan al servidor.

2.9. CALIDAD DEL SISTEMA

Una de las metas principales de la evaluación y comparación de calidad de artefactos Web, radica en medir, analizar y comprender el grado de cumplimiento de un conjunto de características y atributos con respecto a los requerimientos de calidad establecidos, para un perfil de usuario y dominio de aplicación dados. Por una parte, los desarrollos centrados en la Web en los más diversos dominios de aplicación como comercio electrónico, sistemas académicos, financieros, entre otros, se están tomando cada vez más en sistemas complejos. Por lo tanto, desde el punto de vista de la valoración de productos, un proceso de evaluación cuantitativo e integral, que considere la mayoría de las características y atributos relevantes para cierto perfil de usuario, se torna también en una tarea compleja. La complejidad en la evaluación es producto de la gran cantidad de características y atributos que pueden intervenir en los requerimientos de calidad y en las varias relaciones existentes entre los atributos subcaracterísticas y características, entre otros aspectos.

En el proceso de evaluación de requerimientos de calidad de artefactos Web complejos, se observa la necesidad de contar con una metodología cuantitativa, integrada, flexible y robusta, que se apoye en principios y prácticas de Ingeniería de Software para la evaluación y comparación de características y atributos, con el fin de obtener resultados objetivos y justificables.

2.9.1. METODOLOGÍA WEB-SITE QEM

Luís Antonio Olsina propone la Metodología de Evaluación de Calidad de Sitios Web (Web-site QEM) cuyo objetivo es realizar un aporte ingenieril que se adecue a la evaluación, comparación y análisis de calidad de sistemas de información centrados en la Web mas o menos complejos.

Un aspecto importante, que si bien hemos avanzado en parte aún es motivo de futura investigación, reside en determinar un árbol estándar que pueda ser reusado y personalizado para distintos subdominios de aplicaciones, dentro del dominio de la Web. Contar con un árbol de requerimientos general pero a su

vez fácilmente adaptable a diversos dominios, sería de gran beneficio para tareas de aseguramiento y control de calidad posibilitando potencialmente la automatización de las mismas [OLSINA, 1999].

En la siguiente tabla mostramos el árbol de requerimientos de calidad, correspondiente al dominio Web.

1. Usabilidad

- 1.1 Comprensibilidad Global del Sitio
- 1.2 Mecanismos de Ayuda y Retroalimentación en línea
 - 1.2.1 Calidad de la Ayuda
 - 1.2.2 Indicador de Última Actualización
 - 1.2.3 *Facilidad FAQ*
 - 1.2.4 Retroalimentación
 - 1.2.4.1 *Cuestionario*
 - 1.2.4.2 *Libro de Invitados*
 - 1.2.4.3 *Comentarios/Sugerencias*
- 1.3 Aspectos de Interfaces y Estéticos
 - 1.3.1 *Cohesividad al Agrupar los Objetos de Control Principales*
 - 1.3.2 Permanencia y Estabilidad en la Presentación de los Controles Principales
 - 1.3.3 Aspectos de Estilo
 - 1.3.3.1 *Uniformidad en el Color de Enlaces*
 - 1.3.3.2 *Uniformidad en el Estilo Global*
 - 1.3.3.3 *Guía de Estilo Global*
 - 1.3.4 *Preferencia Estética*

2. Funcionalidad

- 2.1 Aspectos de Búsqueda y Recuperación
 - 2.1.1 Mecanismo de Búsqueda en el Sitio Web
 - 2.1.1.1 Búsqueda Restringida
 - 2.1.1.1.1 *de Personas*
 - 2.1.1.1.2 *de Cursos*
 - 2.1.1.1.3 *de Unidades Académicas*
 - 2.1.1.2 *Búsqueda Global*
 - 2.1.2 Mecanismos de Recuperación
- 2.2 Aspectos de Navegación y Exploración
 - 2.2.1 Navegabilidad
 - 2.2.1.1 Orientación
 - 2.2.1.1.1 *Indicador del Camino*
 - 2.2.1.1.2 *Etiqueta de la Posición Actual*
 - 2.2.1.1.2 *Promedio de Enlaces por Página*
 - 2.2.2 Objetos de Control Navegacional
 - 2.2.2.1 Permanencia y Estabilidad en la Presentación de los Controles Contextuales (Subsitio)
 - 2.2.2.2 Nivel de Desplazamiento
 - 2.2.2.2.1 *Desplazamiento Vertical*
 - 2.2.2.2.2 *Desplazamiento Horizontal*
 - 2.2.3 Predicción Navegacional
 - 2.2.3.1 *Enlace con Título (enlace con texto explicatorio)*

- 2.2.3.2 *Calidad de la Frase del Enlace*
- 2.3 Aspectos del Dominio orientados al Estudiante
 - 2.3.1 Relevancia de Contenido
 - 2.3.1.1 Información de Unidad Académica
 - 2.3.1.2 Información de Inscripción
 - 2.3.1.2.1 *Información de los Requerimientos de Ingreso/Admisión*
 - 2.3.1.2.2 *Formulario para Rellenar/Bajar*
 - 2.3.1.3 Información de Carreras
 - 2.3.1.3.3 *Plan de Carrera/Oferta de Cursos*
 - 2.3.1.3.4 Descripción de Cursos
 - 2.3.1.4 Información de Servicios al Estudiante

3. Confiabilidad

- 3.1 No Deficiencia
 - 3.1.1 Errores de Enlaces
 - 3.1.1.1 *Enlaces Rotos*
 - 3.1.1.2 *Enlaces Inválidos*
 - 3.1.1.3 *Enlaces no Implementados*
 - 3.1.2 Errores o Deficiencias Varias
 - 3.1.2.1 *Deficiencias o cualidades ausentes debido a diferentes navegadores (browsers)*
 - 3.1.2.2 *Deficiencias o resultados inesperados independientes de browsers (p.ej. errores de búsqueda imprevistos, deficiencias con marcos (frames), etc.)*
 - 3.1.2.3 *Nodos Destinos (inesperadamente) en Construcción*
 - 3.1.2.4 *Nodos Web Muertos (sin enlaces de retorno)*

4. Eficiencia

- 4.1 Performance
 - 4.1.1 *Páginas de Acceso Rápido*
- 4.2 Accesibilidad
 - 4.2.1 Accesibilidad de Información
 - 4.2.1.1 *Soporte a Versión sólo Texto*
 - 4.2.1.2 Legibilidad al desactivar la Propiedad Imagen del Browser
 - 4.2.1.2.1 *Imagen con Título*
 - 4.2.1.2.2 *Legibilidad Global*
 - 4.2.2 Accesibilidad de Ventanas
 - 4.2.2.1 *Número de Vistas considerando Marcos (frames)*

5. Portabilidad

- 5.1. *Independencia de Hardware*
- 5.2. *Independencia de Software*
- 5.3. *Instalabilidad*

6. Mantenibilidad

- 6.1. *Expandibilidad*
- 6.2. *Estabilidad*

Figura 2.29 Árbol de Requerimientos de Calidad [OLSINA, 1999]

Web-site QEM, incluye un conjunto de fases, actividades, productos, modelos, y algunos constructores intervinientes en el proceso de evaluación, comparación y ranquin de calidad. A continuación se describen las siguientes fases:

- Planificación y Programación de la Evaluación de Calidad
- Definición y Especificación de Requerimientos de Calidad
- Definición e Implementación de la Evaluación Elemental
- Definición e Implementación de la Evaluación Global
- Análisis de Resultados, Conclusión y Documentación
- Validación de Métricas (no mostrada en la figura)

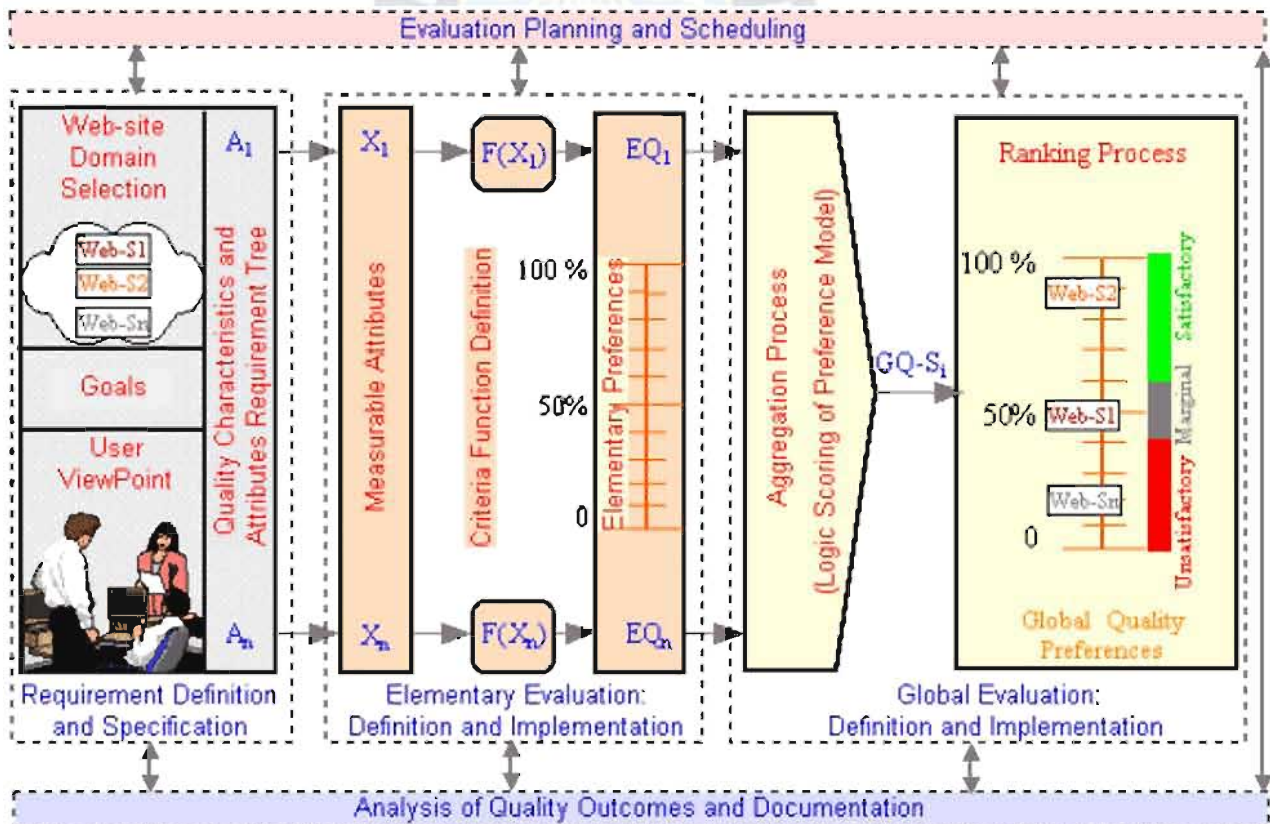


Figura 2.30. Un panorama de los principales módulos intervinientes en el proceso de evaluación y comparación usando Web-site QEM. [OLSINA, 1999]

A continuación se definen y ejemplifican los siguientes procesos de la metodología Web-site QEM, que son parte de algunas de las fases antes mencionadas:

- a) Definiendo el Dominio y Ente para la evaluación de de la calidad
- b) Definiendo metas de Evaluación y Seleccionando el perfil de Usuario.
- c) Especificando Requerimientos de Calidad para artefactos Web.
- d) Definiendo Criterios Elementales e Implementando Procedimientos de Medición
- e) Definiendo las estructuras de agregación e implementando la Evaluación de Calidad Global.
- f) Analizando y comparando los Resultados parciales y globales

2.9.1.1. DEFINIENDO EL DOMINIO Y ENTE PARA LA EVALUACIÓN DE LA CALIDAD

Se define cuál es el dominio de la aplicación a evaluar y definir el ente a evaluar. Tres entes generales que pueden intervenir en procesos de evaluación son: procesos, productos, y recursos. Por ejemplo, para realizar un estudio con el objeto de valorar, comparar y determinar el estado del arte de la calidad sobre características específicas como usabilidad y funcionalidad, se debe seleccionar dentro de un dominio específico un conjunto de sitios o software típicos (artefactos). Dado que las métricas y criterios que utilizaremos para ejemplificar son absolutas, se podría evaluar atributos y características de un solo artefacto.

2.9.1.2. DEFINIENDO METAS DE EVALUACIÓN Y SELECCIONANDO EL PERFIL DE USUARIO

Se definen y refinan las metas y el alcance del proceso de evaluación. Se puede evaluar un proyecto de desarrollo, o un sistema de información Web en la fase operativa, la calidad de un conjunto de características de parte de un sistema, o de un sistema completo, o comparar características y sus preferencias

de calidad global de sistemas comparativos. Los resultados podrían ser utilizados para comprender, mejorar, controlar o predecir la calidad de artefactos Web.

La formulación de metas, las características y atributos de calidad varían conforme al perfil de usuario seleccionado. Considerando estándares como ISO [ISO/IEC 9126], se identifican tres perfiles de usuario, a un alto nivel de abstracción para dominios Web: visitantes, desabolladores, y gerenciadorees. Siguiendo un mecanismo de descomposición, podemos dividir a la categoría visitante en clases más específicas.

2.9.1.3. ESPECIFICANDO REQUERIMIENTOS DE CALIDAD PARA ARTEFACTOS WEB

Se especifican las características, subcaracterísticas y atributos de calidad agrupándolas en un árbol de requerimientos. Respecto de las características de calidad de más alto nivel, se sigue la misma clasificación conceptual que la prescrita en el estándar ISO [ISO/IEC 9126], y la casi idéntica clasificación dada por [IEEE Std 1061], en su Anexo informativo A, (donde le llaman factores). Estas características de alto nivel son: usabilidad, funcionalidad, confiabilidad, eficiencia, portabilidad, y mantenibilidad.

Se puede cuantificar a la calidad de un producto de software, por la apropiada agregación y cuantificación de algunas o todas esas características, a partir de la medición directa e indirecta de atributos y la posterior agregación. De modo que, a partir de esas características, se derivan subcaracterísticas, y, a partir de éstas, siguiendo un proceso de descomposición recursivo, se pueden especificar atributos. Finalmente, considerando el dominio, las metas y el perfil de usuario (sus necesidades y comportamiento), se debe especificar el árbol de requerimientos de calidad. Técnicas de elicitación de requerimientos como cuestionarios y encuentros entre los participantes deben ser planificadas.

2.9.1.3.1. ENFOQUES DE MODELO DE CALIDAD.

Se definen los siguientes enfoques de modelos de calidad:

- El enfoque de modelo fijo de calidad: en la que se asume que todas las características (o factores) y demás componentes necesarios para monitorear un proyecto, son un subconjunto de aquellos publicados en los modelos bien conocidos como el de McCall, Boehm, ISO 9126 e IEEE 1061.
- El enfoque "Defina su propio modelo de calidad": en la que se acepta la concepción general que la calidad está compuesta de muchos características y atributos, pero en donde no se adopta un modelo de calidad establecido. El modelo se define por consenso con todas las partes intervinientes, para un proyecto y producto dado.
- La noción de un enfoque de modelo de calidad mixto es a la vez, pragmático y flexible. Se parte de un modelo fijo (en este caso, a. partir del modelo de calidad (ISO 9126) en la que se asume que todas las características necesarias para monitorear un proyecto de evaluación de calidad de producto, son un subconjunto de las seis características publicadas en el estándar: usabilidad, funcionalidad, confiabilidad, eficiencia, portabilidad, y mantenibilidad.

Se definen las subcaracterísticas de niveles inferiores, los atributos, (los criterios de medición) y las relaciones entre atributos, subcaracterísticas y características entre las partes itervinientes.

2.9.1.4. DEFINIENDO CRITERIOS ELEMENTALES E MPLEMENTANDO PROCEDIMIENTOS DE MEDICIÓN

Se define una base de criterios para la evaluación elemental; realizar el proceso de medición, y puntaje elemental. Un criterio de evaluación elemental declara y especifica cómo medir atributos cuantificables. El resultado final es una preferencia o ranquin elemental, el cual puede ser interpretado como el grado o porcentaje del requerimiento elemental satisfecho.

A partir del árbol de requerimientos, para cada atributo cuantificable A_i (u hoja del árbol) se asocia una variable X_i , que tomará un valor real a partir de un proceso de medición.

Para cada variable medida X_i , $i = 1, \dots, n$ se define una función que representa al criterio elemental y la denominamos preferencia de calidad elemental o indicador elemental (IE_i). Se puede asumir, a IE_i como el porcentaje de requerimiento satisfecho para A_i . El valor de preferencia caerá en uno de los tres niveles de aceptabilidad, esto es:

- insatisfactorio (de 0 a 40%),
- marginal (desde 40 a 60%),
- y satisfactorio (desde 60 a 100%)

El valor de la preferencia de calidad elemental es también un número real pero perteneciente al intervalo unitario I , de manera que:

$$IE_i \in I, i = 1, \dots, n, I = [0, 1]$$

Desde un punto de vista analítico, el criterio elemental se define como la función:

$$F_i : R_i \rightarrow I \quad \text{en donde} \quad IE_i = F_i(X_i), \quad X_{i \min} \leq X_i \leq X_{i \max}$$

2.9.1.4.1. REPRESENTACIÓN NOTACIONAL DE LOS CRITERIOS

Se pueden identificar al menos cuatro tipos diferentes de notaciones para representar a los criterios elementales:

- Notación gráfica (de líneas, barras, etc.)
- Notación en escala de preferencia
- Notación de los puntos de coordenadas relevantes
- Notación analítica

En la presente evaluación se emplea la notación de escala de preferencias para documentar a los atributos, debido a su poder visual y al espacio reducido que ocupa. En ésta se muestran los puntos de mayor interés y su preferencia correspondiente.

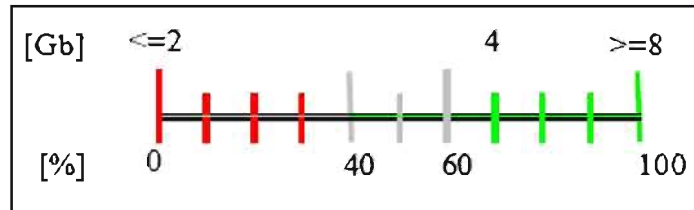


Figura 2.31. Notación de Escala de Preferencia
[OLSINA, 1999]

2.9.1.4.2. TIPOS DE CRITERIOS DE PREFERENCIA DE CALIDAD ELEMENTAL

Dos tipos básicos de criterios elementales son los absolutos y los relativos, y, dentro de los primeros se pueden descomponer en criterios con variables continuas, y criterios con variables discretas. La siguiente figura muestra una jerarquía detallada de los tipos de criterios.

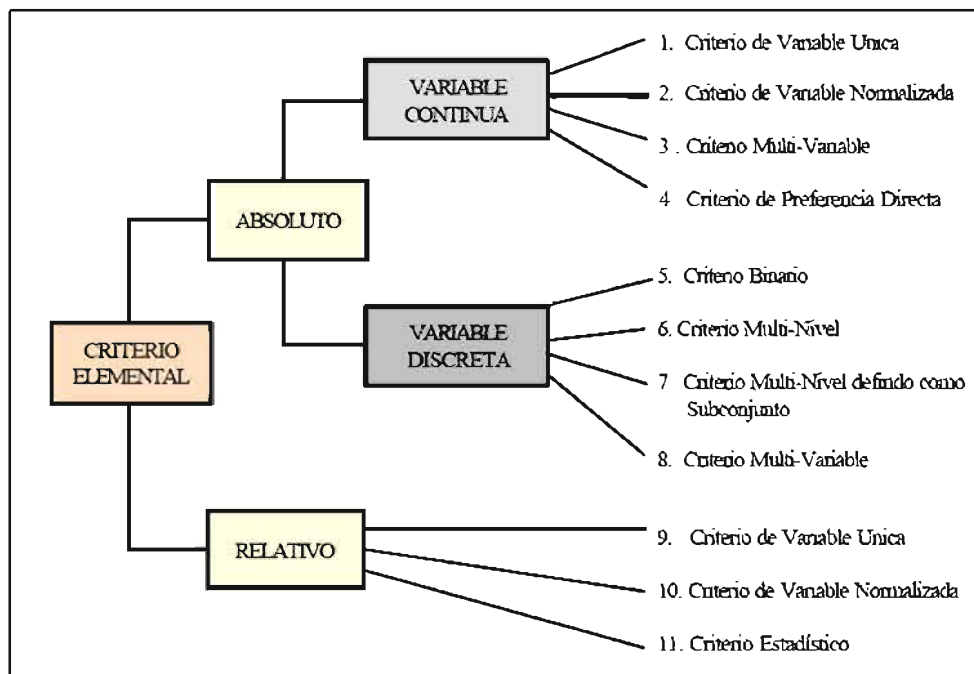


Figura 2.32. Taxonomía de tipos de criterio elementales
[OLSINA, 1999]

Un criterio de evaluación elemental absoluto es aquél que se emplea para determinar la preferencia absoluta de un atributo de un artefacto, y que no está relacionado con indicadores de otros sistemas comparativos. Un criterio absoluto se diferencia de uno relativo en que la meta de este último consiste solamente en la determinación de los indicadores relativos de los sistemas comparados sin evaluar la calidad de cada sistema de un modo individual e independiente.

Las siguientes tablas muestran tres tipos de plantillas: para el componente característica, subcaracterística, y para el atributo (elemento de más bajo nivel en la jerarquía). [OLSINA, 1999]

Título	Código
Tipo: Característica	
Sub-característica/s (Código/s):	
Definición/Comentarios:	
Modelo para determinar el Computo Global	
Herramienta Empleada:	
Peso:	
Operador Aritmético/Lógico:	
Ejemplo/s	
Valor/es de Preferencia/s Computado/s:	

Tabla 2.9. Plantilla para especificar a una característica de más alto nivel [OLSINA, 1999]

Título	Código
Tipo: Atributo	
Característica de mas alto nivel (Código):	
Super-característica (Código):	
Definición/Comentarios:	
Tipo de Criterio Elemental:	
Plantilla de Referencia de Variables y Parámetros:	
Escala de Preferencia:	
Tipo de Recolección de Datos:	
Herramienta Empleada:	
Ejemplo/s	
Valor/es de Preferencia/s Computado/s:	

Tabla 2.10. Plantilla para especificar a un atributo. [OLSINA, 1999]

Título:	Título:	Tipo: Subcaracterística
Super-característica (Código):		
Sub-característica (Código):	Atributo/s (Código/s)	
Definición/Comentarios:		
Modelo para determinar el Computo Parcial:	Herramienta Empleada:	
Peso:	Operador Aritmético/Lógico:	
Ejemplo/s:	Valor/es de Preferencia/s Computado/s:	

Tabla 2.11. Plantilla para especificar a una subcaracterística. [OLSINA, 1999]

2.9.1.5. DEFINIENDO LAS ESTRUCTURAS DE AGREGACIÓN E IMPLEMENTANDO LA EVALUACIÓN DE CALIDAD GLOBAL

Se establecen estructuras de agregación de preferencias elementales para producir la preferencia de calidad global. Posteriormente, se debe implementar el proceso de evaluación de modo de obtener un indicador de calidad global para cada sistema evaluado. Por lo tanto, aplicando un mecanismo de agregación paso a paso, las preferencias elementales se pueden agrupar convenientemente para producir al final el resultado global.

El criterio de agregación tiene como entrada dos indicadores elementales $CrE(X_1)$, y $CrE(X_2)$. Esto es, para cada par de valores de las variables X_1 y X_2 se generan dos preferencias o indicadores elementales correspondientes: IE_1 e IE_2 . El problema consiste ahora en expresar la preferencia o indicador global IG_1 como una función de agregación de los indicadores elementales. Como muestra la siguiente figura.

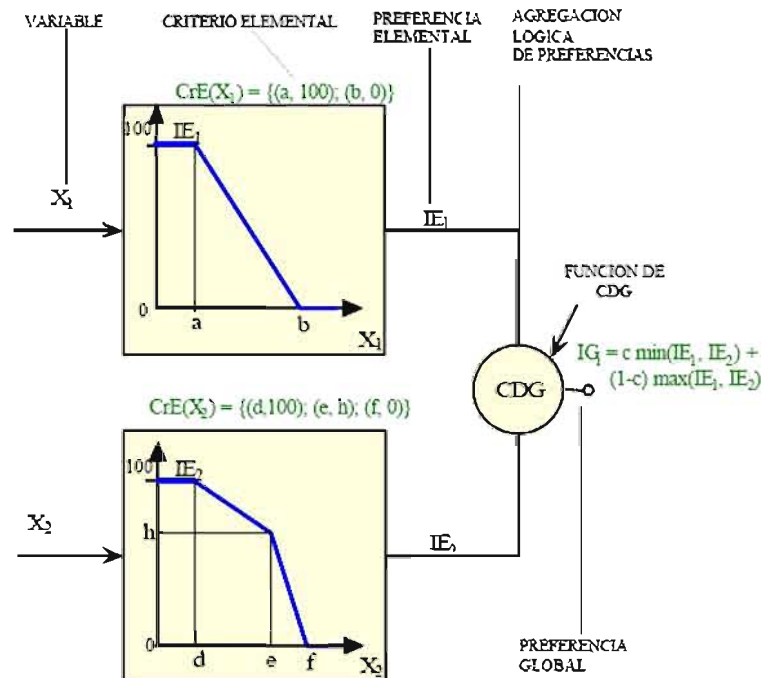


Figura 2.33. Panorama del proceso de determinación de la preferencia de calidad global a partir de preferencias elementales. [OLSINA, 1999]

Una propiedad básica de dicha función es la siguiente desigualdad:

$$\text{Min}(IE_1, IE_2) \leq IG_1(IE_1, IE_2) \leq \text{Max}(IE_1, IE_2)$$

En donde,

$$\text{Min}(IE_1, IE_2) = IE_1 \cdot IE_2 ; \quad \text{esto es, la conjunción ;}$$

$$\text{Max}(IE_1, IE_2) = IE_1 + IE_2 ; \quad \text{esto es, la disyunción}$$

INTERPRETACIÓN: la preferencia global no puede ser más preferida (o mejor) que su parte más preferida (la mejor), ni puede ser menos preferida (o peor) que su parte menos preferida (la peor).

2.9.1.5.1. GRADOS DE CONJUNCIÓN Y DISYUNCIÓN

El indicador resultante IG_1 debe estar entre alguno de los valores extremos de IE_1 e IE_2 . Con el fin de ajustar las propiedades lógicas deseadas de la función de agregación de preferencias, se introduce el concepto de un parámetro ajustable c denominado el grado de conjunción (como así también el parámetro ajustable d denominado el grado de disyunción). El mismo especifica la posición de IG_1 con respecto del $\text{Min}(IE_1, IE_2)$ y el $\text{Max}(IE_1, IE_2)$.

Si $c=1$ (y $d=0$), entonces

IG_1 representa el valor resultante de una función de conjunción pura; si $c=0$ (y $d=1$), entonces IG_1 representa el valor resultante de una función de disyunción pura ($IG_1 = \text{Max}(IE_1, IE_2)$). Finalmente,

Si $0 < c < 1$ entonces

$$\text{Min}(IE_1, IE_2) < IG_1(IE_1, IE_2) < \text{Max}(IE_1, IE_2).$$

La función de agregación ajustable que tiene esas propiedades se denomina "Función de Conjunción-Disyunción Generalizada" (CDG).

Algunas propiedades deseables de la función CDG deben ser:

- Debe ser posible agrupar más de dos indicadores o preferencias elementales
- Debe proveer un modelo para representar requerimientos mandatorios (si el requerimiento obligatorio no es satisfecho, la función debe producir una referencia parcial, o global de cero)
- Debe proveer un nivel ajustable de importancia relativa (o peso) de cada preferencia de entrada

Las propiedades antes mencionadas de la función CDG se pueden satisfacer por medio de la media de potencia pesada, que se expresa del siguiente modo:

$$IG(r) = (P_1IE_1^r + P_2IE_2^r + \dots + P_mIE_m^r)^{1/r} ; \quad -\infty \leq r \leq +\infty ; \quad 0 \leq IE_i \leq 1 ;$$

$$(P_1 + P_2 + \dots + P_m) = 1 ; \quad P_i > 0 ; \quad i = 1 \dots m ;$$

La ubicación de $IG(r)$ entre el valor mínimo y máximo se puede ajustar seleccionando el valor de r . Si el valor de $IG(r)$ está más cercano al mínimo entonces tal criterio especifica el requerimiento para la simultaneidad de entradas (solamente altos valores en las entradas producen un alto valor en la salida). Si el valor de $IG(r)$ está más cercano al máximo, entonces tal criterio especifica el requerimiento para la reemplazabilidad de entradas (un alto valor en una de las entradas produce un alto valor en la salida).

2.9.1.5.2. EMPLEO DE LOS OPERADORES DE LSP PARA MODELAR RELACIONES LÓGICAS ENTRE CARACTERÍSTICAS Y ATRIBUTOS

Las principales relaciones lógicas se definen del siguiente modo:

- Simultaneidad (o relación de conjuntividad): cuando los participantes en el proceso de evaluación perciben que dos o más entradas deben estar presentes simultáneamente
- Reemplazabilidad (o relación de disyuntividad): cuando los participantes en el proceso de evaluación perciben que dos o más entradas puede estar presentes alternativamente (por ej., la presencia de un atributo puede reemplazar a la ausencia de otro)

- Neutralidad (o relación ni de conjuntividad ni de disyuntividad): cuando se percibe que dos o más preferencias de entrada pueden agruparse de un modo independiente
- Relación Simétrica: cuando se percibe que dos o más preferencias de entrada afectan de la misma manera lógica aunque con diferentes grados de importancia
- Relación Asimétrica: cuando se requiere modelar requerimientos mandatorios combinados con requerimientos no-mandatorios (atributos obligatorios se combinan con otros deseables y/u opcionales), o cuando condiciones necesarias se combinan con condiciones suficientes.

La figura 2.34 muestra un modelo de 17 niveles de operadores o conectores lógicos que representan a la función CDG

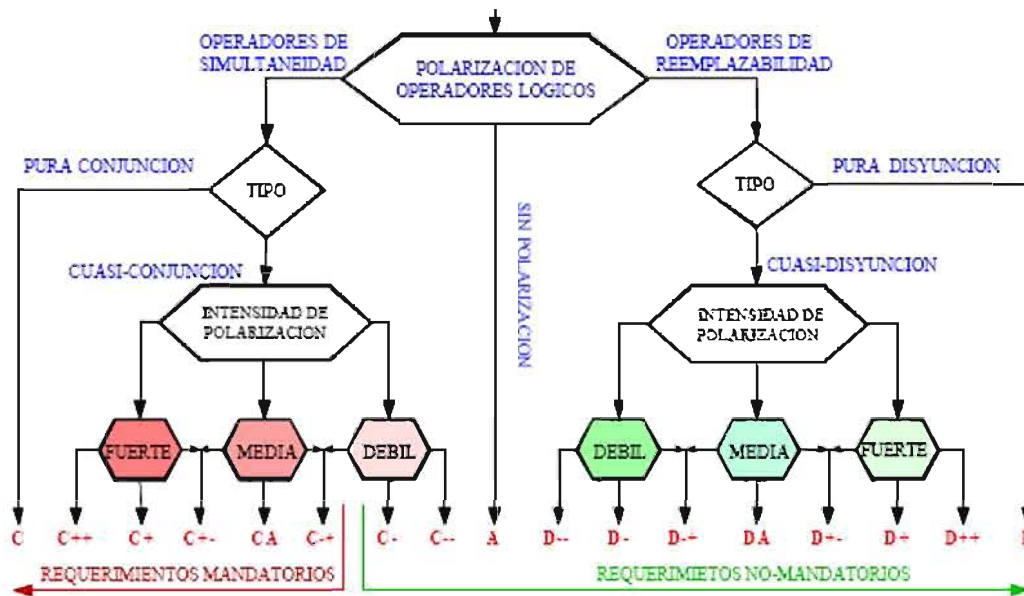


Figura 2.34. Operadores lógicos Conjuntivos y Disyuntivos de LSP y niveles de Polarización. [OLSINA, 1999]

Los principales operadores de LSP son la media aritmética (A) que modela la relación de neutralidad. Dentro de los operadores conjuntivos encontramos el operador (C) que modela a la conjunción pura, y dentro de las funciones de la cuasi-conjunción se encuentran tres niveles de intensidad: débil (C-), medio (CA),

y fuerte (C+). Es importante destacar, que los operadores lógicos de cuasi-conjunción representan conectores "y" flexibles. Además de los conectores de cuasi-conjunción descriptos, se cuenta con operadores de valores intermedios; por ejemplo, el operador C—se posiciona entre A y C- y el operador C-+ está entre CA y C-, y así sucesivamente

2.9.1.5.3. TIPOS DE FUNCIONES DE AGREGACIÓN

A partir de la combinación de los operadores lógicos descritos en la sección previa, las funciones de agregación de preferencia se pueden clasificar en simples y compuestas.

Las funciones de agregación simples modelan relaciones de entradas simétricas, en tanto que las funciones de agregación compuestas modelan relaciones de entrada asimétricas.

La figura 2.35 observamos tres ejemplos del primer tipo. El diagrama de la izquierda representa a una función de tres entradas a un conector de cuasi-conjunción Se utiliza el operador C-+ (que modela requerimientos mandatorios); asimismo cada entrada tiene su respectivo peso. En el diagrama del centro se muestra a una función de neutralidad con tres entradas, utilizando el operador A, que modela requerimientos independientes de un grupo. Por último, el dibujo de la derecha representa una función de tres entradas, con sus respectivos pesos, a una función de cuasi-disyunción utilizando el operador DA, que modela requerimientos alternativos de un grupo de preferencias.

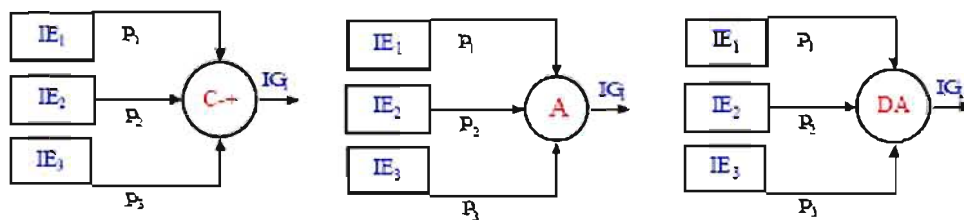


Figura 2.35. Tres funciones simples de agregación de preferencias
[OLSINA, 1999]

2.9.1.5.4. CONSIDERACIONES SOBRE LAS ACTIVIDADES EN EL PROCESO DE AGREGACIÓN

En el proceso de agregación de las preferencias elementales, parciales y global se identifican las siguientes actividades básicas:

- Selección del tipo de relación lógica entre elementos y/o componentes: es decir, la cuestión básica a considerar consiste en preguntarse: ¿es la relación entre las entradas con una polarización lógica conjuntiva, disyuntiva o neutra?
- Selección del tipo de función conforme a la relación de entradas entre elementos y/o componentes: es decir, la cuestión básica a considerar consiste en decidir ¿es la relación de entradas simétrica o asimétrica?
- Selección del operador conforme al nivel de intensidad de la polarización lógica
- Selección de la importancia relativa de cada entrada conforme a los requerimientos (esto es, la selección de los pesos)

2.9.1.6. ANALIZANDO Y COMPARANDO LOS RESULTADOS PARCIALES Y GLOBALES

Se analizan, evalúan y comparan resultados parciales y globales considerando las metas y la vista de usuario establecidas. Este proceso culmina con la documentación de las conclusiones y recomendaciones. El proceso de evaluación, por medio de la metodología Web-site QEM, produce información elemental, parcial y global. La misma puede ser fácilmente analizada por medio de un modelo de seguimiento o trazabilidad, esto es, seguimiento hacia atrás (backward) y hacia delante (forward), y eficientemente empleada en actividades de toma de decisión.

2.10 ESTIMACIÓN DEL COSTO DEL PROYECTO

Existen una serie de métricas propuestas por la **Ingeniería del Software** para determinar el esfuerzo de un proyecto, el alcance del mismo y la productividad de sus programadores. Vamos a aplicar algunas de las mismas a este desarrollo, para calibrar su dificultad y rendimiento obtenido.

Las métricas orientadas a tamaño

La métrica del software es un factor realmente importante en el análisis de un proyecto. Las métricas orientadas al tamaño proporcionan medidas directas del software y del proceso por el cual se desarrolla. Se basan en la medición del número de Líneas De Código - LDC - que contiene el desarrollo, entendiendo por línea de código una sentencia del lenguaje de programación (se excluyen comentarios y líneas en blanco de los fuentes). Una forma de clasificarlos es atendiendo al número de líneas de código, como se muestra en la tabla:

Categoría	Programadores	Duración	Líneas de código	Ejemplo
Trivial	1	0 - 4 semanas	< 1k	Utilidad de ordenación
Pequeño	1	1 - 6 meses	1k - 3k	Biblioteca de funciones
Media	2 - 5	0,5 - 2 años	3k - 50k	Compilador de C
Grande	5 - 20	2 - 3 años	50k - 100k	SO pequeño
Muy grande	100 - 1000	4 - 5 años	100k - 1M	Grandes SO
Gigante	1000-5000	5 - 10 años	> 1M	Sistema de Distribución

Tabla 2.12 Categoría de un proyecto en función de sus líneas de código.
[OLSINA, 1999]

SpiderBot ha generado más de 8.000 líneas de código, con lo que nos enfrentamos a un proyecto software con una clasificación de complejidad media, para el cual se necesitarían de 2 a 5 programadores trabajando de medio año a 2 años.

2.10.1 EL MÉTODO COCOMO II

Una metodología que se encarga de medir proyectos software es COCOMO. La metodología COCOMO (CONstructive COSt MOdel) se debe a Barry Boehm, y está orientada a líneas de código.

Hay una jerarquía de modelos COCOMO: básico, intermedio y avanzado, la cual se aplica a tres tipos diferentes de software:

1. **Orgánico:** proyectos relativamente sencillos, menores de 50.000 líneas de código. Se tiene experiencia en proyectos similares y se encuentra en un entorno estable.
2. **Semiacoplado:** proyectos intermedios en complejidad y tamaño. La experiencia en este tipo de proyectos es variable, y las restricciones intermedias.
3. **Empotrado:** proyectos bastante complejos, en los que apenas se tiene experiencia y en un entorno de gran innovación técnica. Se trabaja con unos requisitos muy restrictivos y de gran volatilidad.

Dado que sólo se va a emplear una variable para la estimación (la línea de código), se empleará COCOMO básico, ya que es un modelo uní variable estático, con lo que se obtiene una **valoración objetiva** del esfuerzo realizado. Este proyecto será considerado como software orgánico, ya que posee menos de 50.000 líneas de código.

La ecuación del esfuerzo de COCOMO básico tiene la siguiente forma:

$$E = \text{Esfuerzo} = a KLDC^b \text{ (persona x mes)}$$

Donde *KLDC* es el número de líneas de código, distribuidas en millares, para el proyecto.

La ecuación del tiempo de desarrollo es:

$$T = \text{Tiempo de duración del desarrollo} = c \text{ Esfuerzo}^d \text{ (meses)}$$

Por su parte los coeficientes a, b, c y d se obtienen empíricamente del estudio de una serie de proyectos, y sus valores son:

Proyecto de software	a	b	c	d
Orgánico	2,4	1,05	2,5	0,38
Semiacoplado	3,0	1,12	2,5	0,35
Empotrado	3,6	1,20	2,5	0,32

Tabla 2.13. Coeficientes COCOMO

En el desarrollo de SpiderBot se han codificado 8,2 miles de líneas de código.

$$\text{Esfuerzo realizado} = 2,4 * 8,2^{1,05} = 21,9 \text{ personas-mes}$$

$$T = 2,5 * 21,9^{0,38} = 8,1 \text{ mes}$$

$$\text{Nº de personas para desarrollar el proyecto} = E/T = 21,9 / 8,1 \approx 3 \text{ personas}$$

La controversia: Líneas de código frente a puntos de función

Existe en el mundo de la Ingeniería del Software una viva polémica sobre qué tipo de métricas son mejores para evaluar un proyecto: las orientadas a tamaño o las que utilizan *puntos de función*.

El centro de controversia está en considerar las líneas de código como medida clave, ya que los que se oponen a su uso, aducen que las medidas basadas en líneas de código son dependientes del lenguaje de programación.

En cualquier caso esta polémica queda apartada gracias a Casper Jones, quién creó la siguiente tabla 2.14 de correspondencia entre algunos de los lenguajes de programación más conocidos con su número de equivalencia entre líneas de código por punto de función:

Lenguaje	<i>LDC/PF</i>
Ensamblador	320
C	150
Cobol	106
Pascal	91
Basic	64
TCL	64
Java	53
C++	29

Tabla 2.14 Conversión Líneas de código a Puntos de función

2.11. HERRAMIENTAS DE CONSTRUCCIÓN DE SOFTWARE

Existen diversas alternativas para el desarrollo de aplicaciones Web: PHP, páginas ASP, ColdFusion, J2EE, NET etc. y la mayoría de ellas están actualmente en un estado tecnológico excelente, pero hay algo que las diferencia y es su orientación hacia un modelo de desarrollo de componentes empresarial. Esto quiere decir que la plataforma tecnológica tiene que servir y propiciar, desde el entorno de ejecución hasta el entorno de desarrollo. Estos servicios pueden ser un sistema de autenticación y seguridad, un entorno transaccional, soporte para servidores, instanciación remota de componentes, Framework de acceso a soportes de nuevas tecnologías como WebServices, XML, etc.

2.11.1. PHP

PHP es un lenguaje de programación usado frecuentemente para la creación de contenido para sitios web con los cuales se puede programar las páginas html y los códigos de fuente. PHP es un acrónimo recursivo que significa "PHP Hypertext Pre-processor" (inicialmente PHP Tools, o, Personal Home Page Tools), y se trata de un lenguaje interpretado usado para la creación de aplicaciones para servidores, o creación de contenido dinámico para sitios web.

El fácil uso y la similitud con los lenguajes más comunes de programación estructurada, como C y Perl, permiten a la mayoría de los programadores experimentados crear aplicaciones complejas con una curva de aprendizaje muy suave. También les permite involucrarse con aplicaciones de contenido dinámico sin tener que aprender todo un nuevo grupo de funciones y prácticas.

Su interpretación y ejecución se da en el servidor web, en el cual se encuentra almacenado el script, y el cliente sólo recibe el resultado de la ejecución. Cuando el cliente hace una petición al servidor para que le envíe una página web, generada por un script PHP, el servidor ejecuta el intérprete de PHP, el cual procesa el script solicitado que generará el contenido de manera dinámica, pudiendo modificar el contenido a enviar, y regresa el resultado al servidor, el cual se encarga de regresárselo al cliente. Además es posible utilizar PHP para

generar archivos PDF, Flash, así como imágenes en diferentes formatos, entre otras cosas.

Permite la conexión a diferentes tipos de servidores de bases de datos tales como MySQL, Postgres, Oracle, ODBC, DB2, Microsoft SQL Server, Firebird y SQLite; lo cual permite la creación de Aplicaciones web muy robustas.

PHP también tiene la capacidad de ser ejecutado en la mayoría de los sistemas operativos tales como UNIX (y de ese tipo, como Linux), Windows y Mac OS X, y puede interactuar con los servidores de web más populares ya que existe en versión CGI, módulo para Apache, e ISAPI.

La última versión a Mayo 2007 es la 5.2.2. Los principales usos del PHP son los siguientes:

- Programación de páginas web dinámicas, habitualmente en combinación con el motor de base datos MySQL, aunque cuenta con soporte nativo para otros motores, incluyendo el estándar ODBC, lo que amplía en gran medida sus posibilidades de conexión.
- Creación de aplicaciones gráficas independientes del navegador, por medio de la combinación de PHP y GTK (GIMP Tool Kit), lo que permite desarrollar aplicaciones de escritorio en los sistemas operativos en los que está soportado.
- Es un lenguaje multiplataforma.
- Capacidad de conexión con la mayoría de los manejadores de base de datos que se utilizan en la actualidad, destaca su conectividad con MySQL
- Leer y manipular datos desde diversas fuentes, incluyendo datos que pueden ingresar los usuarios desde formularios HTML.
- Capacidad de expandir su potencial utilizando la enorme cantidad de módulos (llamados ext's o extensiones).
- Posee una amplia documentación en su página oficial, entre la cual se destaca que todas las funciones del sistema están explicadas y ejemplificadas en un único archivo de ayuda.

- Es libre, por lo que se presenta como una alternativa de fácil acceso para todos.
- Permite las técnicas de Programación Orientada a Objetos.
- Permite crear los formularios para la web.
- Biblioteca nativa de funciones sumamente amplia e incluida

No requiere definición de tipos de variables ni manejo detallado del bajo nivel.

2.11.2. MySQL

MySQL es un sistema de gestión de base de datos, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB desarrolla MySQL como software libre en un esquema de licenciamiento dual. Está desarrollado en su mayor parte en ANSI C.

MySQL es software de fuente abierta. Fuente abierta significa que es posible para cualquier persona usarlo y modificarlo. Cualquier persona puede bajar el código fuente de MySQL y usarlo sin pagar. Cualquier interesado puede estudiar el código fuente y ajustarlo a sus necesidades. MySQL usa el GPL (GNU General Public License) para definir que puede hacer y que no puede hacer con el software en diferentes situaciones. Si usted no se ajusta al GLP o requiere introducir código MySQL en aplicaciones comerciales, usted puede comprar una versión comercial licenciada.

MySQL es un sistema de administración de bases de datos. Esta puede ser desde una simple lista de compras a una galería de pinturas o el vasto volumen de información en un red corporativa. Para agregar, acceder a y procesar datos guardados en un computador, usted necesita un administrador como MySQL Server.

MySQL es muy utilizado en aplicaciones web como MediaWiki o Drupal, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación web está muy ligada a PHP, que a menudo aparece en combinación con MySQL. MySQL es una base de datos muy rápida en la lectura cuando utiliza

el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones.

Características de la versión 5.2.2

- Un amplio subconjunto de ANSI SQL 99, y varias extensiones.
- Soporte a multiplataforma
- Procedimientos almacenados
- Triggers
- Cursors
- Vistas actualizables
- Soporte a VARCHAR
- Query caching
- Sub-SELECTs (o SELECTs anidados)
- Soporte completo para Unicode
- Conforme a las reglas ACID usando los motores InnoDB, BDB y Cluster
- Usa tablas en disco b-tree para búsquedas rápidas con compresión de índice
- Tablas hash en memoria temporales
- Completo soporte para operadores y funciones en cláusulas select y where.
- Completo soporte para cláusulas group by y order by, soporte de funciones de agrupación
- Seguridad: ofrece un sistema de contraseñas y privilegios seguro mediante verificación basada en el host y el tráfico de contraseñas está cifrado al conectarse a un servidor.
- Soporta gran cantidad de datos. MySQL Server tiene bases de datos de hasta 50 millones de registros.
- Se permiten hasta 64 índices por tabla (32 antes de MySQL 4.1.2). Cada índice puede consistir desde 1 hasta 16 columnas o partes de columnas. El máximo ancho de límite son 1000 bytes (500 antes de MySQL 4.1.2).

- Los clientes se conectan al servidor MySQL usando sockets TCP/IP en cualquier plataforma. En sistemas Windows se pueden conectar usando named pipes y en sistemas Unix usando ficheros socket Unix.
- En MySQL 5.0, los clientes y servidores Windows se pueden conectar usando memoria compartida.

2.11.3. SERVIDOR HTTP APACHE

El **servidor HTTP Apache** es un software (libre) servidor HTTP de código abierto para plataformas Unix (BSD, GNU/Linux, etc.), Windows , Macintosh y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. Cuando comenzó su desarrollo en 1995 se basó inicialmente en código del popular NCSA HTTPd 1.3, pero más tarde fue reescrito por completo. Su nombre se debe a que originalmente Apache consistía solamente en un conjunto de parches a aplicar al servidor de NCSA. Era, en inglés, *a patchy server* (un servidor "parcheado").

Apache presenta entre otras características mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido, pero fue criticado por la falta de una interfaz gráfica que ayude en su configuración.

Apache tiene amplia aceptación en la red: en el 2005, Apache es el servidor HTTP más usado, siendo el servidor HTTP del 70% de los sitios web en el mundo y creciendo aún su cuota de mercado (estadísticas históricas y de uso diario proporcionadas por Netcraft).

2.11.4. INSTALACIÓN DE PHP5 CON WAMP5

Existe un paquete de instalación llamado WAMP5 que puede instalar en conjunto Apache, PHP 5, MySQL y PHPMyAdmin.

Existe una manera de comenzar a utilizar PHP5 en Windows sin tener que sufrir las complicaciones típicas de la instalación de los servidores necesarios para programar en PHP. Se trata de instalar un paquete llamado WAMP, que permite instalar y configurar en un solo proceso el servidor Apache, la base de datos MySQL y el módulo de programación en PHP versión 5.

WAMP es un sistema indicado para los usuarios que no tienen instalado en el sistema ninguno de los programas necesarios para programar en PHP (Apache, PHP y MySQL), ya que realiza una instalación completa y desde cero. Pero también pueden utilizar este programa los usuarios que disponen de Apache, PHP y/o MySQL en su sistema. En cuyo caso, simplemente se realizará otra copia de las aplicaciones en un directorio distinto, que en principio, no tiene por qué interferir con las otras instalaciones alojadas en nuestro equipo.

El software que se instala con WAMP5 contiene los siguientes servidores y programas:

- Apache 1.3.31. El servidor de páginas web más extendido del mercado. Aunque la última versión de este servidor es Apache 2, se instala una versión anterior que resulta más estable. Existe un Add-on que permite sustituir la versión 1.3.31 de Apache por la última versión.
- PHP5. El motor renovado del lenguaje.
- MySQL. La base de datos más extendida para utilizar con PHP.
- PHPmyadmin. Un software que permite administrar una base de datos a través de una interfaz web.
- SQLitemanager. Un sistema para administrar una base de datos a partir de sentencias SQL.

Cuando instalamos WAMP5 se crea un grupo de programas llamado WampServer, donde podremos encontrar una opción que pone "Start

Wampserver", que será necesario ejecutar si no hemos seleccionado que el servidor se inicie automáticamente.

Podemos probar si los servicios están corriendo perfectamente accediendo a la página de inicio del servidor, escribiendo en la barra de direcciones de nuestro navegador algo como <http://localhost/>.



CAPÍTULO III



DESARROLLO DEL SISTEMA

ANÁLISIS Y DISEÑO



CAPÍTULO III

DESARROLLO DEL SISTEMA ANÁLISIS Y DISEÑO

3.1. INTRODUCCIÓN

El proceso de Desarrollo del sistema presenta tareas adicionales, como el de realizar una investigación exhaustiva de todos los procesos realizados en el Área académica del INSSB, además de las funciones que cada Unidad debe cumplir. En este Capítulo se desarrollará las fases descritas detalladamente en el Marco Teórico: Planeación y elaboración, Construcción y Aplicación. En los distintos flujos de trabajo del proceso según Craig Larman.

3.2. PROCESO DE DESARROLLO.

Para el Proceso de Desarrollo se realizó en una primera etapa la recolección de información a través de entrevistas, estudio de manuales (funciones y procedimientos), consultas a documentación existente (formatos de registro personal, matriculas, boletas de inscripción, boletines, etc.), en una segunda etapa se vio conveniente observar y tomar nota de los proceso de matriculación e inscripción de estudiantes durante las gestiones II/2006 y I/2007 para conocer mejor como se realiza el tratamiento de la información de cada estudiante, además que es necesario realizar un análisis de las unidades académicas del INSSB y de las diferentes especialidades que ofrece a la comunidad estudiantil.

3.2.1. ANÁLISIS ACTUAL DEL INSTITUTO NORMAL SUPERIOR SIMON BOLIVAR

Inicialmente el INSSB contaba solo con tres Niveles de formación: Inicial, Primaria y Secundaria. Mediante resolución del Ministerio de Educación se propuso implementar a partir de esta gestión (I/2007), un nuevo nivel que contemplara especialidades técnicas, quedando las carreras y especialidades conformadas de la siguiente manera:



Figura 3.1. Especialidades por carrera que ofrece el INSSB
Fuente: Elaboración Propia

3.2.1.1. ESTRUCTURA ORGANIZACIONAL.

La estructura organizacional esta conformada de la siguiente manera:

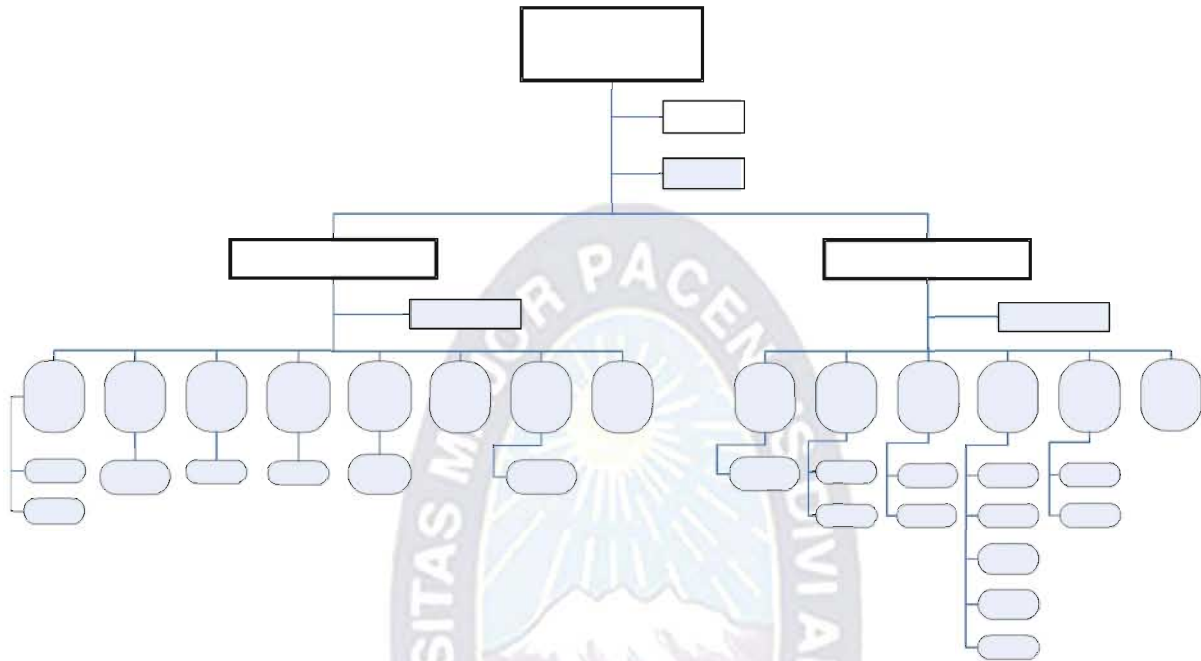


Figura 3.2. Organigrama del Instituto Normal Superior Simón Bolívar [LOZADA, 2003]

Por medio de las técnicas de recolección de datos anteriormente mencionadas, se identificaron a los siguientes usuarios involucrados en el Área Académica con sus respectivas funciones.

- **DIRECTOR GENERAL**

Objetivo General. Planear, organizar, ejecutar y controlar las actividades académicas y administrativas.

Funciones específicas.- Impartir instrucciones precisas al personal para el efectivo cumplimiento de sus actividades, verificar y firmar toda la documentación que compete a su unidad, proponer y promover las políticas académicas y administrativas, representar a la institución en todos los eventos que le compete.

- **DIRECTOR ACADEMICO**

Objetivo General.- Planear, organizar y ejecutar las actividades en base a los lineamientos generales de la Institución

Funciones específicas.- Planificar y programar las actividades de enseñanza y aprendizaje, llevar el kardex de avance académico de los estudiantes, homologar materias en coordinación con los jefes de área, coordinar la inscripción de materias, supervisar la forma de evaluación académica de los estudiantes, cumplir con otras tareas inherentes a su cargo.

- **RESPONSABLE DE ARCHIVO KARDEX**

Objetivo General.- Manejo y control de Archivo Kardex de una unidad académica, apoyando en las labores auxiliares en el campo académico.

Funciones específicas.- Organizar y controlar el Archivo y Kardex de la unidad a fin de dar celeridad y fluidez a los diferentes requerimientos de trámites, certificación e información que provengan de las unidades del INSSB y de parte de los estudiantes. Registra y archiva los datos de las actas oficiales de cada gestión académica, garantizando su seguridad, acceso y uso apropiado. Planificación de inscripciones de alumnos nuevos y antiguos, atender a los estudiantes en la recepción de consultas académicas.

- **RESPONSABLE DE CENTRO INFORMATICO**

Objetivo General.- Encargado de que los sistemas de información y los medios electrónicos de procesamiento de datos funcionen adecuada, permanente y eficientemente, garantizando la seguridad y la actualización.

Funciones específicas.- Administración técnica del servidor del INSSB, disposición de copias de seguridad acerca de la información institucional, desarrollo de software según las necesidades del Instituto, mantenimiento del

hardware de cada unidad, modificación de la base de datos, actualización de la información académica y de la página Web.

3.2.1.2. PROCESOS ACADEMICOS GENERALES

A continuación se detallan los principales procesos detectados durante el Registro de Estudiantes:

1. REGISTRO DE LA INFORMACION DEL ESTUDIANTE (Solo estudiantes nuevos)

Una vez enviada las listas de los estudiantes aprobados en los exámenes de admisión a los Institutos Normales Superiores por el Ministerio, personal de Archivo Kardex procede a registrar toda la información del estudiante, previa presentación de documentos establecidos como prerrequisitos. Este proceso es uno de los mas morosos ya que se registran todos los datos necesarios, según el formato de Registro Personal (Ver Anexo F), lo cual debería ser de forma rápida para la atención mejorada para los propios estudiantes y encargados del registro personal.

Luego el siguiente proceso es el de Matriculación e Inscripción, para el cual sale un cronograma según carreras y semestres.

2. MATRICULACIÓN

La matriculación de estudiantes regulares esta a cargo de la Unidad de Contabilidad, la cual llena las matriculas (valoradas) manualmente y cobra un monto de 50 Bs por matricula, en esta Unidad existe poco personal (2 personas) para la atención del estudiantado haciendo morosa la atención.

3. INSCRIPCION DE ESTUDIANTES

Las inscripciones se realizan después de la matriculación, personal de Archivo y Kardex es el que esta encargado de este proceso, primeramente los estudiantes llenan la boleta de inscripción (manualmente) con las materias a cursar, la cual es verificada mediante listas de materias aprobadas, si las materias a cursar son las correctas se sella la boleta y se registran sus datos, si no se corrige la boleta y se actualiza las materias a cursar, esta información es almacenada en hojas de texto de Microsoft Excel y no se cuenta con una base de datos, con lo cual se almacena y procesa información redundante y ambigua. Esta verificación manual es la que retrasa este proceso.

4. EMISION DE BOLETINES (Listas Oficiales por paralelo)

Una vez concluida el periodo de inscripciones se procede a elaborar los Boletines o Listas oficiales de cada paralelo, el cual se realiza en la Unidad de Archivo Kardex, estas listas son extractadas y seleccionadas de los archivos de Excel en los que se encuentran los estudiantes inscritos por gestión.

5. REGISTRO Y CONTROL DE NOTAS

El registro y control de notas es realizado a la culminación de cada gestión, los docentes transcriben sus notas manualmente a los boletines de notas, los cuales son sellados y archivados en las actas de cada gestión. Estas notas son copiadas en hojas de texto de Microsoft Excel, llamadas Centralizadores, las cuales posteriormente son impresas y sirven para el control de prerrequisitos en la siguiente gestión.

3.3. FASE DE PLANEACIÓN Y ELABORACIÓN.

En la fase de Planeación y Elaboración del desarrollo del proyecto se centra en el informe preliminar de los requerimientos del sistema además de la descripción de las áreas involucradas, se definen los casos de uso, se presenta el modelo conceptual inicial, se registra términos mas significativos en el glosario, se define los diagramas de secuencia del sistema, se define contratos para las operaciones del sistema y se define los diagramas de estado.

3.3.1. INFORME PRELIMINAR

El informe preliminar es el resultado de la investigación acerca de los requerimientos que se necesitan, además de entender el funcionamiento de los procesos realizados en el Área Académica del INSSB, se necesita conocer las tareas que realizan, los departamentos o unidades funcionales involucradas además de los procedimientos administrativos, esta tarea se lograra a través de observación, entrevistas, manuales (procedimientos y funciones), cuestionarios y documentos que nos facilitan la recopilación de dicha información.

3.3.2. ESPECIFICACIÓN DE REQUERIMIENTOS.

Los requerimientos son una descripción de las necesidades del usuario su, meta es identificar y documentar lo que se necesita de una forma clara. Los puntos que tocaremos serán: panorama general, clientes, metas, funciones del sistema.

3.3.2.1. PANORAMA GENERAL

El Objetivo de este proyecto es, desarrollar e implementar el Sistema de Gestión Académica para el Instituto Normal Superior Simón Bolívar, que coadyuvará en los procesos de Registro de estudiantes, Inscripción, Matriculación de Estudiantes, Registro y control de Notas, para un mejor manejo de la información utilizando nuevas tecnologías, que permita involucrar a administrativos, docentes y estudiantes en el proceso educativo, ofreciendo información oportuna y coherente a cada uno de ellos.

3.3.2.2. CLIENTES

Las Unidades académicas del Instituto Normal Superior Simón Bolívar, manejan mucha información y necesitan conocer información oportuna y precisa según sus especificaciones, además de otros usuarios indirectos como ser estudiantes, docentes y personal administrativo.

- **ESTUDIANTES**

Usuarios Indirectos quienes necesitan:

- Conocer sus datos de Registro Personal
- Deudas de documentos.
- Obtener la Hoja de Ingreso Personal.
- Obtener la Matricula.
- Obtener la Boleta de inscripción.
- Conocer Materias a cursar
- Conocer Materias Inscritas y paralelos
- Obtener Historial de Notas.
- Obtener su Record Académico.
- Obtener sus Globales.

- **ENCARGADO DE ARCHIVO KARDEX**

Unidad encargada del manejo de la información académica, necesita:

- Información detallada de los estudiantes.
- Emitir la Hoja de Registro Personal después del proceso de Registro de estudiantes nuevos.
- Verificación automática de materias a cursar y materias de arrastre de un estudiante.
- Emitir boletas de inscripción.
- Lista de estudiantes inscritos por gestión, carrera y especialidad.
- Emitir las listas oficiales de estudiantes inscritos por paralelo.
- Reportes del registro de estudiantes nuevos.
- Reportes de estudiantes inscritos por gestión.

- Actualización de documentos.
- Emitir Historial de Notas por semestre.
- Emitir Record Académico.
- Emitir Globales a la conclusión de los estudios del estudiante.

- **ENCARGADO DE LA UNIDAD DE CONTABILIDAD**

Unidad encargada de la venta de valores y realizar el proceso de matriculación, necesita:

- Registrar la matriculación de los estudiantes por gestión.
- Emitir la matricula del estudiante.
- Lista de estudiantes matriculados por gestión, carrera y especialidad.
- Reportes de estudiantes matriculados.

- **ADMINISTRADOR DEL SISTEMA**

Persona encargada de manejar el sistema, requiere:

- Administrar el sistema, para que los usuarios habilitados realicen sus funciones y consultas eficientemente.
- Verificar la comunicación con las terminales.
- Autenticar a los usuarios del sistema (personal administrativo, docentes y personal encargado,) y asignar los niveles de usuario.
- Administrar el diseño académico
 - Asignación de materias.
 - Elaboración del pensum.
 - Habilitación de paralelos por especialidad y semestre.
- Administrar los proceso académicos (registro de estudiantes nuevos, inscripción y matriculación).
- Realizar Modificaciones a registros de la base de datos, previa solicitud de instancias superiores.

- **DOCENTE.**

Profesional que necesita:

- Realizar el registro de notas de estudiantes inscritos en su materia.
- Imprimir boletines de notas.
- Listado de materias a dictar con sus respectivos horarios.
- Listado de los estudiantes inscritos en su materia.

- **DIRECCION ACADEMICA Y DIRECCION GENERAL.**

Nivel ejecutivo de la Institución que necesita conocer reportes y cuadros comparativos de estudiantes registrados e inscritos, como ser:

- Emitir reporte general y específico de estudiantes nuevos admitidos por gestión, carrera y especialidad.
- Emitir reporte general y específico de estudiantes matriculados por gestión, carrera y especialidad.
- Emitir reporte general y específico de estudiantes inscritos por gestión, carrera, especialidad y paralelos.
- Emitir reportes de datos personales de los estudiantes (edad (promedio, mínima y máxima), sexo, estado civil).
- Emitir reportes de datos de procedencia del colegio de egreso (tipo de colegio (fiscal, particular o convenio), ciudad de egreso).
- Emitir reportes de datos socioculturales (idioma nativo que habla, lugar de nacimiento).

3.3.2.3. METAS

1. Registro de Estudiantes nuevos a la Base de Datos
2. Emisión de Hoja de Ingreso Personal.
3. Inscripción controlada.
4. Emisión de boletas de inscripción.
5. Matriculación de estudiantes.
6. Emisión de Matriculas de estudiantes
7. Despliegue de listas ordenadas de estudiantes inscritos por paralelo.
8. Registro de notas de los estudiantes.
9. Emisión de boletín de notas.
10. Emisión de historial de notas.
11. Emisión de Record Académico.
12. Emisión de globales.
13. Administración del diseño académico (pensum, paralelos materias).
14. Emisión automática de reportes.
15. Administración eficiente del sistema.
16. Administración de usuarios y niveles de usuario.
17. Control y seguridad del acceso de usuarios.

3.3.2.4. FUNCIONES DEL SISTEMA

Las funciones o requerimientos del sistema son lo que este deberá realizar, dicho sistema se detallará a continuación.

3.3.2.4.1. FUNCIONES BÁSICAS.

En la *tabla 3.1* observamos las funciones mínimas que debe realizar nuestro proyecto.

Ref. N°	Función	Categoría
R1.	Ofrece un registro persistente de datos académicos.	Evidente
R2.	Emitir la hoja de Registro Personal.	
R3.	Matricular estudiantes.	Evidente
R4.	Emitir Matricula del estudiante.	Evidente
R5.	Inscripción de estudiantes.	Evidente
R6.	Emisión de Boletas de Inscripción.	Evidente
R7.	Verificación interna de notas para controlar materias a cursar y materias de arrastre.	Oculto
R8.	Verificación interna de Prerrequisitos.	Oculto
R9.	Calculo del número de materias a cursar.	Evidente
R10.	Control de cupos por paralelo.	Oculto
R11.	Ofrece mecanismos de comunicación entre los procesos.	Oculto
R12.	Emisión de lista oficial de estudiantes inscritos por paralelo.	Evidente
R13.	Administración de usuarios y niveles de usuarios.	Evidente
R14.	Administración del Diseño Académico (Pensum, Materias y Paralelos).	Evidente
R15.	Administración de Procesos Académicos (Registro, Matriculación e Inscripción).	Evidente
R16.	Emisión de reportes sobre estudiantes registrados por carrera y especialidad por gestión.	Evidente
R17.	Registro de Notas de los estudiantes.	Evidente
R18.	Emision del boletín de Notas.	Evidente
R19.	Emisión de historial de notas, record académico, globales.	Evidente

Tabla 3.1 Funciones Básicas

3.3.2.5. ATRIBUTOS DEL SISTEMA

Atributos	Detalles y restricciones de frontera
Facilidad de uso	Cuando se ingrese al Sistema cada usuario tendrá un menú distinto, de acuerdo al nivel de usuario que tenga, cada uno con diferentes funciones de fácil uso.
Tiempo de respuesta	Cuando se actualiza los datos de la información requerida: inscripción, matriculación, datos personales de estudiantes, listas por paralelo, consultas y reportes, las opciones aparecen a lo mucho dentro 2 segundos.
Metáfora de interfaz	La interfaz será orientada de una forma accesible y comprensible al usuario con mensajes de acciones a cada evento del sistema según el modelo de navegación de los casos de uso extendido. Maximiza una navegación fácil mediante páginas Dinámicas PHP.
Tolerancia a fallas	Muestra un mensaje de error cuando no se encuentra algún registro, cuando encuentra duplicación de datos, o cuando no tiene acceso a ciertas áreas.
Plataforma	Microsoft Windows con navegador Internet Explorer o Mozilla.

Tabla 3.2 Tabla de Atributos del Sistema

3.4. FASE DE CONSTRUCCION

3.4.1. ETAPA DE ANÁLISIS DEL SISTEMA

El segundo paso de macro nivel es el de Construcción donde se cumplen los ciclos del desarrollo iterativo e incremental, el objetivo es la creación del sistema empezando en el primer ciclo. Dentro de la fase de Construcción se incluye el análisis que compone las actividades desarrolladas en la *Figura 2.12* del capítulo anterior, se muestra la dependencia de los artefactos que serán creados en esta fase que son: los Casos de Uso esenciales, los Diagramas de Casos de Uso, Modelo Conceptual, Glosario, Diagrama de Secuencia del Sistema y Contratos de Operaciones.

3.4.1.1. CASOS DE USO.

El caso de uso es un documento narrativo que describe la secuencia de eventos de un actor (agente externo) que utiliza un sistema para completar un proceso. La identificación de casos de uso requiere un conocimiento medio acerca de los requisitos, y se basa en la revisión de los documentos de requisitos existentes, y en el uso de la técnica de *brainstorming* entre los miembros del equipo de desarrollo.

A continuación se describen las características de los actores identificados:

ACTOR	DESCRIPCION
☺	Administrador del Sistema.
☺	Estudiantes
☺	Encargado y personal de archivo kardex
☺	Encargado de la unidad de contabilidad
☺	Docentes
☺	Director Académico

Tabla 3.3. Actores Identificados

3.4.1.2. DIAGRAMA DE CASO DE USO

Un diagrama de caso de uso explica gráficamente un conjunto de casos de uso de un sistema, en la figura 3.3 observamos el diagrama de caso de uso del Sistema planteado.

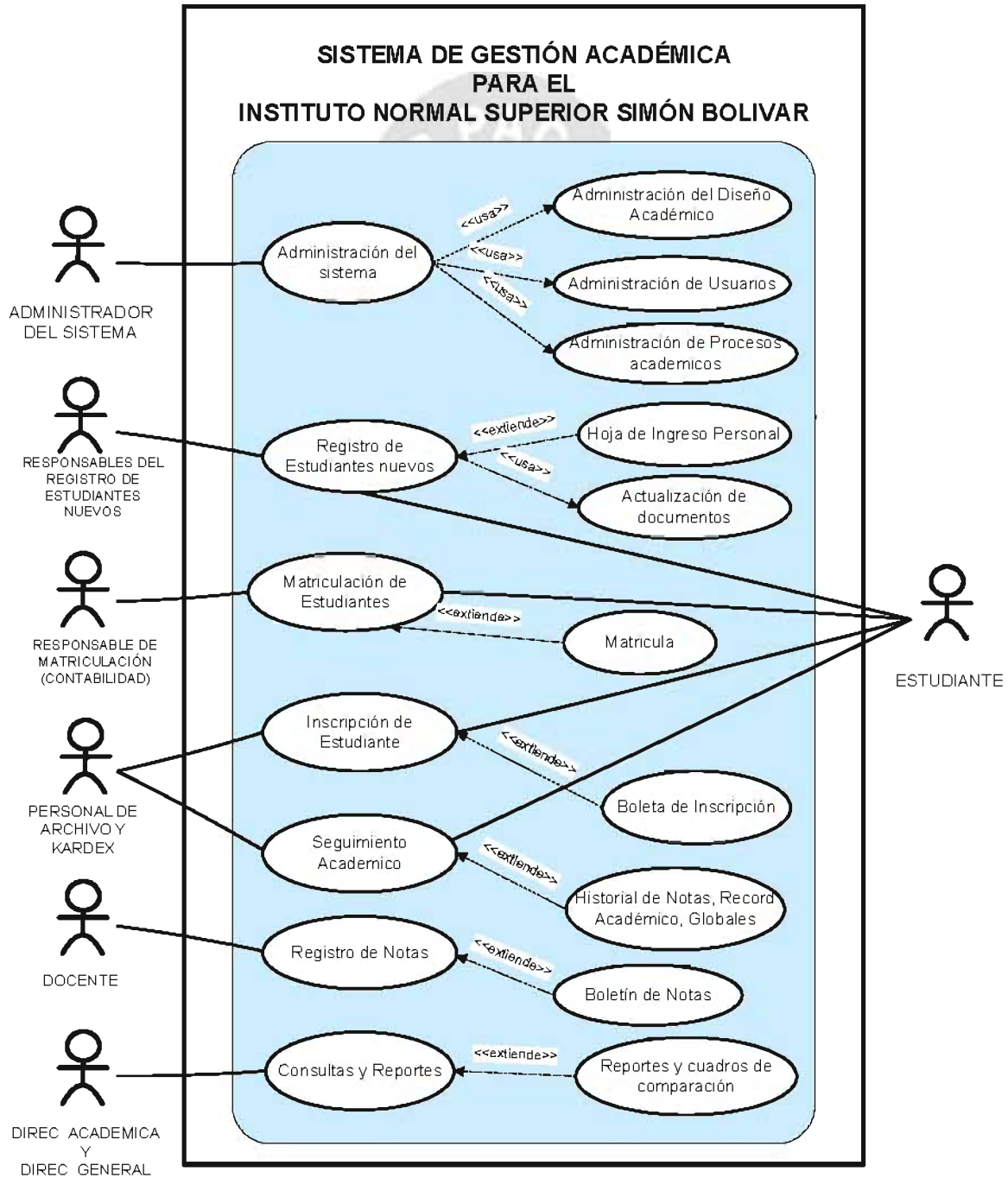
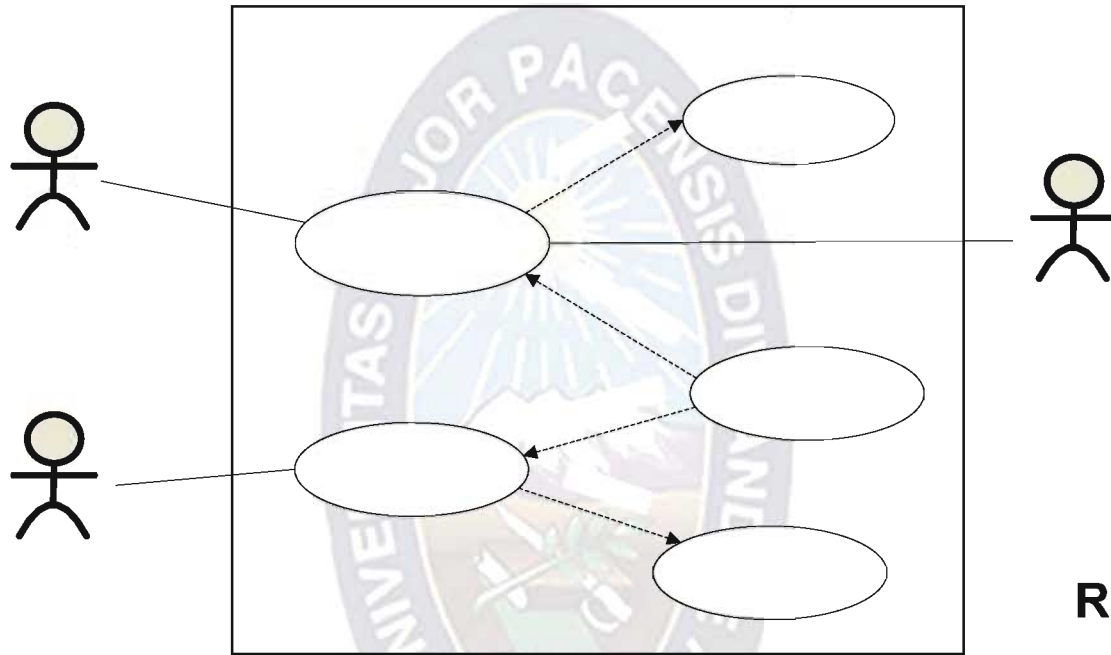


Figura 3.3. Diagrama de Casos de Uso del Sistema
Fuente: Elaboración Propia

Ahora veremos los casos de uso principales basado en Actores que se relacionan con la institución luego identificaremos los procesos que inicia o en los que participa. Como muestra de una explicación completa tomaremos el Proceso de Registro de estudiantes nuevos, Inscripción, Matriculación y registro de Notas.

DIAGRAMA DE CASO DE USO: REGISTRO DE ESTUDIANTES NUEVOS



REGISTRO D

Figura 3.4. Diagrama de Caso de uso: Registro de estudiantes nuevos

Caso de uso: Registro de Estudiantes nuevos	
Actores:	Personal de Archivo Kardex, estudiante, Administrador del sistema
Tipo:	Primario esencial Registra estudiante nuevo en el Sistema
Descripción:	El estudiante llega a una maquina para registrar sus datos. Si la base de datos del Sistema, el personal de archivo kardex llenara el formulario con los datos del estudiante, e imprimirá la hoja de Registro Personal, si existe algún error pasa donde el Administrador del Sistema quien realiza las correcciones correspondientes, posteriormente se procede a la impresión de la hoja de Registro Personal corregida.

Tabla 3.4. Caso de uso: Registro de estudiantes nuevos

DIAGRAMA DE CASO DE USO: MATRICULACION

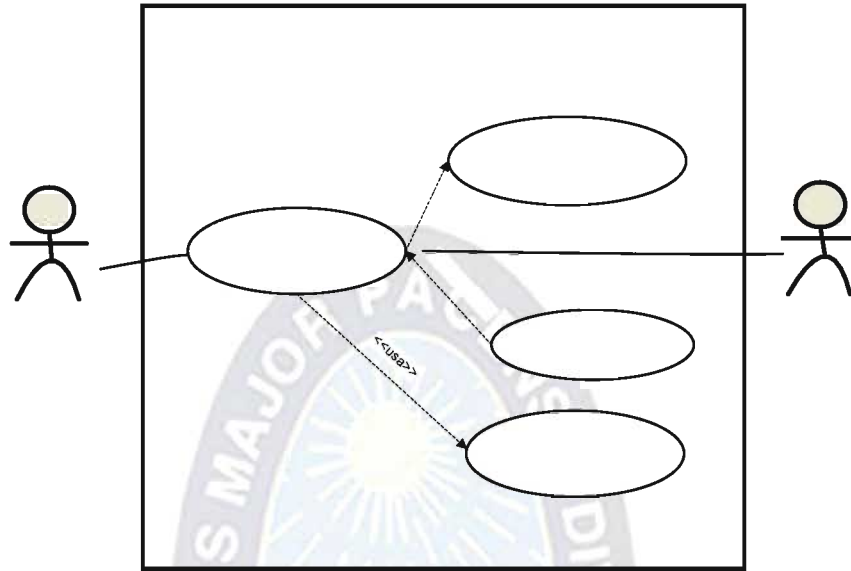


Figura 3.5. Diagrama de Caso de uso: Matriculación

Caso de uso: Matriculación del Estudiante.	
Actores:	Personal de la unidad de Contabilidad, estudiante. Matricular a Estudiante
Tipo:	Primario esencial
Descripción:	<p style="text-align: center;">RESPONSABLE DE MATRICULACIÓN (CONTABILIDAD)</p> <p>El estudiante llega a la contabilidad con la matricula vacía valorada previamente, el personal de contabilidad introducirá su C.I. del estudiante, el sistema le mostrara los datos del estudiante en un formato de hoja blanca que coincide con el formato de la matricula valorada, una vez verificado los datos se procede a la impresión de la matricula.</p>

Tabla 3.5. Caso de uso: Matriculación

DIAGRAMA DE CASO DE USO: INSCRIPCION

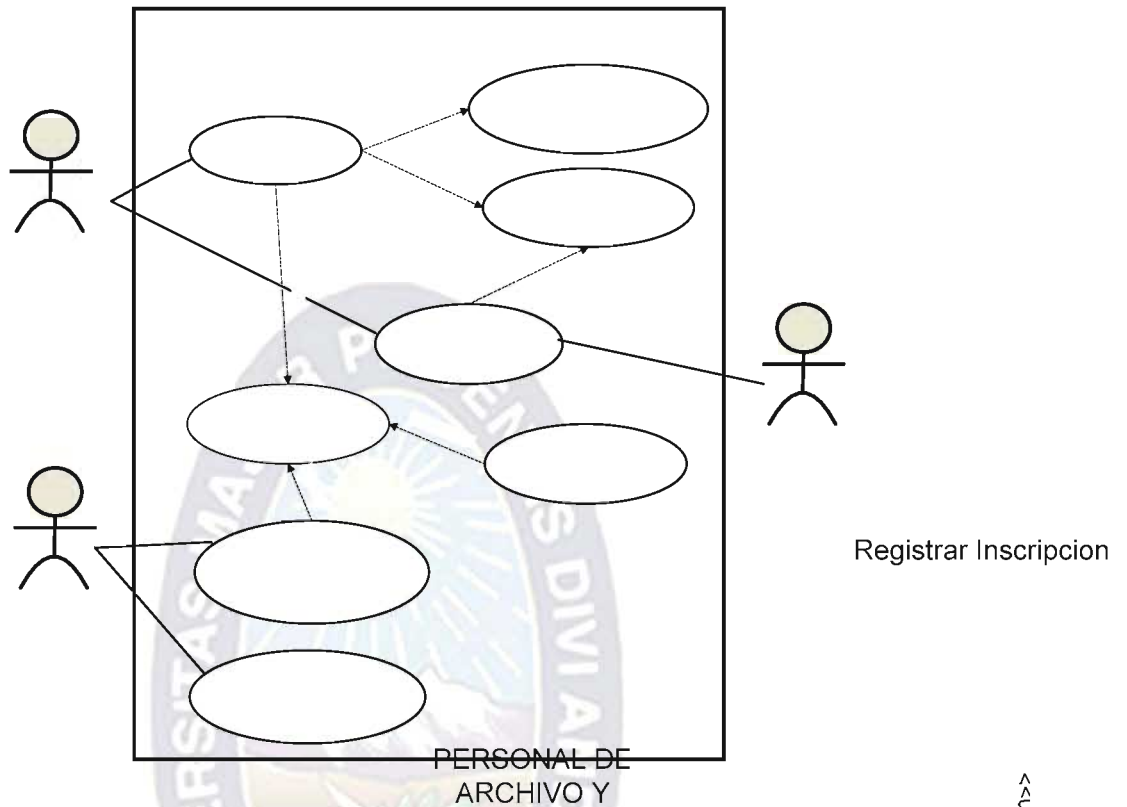


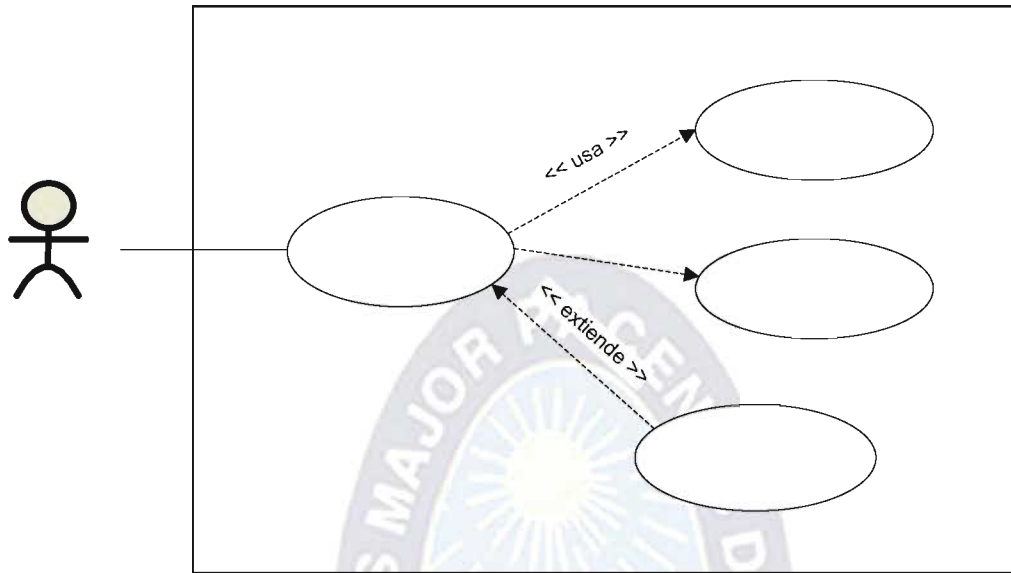
Figura 3.6. Diagrama de Caso de uso: Inscripción

Caso de uso: Inscripción de estudiante	
Actores:	Personal de Archivo Kardex, estudiante, Administrador del sistema
Tipo:	Primario esencial
Descripción:	El estudiante llega a una maquina para inscribirse en las materias correspondientes, el personal de archivo kardex introducirá su C.I. del estudiante donde el sistema le mostrara las materias a cursar que debe seleccionar el estudiante, procediendo luego a la grabación de la inscripción, posteriormente se procede a la impresión de boleta de inscripción con las materias a inscritas en la presente gestión.

Tabla 3.6. Caso de uso: Inscripción

Habilitar materias y paralelos

DIAGRAMA DE CASO DE USO: REGISTRO DE NOTAS



REGIST

Figura 3.7. Diagrama de Caso de uso: Registro de Notas

Caso de uso: Registro de Notas		Registrar Notas
Actores:	Docente	
Tipo:	Primario esencial	
Descripción:	<p>Los docentes ingresan al sistema, se autentifican y proceden a registrar sus notas, una vez registradas se procede a la impresión de los boletines de calificación, para tener constancia física de las notas, estos son archivados en las actas de cada gestión.</p>	

Tabla 3.7. Caso de uso: Registro de Notas

DIAGRAMA DE CASO DE USO: ADMINISTRACION DEL SISTEMA



Figura 3.8. Diagrama de Caso de uso: Administración del Sistema

Caso de uso: Administración el Sistema	
Actores:	Usuarios, Administrador del sistema
Tipo:	Primario esencial
Descripción:	Las personas que tendrán acceso al sistema se registran como usuarios, el administrador les asigna un nivel de usuario correspondiente a la función que desempeñará. Administra y configura el diseño académico, habilitando y asignando paralelos, docentes y horarios. También modifica algunos registros previa solicitud de instancias superiores.

Tabla 3.8. Caso de uso: Administración del Sistema

3.4.1.3. MODELO CONCEPTUAL.

Un modelo conceptual explica los conceptos significativos en un dominio del problema es el artefacto mas importante a crear durante el análisis orientado a objetos.

En este flujo de trabajo del proceso se centra en la creación del Modelo Conceptual y este se llevara a cabo partiendo de los Casos de Uso.

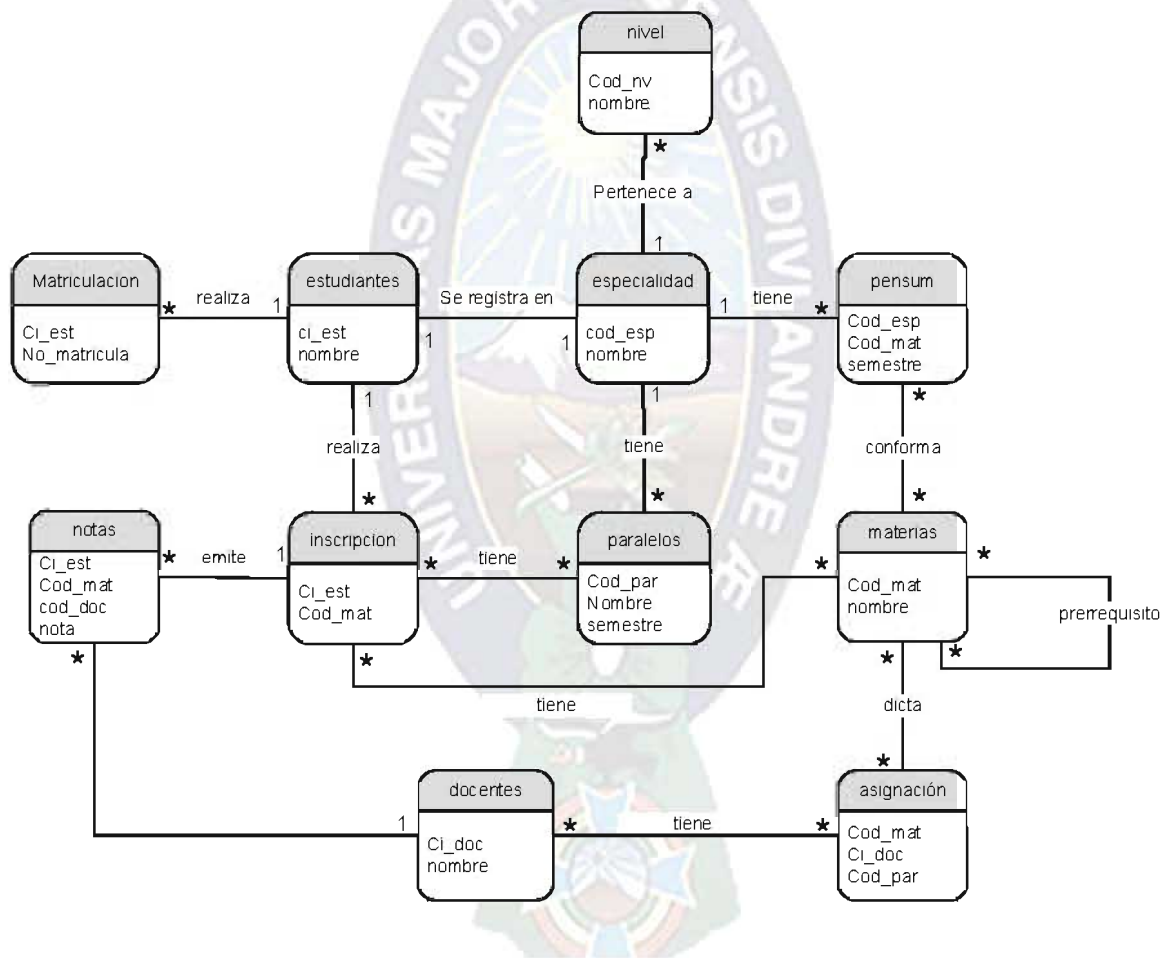


Figura 3.9. Diagrama Modelo Conceptual Preliminar

3.4.1.4. TERMINOS EN EL GLOSARIO.

El Glosario o diccionario modelo (semejante a un diccionario de datos) es un documento simple en el cual definen términos que requieren explicación para mejorar la comunicación y aminorar el riesgo de malos entendidos. Para un mejor estudio de nuestro Glosario indicaremos los datos importantes mediante una tabla que estará dividida en tres secciones: Término, categoría y comentario.

Donde Término nos indicará el caso de uso requerido, Categoría el tipo de uso que se le da al requerimiento anunciado, Comentario es la breve explicación para un mejor entendimiento en la tabla 3.9 Glosario.

TERMINO	CATEGORÍA	COMENTARIOS
Registrar Datos en el sistema	CASO DE USO	Su labor se halla en tener registrados todos los datos tanto del Estudiantes, Usuarios, Especialidades, pensum, materia, horarios y paralelos que necesita manejar la unidad de Archivos.
Nombre	TIPO	Un nombre que tiene cada dato ya sea de Estudiante, usuario, especialidad, materia, horario u otro.
Conexión al sistema	CASO DE USO	Los usuario habilitados para ingresar al sistema, hacen una conexión al sistema, introduciendo código y contraseña, el sistema devuelve la conexión admitiéndole el ingreso al mismo siempre y cuando estos datos sean validos.
Matriculación	CASO DE USO	Proceso que se realiza semestralmente, donde los estudiantes adquieren su matricula.
Inscripción	CASO DE USO	Esta función es la mas importante, deben estar bien registrados los datos, para que el sistema pueda realizar dicho proceso
Emitir Boleta de Inscripción	ATRIBUTO	Personal de Archivo y Kardex emite la boleta de inscripción con las materias a cursar del estudiante

Emitir Lista de estudiantes inscritos por paralelo	ATRIBUTO	Posteriormente a la inscripción, personal de archivo y kardex procede a emitir las listas oficiales de estudiantes inscritos por paralelo.
Cambio Contraseña/Clave.	CASO DE USO	El estudiante, docente, padre de familia o autoridades introduce su clave actual , el sistema valida la clave, el usuario lee las recomendaciones para introducir una nueva clave y luego el usuario introduce su nueva clave para que luego el sistema actualiza la base de datos con la nueva clave introducida
Visualización de Consultas y Reportes	CASO DE USO	Las Direcciones General y Académica pueden acceder a la información Académica que desee acerca de los reportes y estadísticas de los estudiantes después de cada proceso de inicio y cierre de gestión
Registrar Notas	CASO DE USO	De uso exclusivo del docente, quien ingresa a este modulo y registra las notas de los estudiantes, para luego imprimir los boletines de notas.
Administración del Sistema	CASO DE USO	Específicamente para el ingreso del administrador quien tiene acceso a todas las opciones del sistema, para brindar un adecuado control del sistema

Tabla 3.9. Glosario

3.4.1.5. DIAGRAMAS DE SECUENCIA

Los diagramas de secuencia muestran de manera gráfica los eventos u operaciones que fluyen de los actores al sistema, como es que este responde a determinadas operaciones, definiendo el comportamiento del sistema como una "caja negra", es decir describir lo que hace, sin explicar la manera de cómo lo hace.

Una de las primeras interacciones del sistema con los usuarios es el registro y autenticación de los usuarios autorizados; el sistema verifica si están registrado en la base de datos validando el C.I. y contraseña si están registrados pueden acceder al sistema, caso contrario son rechazados.

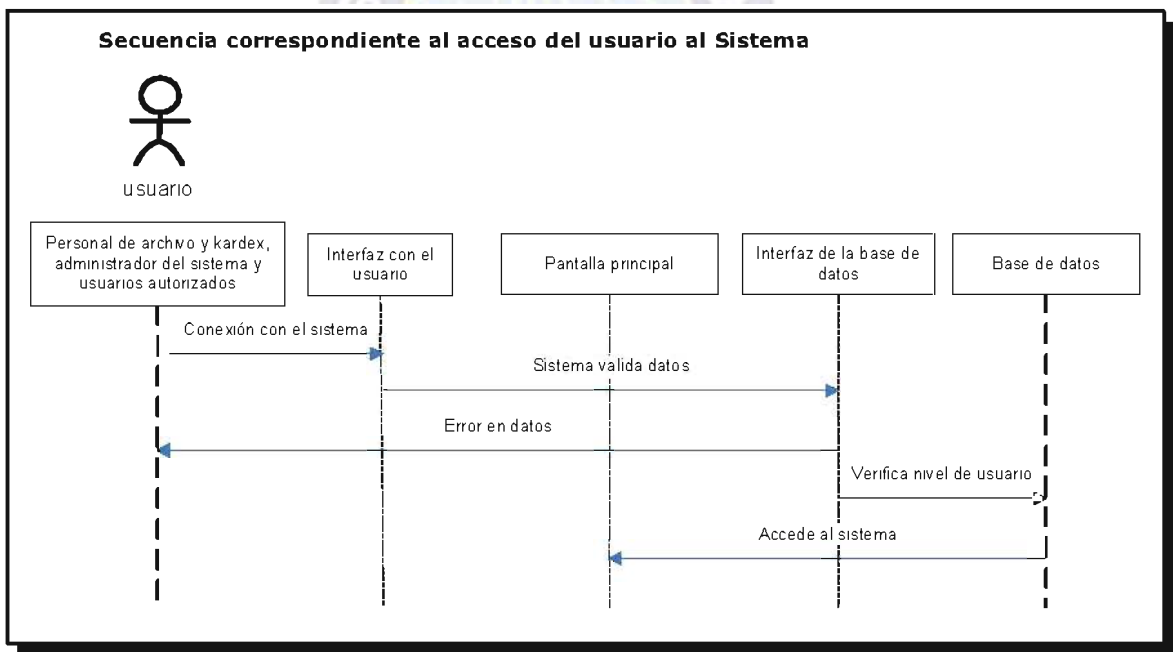


Figura 3.10. Secuencia correspondiente al control de acceso del usuario

Al comienzo de cada gestión el sistema inicia con el registro de estudiantes nuevos, proceso del cual están encargados el personal de archivo y kardex, estos se autentifican y proceden a registrar la información del estudiante, si los datos son validos se graban en la base de datos, luego se emite el Registro Personal del Estudiante, el cual se archiva en su File Personal.

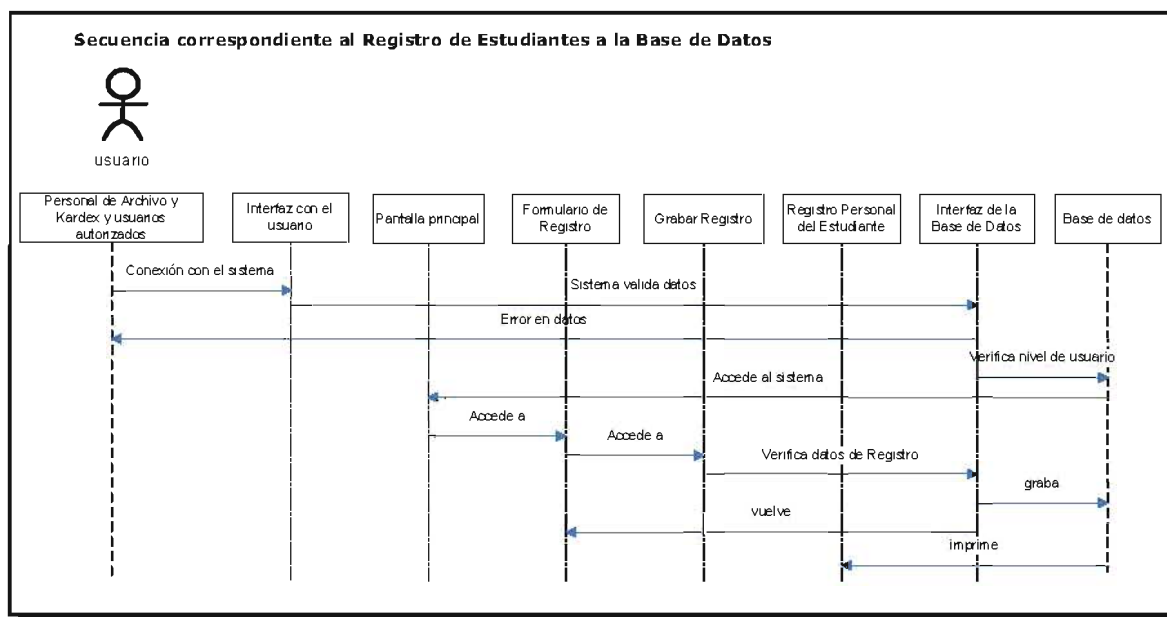


Figura 3.11. Secuencia correspondiente al registro de estudiantes

Cuando ya se tiene la información del estudiante registrados en la base de datos, se procede a matricular al estudiante y emitir su matricula correspondiente, este trabajo esta encargado a personal de la Unidad de Contabilidad.

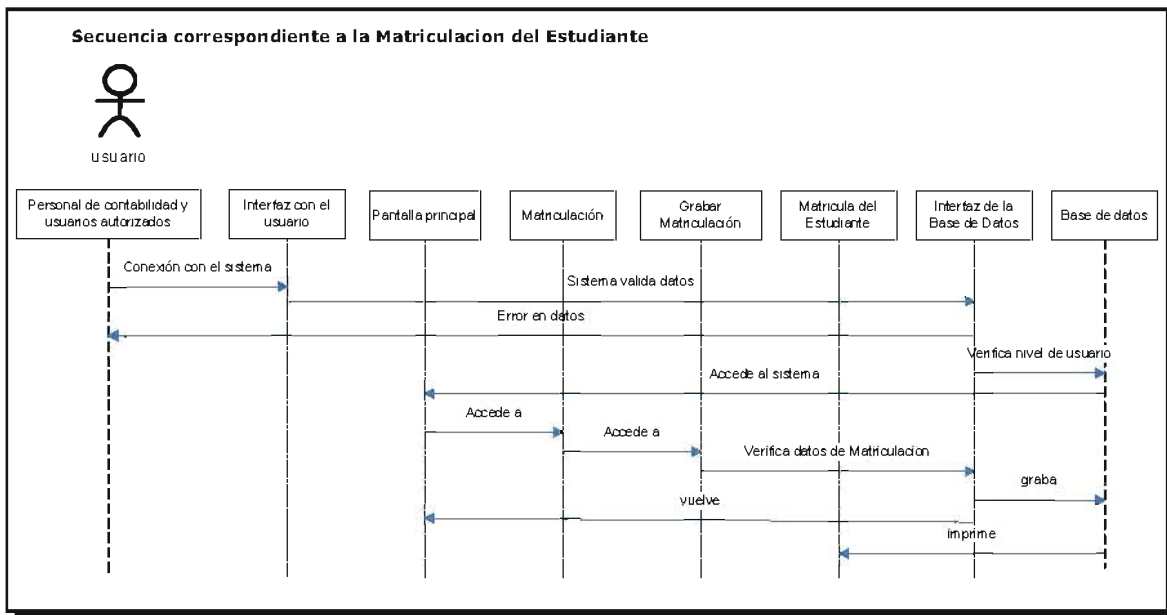


Figura 3.12. Secuencia correspondiente a la Matriculación

La secuencia correspondiente a la inscripción inicia, cuando ya están matriculados el sistema emite las materias a cursar del cual se elige las materias a inscribirse, después se procede a grabar la inscripción para luego emitir la boleta de inscripción como se puede observar en la *Figura. 3.13*.

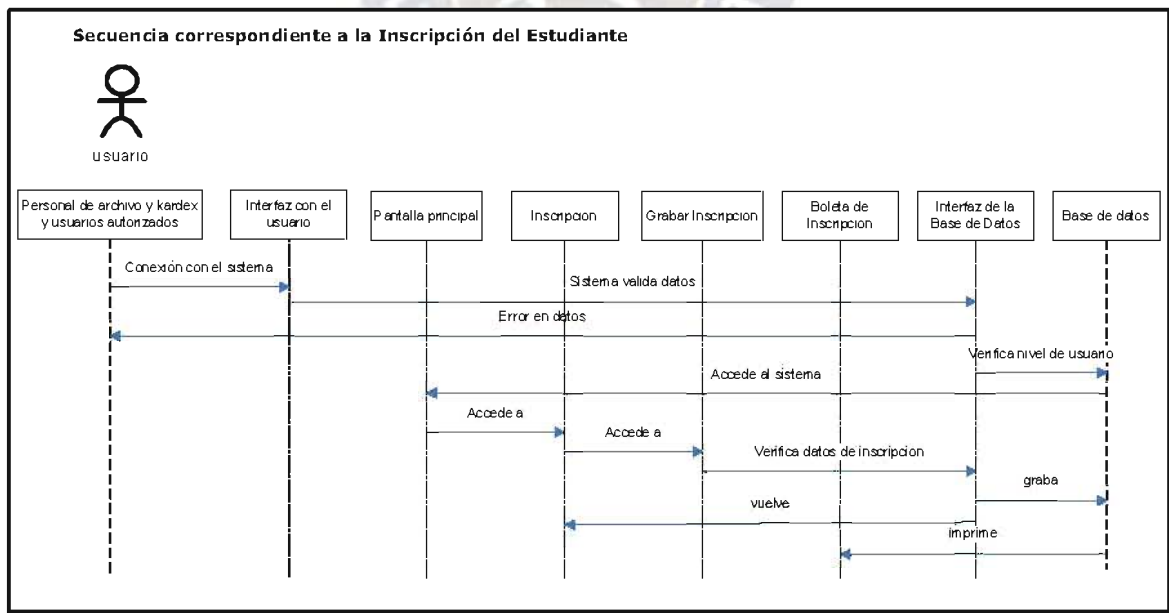


Figura 3.13. Secuencia correspondiente a la Inscripción

La secuencia que corresponde al proceso de registro de notas, comienza un vez que los estudiantes están inscritos en sus materias y paralelos correspondientes, los docentes se autentifican e ingresan al sistema, acceden y llenan al registro de notas por materia y finalmente proceden a emitir el boletín de notas como se puede observar en la *Figura. 3.14.*

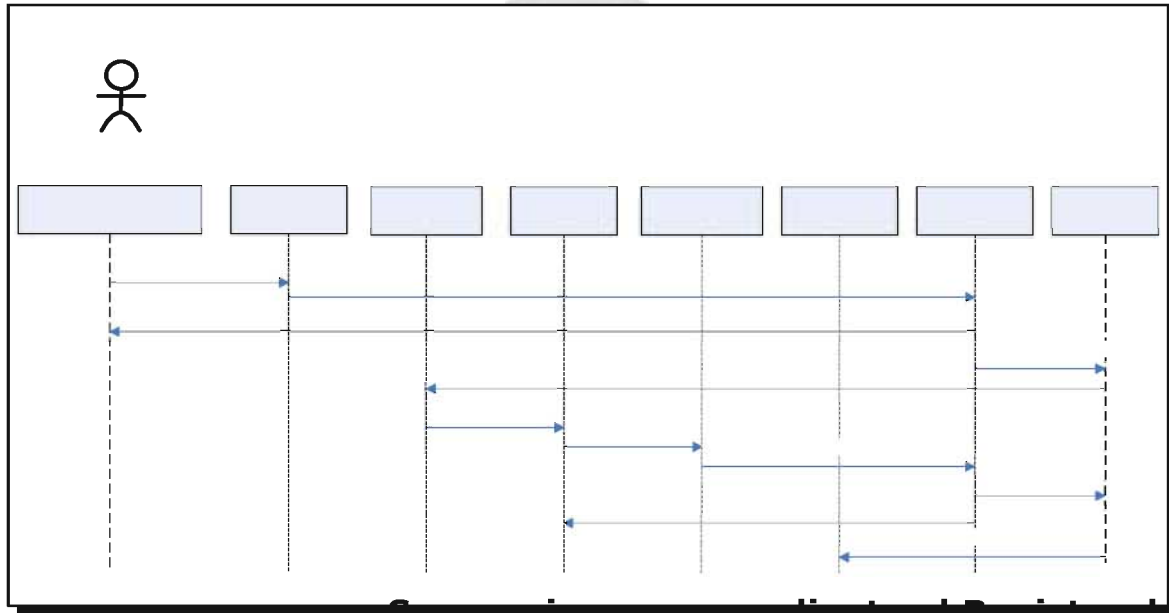
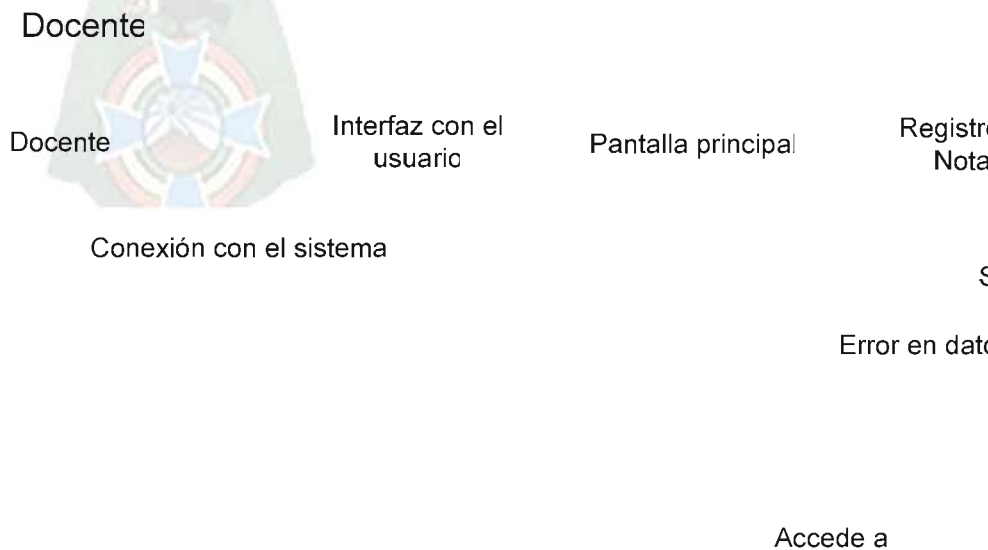


Figura 3.14. Secuencia correspondiente al Registro de Notas



3.4.1.6. CONTRATOS DE OPERACIONES.

Los contratos contribuyen a definir el comportamiento de un sistema; describen el efecto que sobre el tienen las operaciones.

En los contratos de operaciones se describe el comportamiento del sistema a partir de cómo cambia el estado cuando se llama a una operación.

1. Contrato para la verificación de usuario	
Nombre	autenticarUsuario(CI, contraseña)
Responsabilidades	Realizar una búsqueda del CI y contraseña del usuario en la base de datos, luego verificar que nivel de usuario tiene asignado, para desplegar un menú de opciones adecuado a sus necesidades
Tipo	Sistema
Referencias cruzadas	Funciones del sistema: R11, R13.
Notas	Realizar una búsqueda y acceso rápido a la base de datos.
Excepciones	Si la información introducida [CI y contraseña] no es válida, se indicará con el mensaje "llene los campos vacíos". Si los datos no son validos se despliega el mensaje "Introduzca solo caracteres numéricos o alfanuméricos"
Salida	Despliegue del menú de usuario que le corresponde
Precondiciones	El sistema tendrá en la base de datos la información de cada usuario, referenciado por su [CI y contraseña].
Poscondiciones	o Habilitar a usuario para que tenga acceso a las opciones determinadas.

Tabla 3.10. Contrato correspondiente al caso de uso: Verificaron de usuario

A continuación se muestra en la Tabla 3.11 el contrato referente al Registro de estudiantes nuevos, donde se indica los pasos a seguir para realizar mencionado proceso.

1. Contrato para el Registro de Estudiantes Nuevos	
Nombre	registrarEstudiante (CI, datos del estudiante)
Responsabilidades	Grabar la información del estudiante a la base de datos. Asignarle un código de estudiante. Emitir el Registro Personal del Estudiante
Tipo	Sistema
Referencias cruzadas	Funciones del sistema: R1, R2.
Notas	Grabar de forma rápida y persistente a la base de datos.
Excepciones	Si no se llenan los campos obligatorios, se muestra un mensaje "llene el campo ...", si la información introducida [datos] no son válidas, se indicará con un mensaje de datos no validos
Salida	Grabar los datos del estudiante a la base de datos impreso el Registro Personal del Estudiante.
Precondiciones	Usuario autorizado para registrar al estudiante a la base de datos.
Poscondiciones	<ul style="list-style-type: none"> ○ Se inscribirá al alumno siempre y cuando tenga todos los prerequisites. ○ Se asociará con el registro de alumnos.

Tabla 3.11. Contrato correspondiente al caso de uso: Registro de estudiantes nuevos

A continuación se muestra en la Tabla 3.12 el contrato referente a la matriculación, donde se indica los pasos a seguir para realizar mencionado proceso.

1. Contrato para la Matriculación	
Nombre	matricularEstudiante(CI, tipoPago)
Responsabilidades	Buscar al estudiante hasta encontrarlo para realizar su respectiva matriculación, seleccionar la gestión y la forma de pago.
Tipo	Sistema
Referencias cruzadas	Funciones del sistema: R3, R4.
Notas	Realizar una búsqueda y acceso rápido a la base de datos.
Excepciones	Si la información introducida [CI] no son válidos, se indicará con un mensaje que "ya se matriculo en esta gestión".
Salida	Matricula impresa
Precondiciones	El estudiante deberá estar registrado en la base de datos de, este será referenciado por su [CI].
Poscondiciones	<ul style="list-style-type: none"> ○ Se matriculará al estudiante siempre y cuando este registrado en la base de datos, y sino ha tenido ningún tipo de observaciones. ○ Se asociará con el registro de alumnos.

Tabla 3.12. Contrato correspondiente al caso de uso: Matriculación

A continuación se muestra en la Tabla 3.13 el contrato referente a la inscripción, donde se indica los pasos a seguir para realizar mencionado proceso.

1. Contrato para la Inscripción	
Nombre	InscribirEstudiante(CI,seleccionar materias)
Responsabilidades	Realizar una búsqueda del estudiante hasta encontrar para realizar su respectiva inscripción, seleccionar las materias a inscribirse y el paralelo, luego le listará las materias a cursar según los prerrequisitos.
Tipo	Sistema
Referencias cruzadas	Funciones del sistema: R5, R6, R7, R8, R9 y R10.
Notas	Realizar una búsqueda y acceso rápido a la base de datos.
Excepciones	Si la información introducida [CI del estudiante] no es válido, se indicará con un mensaje que se cometió algún error
Salida	Reporte impreso llamado Boleta de Inscripción.
Precondiciones	El sistema tendrá en la base de datos la información de cada alumno referenciado por su [CI].
Poscondiciones	<ul style="list-style-type: none"> o Se inscribirá al alumno siempre y cuando tenga todos los requisitos, y sino ha tenido ningún tipo de observaciones. o Se asociará con el registro de estudiantes. o El proceso permitirá acceder a las notas de los alumnos el cual nos mostrara que materias aprobó.

Tabla 3.13. Contrato correspondiente al caso de uso: Inscripción

A continuación se muestra en la Tabla 3.14 el contrato referente al registro de notas, que realiza el docente una vez culminada la gestión, a continuación se indica los pasos a seguir para realizar mencionado proceso.

1. Contrato para el Registro de Notas	
Nombre	registrarNotas(materia, paralelo, especialidad)
Responsabilidades	<p>Buscar a los estudiantes inscritos en su materia y paralelo según especialidad para registrar las notas finales de cada estudiante.</p> <p>Emitir el boletín de notas, el cual se archivara en las actas correspondientes a la presente gestión.</p>
Tipo	Sistema
Referencias cruzadas	Funciones del sistema: R3, R4.
Notas	Realizar una búsqueda y acceso rápido a la base de datos.
Excepciones	Si los registros de la presente gestión (periodo y año) ya están llenados no se podrá ingresar nuevamente al registro, y se indicará un mensaje que "ya se registró las notas de esta materia en esta gestión".
Salida	Boletín de Notas impreso.
Precondiciones	El estudiante deberá estar inscrito en la materia y paralelo que dicta el docente, en la especialidad indicada.
Poscondiciones	<ul style="list-style-type: none"> ○ Se registrará la nota del estudiante siempre y cuando este inscrito en la materia y paralelo correspondiente. ○ Se emitirá el boletín de calificaciones una vez culminado el registro de notas de todos los estudiantes.

Tabla 3.14. Contrato correspondiente al caso de uso: Registro de Notas

3.4.2. ETAPA DE DISEÑO Y CONSTRUCCIÓN.

La etapa de diseño es la segunda parte de la fase de construcción, su principal objetivo es el diseño lógico del sistema.

3.4.2.1. CASOS REALES DE USO.

En la fase de planeación y elaboración se desarrollaron los casos de uso de alto nivel y los casos de uso expandidos, ahora mediante los casos de uso reales describiremos el diseño de forma concreta la manera en la que se realiza un caso de uso (proceso), donde se detalla la problemática que tienen los usuarios y de que pasos, seguirán para la solución de la misma.

Registro de Estudiantes

Caso de uso: Registro de Estudiantes	
Actores	Personal de Archivo y Kardex (iniciador), Estudiante, Administrador del Sistema.
Propósito:	Registrar los datos del estudiante, especificados en el formulario.
Resumen:	Un Estudiante llega a registrar sus datos en el sistema, El personal de archivo y kardex registra los datos especificados en el formulario. El administrador del sistema es el único que puede modificar los datos del estudiante.
Tipo:	Primario y real
Referencias cruzadas:	Funciones: R1, R2, R11, R15.

Tabla 3.15. Casos de Uso Reales: Registro de Estudiantes

A continuación se especifica el formulario del registro de estudiantes nuevos.

**REGISTRO PERSONAL DEL ESTUDIANTE
GESTION 2007**

DATOS PERSONALES

CAÑILLO NINA JULIAN CARLOS SEXO
(*) APELLIDO PATERNO (*) APELLIDO MATERNO (*) NOMBRES M F

Ci: 5099321 LP **FECHA DE NACIMIENTO** 12 ABRIL 1990
(*) DIA (*) MES (*) AÑO

LUGAR DE NACIMIENTO LA PAZ MURILLO
(*) DEPARTAMENTO (*) PROVINCIA

(*) ESTADO CIVIL: SOLTERO/A (*) IDIOMA NATIVO QUE HABLE: AIMARA

DATOS DE REFERENCIA

DIRECCION: MIRAFLORES BILIVAR 3689
(*) Zona o Barrio (*) Calle o Avenida Número

TELEFONO: 2 233564 EMAIL: jc_carrillo@hotmail.com

DATOS REFERIDOS AL COLEGIO DE PROCEDENCIA

Hugo Dávalos 2003 FISCAL La Paz
(*) Colegio de Egreso (*) Año de Egreso (*) Colegio de Procedencia (*) Ciudad de Egreso

DOCUMENTOS

SI NO

TÍTULO DE BACHILLER: 0035698 UMSA 2004
Número Universidad Año

LIBRETA DE SERVICIO MILITAR: 256987 LA PAZ 2003
Número Lugar Año

CERTIFICADO DE NACIMIENTO ORIGINAL

CEDULA DE IDENTIDAD LEGALIZADA

CERTIFICADO PRUEBA DE ADMISION

OBSERVACION

DATOS DE INSCRIPCION

GESTION DE INGRESO: 1 / 2007
(*) PERIODO (*) AÑO

CARRERA ESPECIALIDAD

NIVEL INICIAL

NIVEL PRIMARIO

NIVEL SECUNDARIO MATEMATICA

NIVEL PRIMARIO Y SECUNDARIO

OBSERVACIONES:

ACEPTAR LIMPIAR

(*) Datos Obligatorios

ATRAS INICIO SALIR

Figura 3.15. Ventana Formulario de registro de estudiantes nuevos.

Curso normal de los eventos

Acción de los actores

1. Un estudiante nuevo llega a registrar sus datos en el sistema
2. El personal encargado referencia al estudiante mediante su CI llenando además sus otros datos. Los campos etiquetados con (*) son de llenado obligatorio.
4. El personal encargado presiona el icono IMPRIMIR Hoja de Ingreso Personal.
5. Personal encargado cierra la vista de impresión y vuelve al formulario de Registro Personal.

Respuesta del sistema

3. Verifica si no existe error en el CI (duplicidad), además verifica la integridad y formato (numérico y alfanumérico) de los datos.
Si los datos son correctos aparece un mensaje de registro correcto con un icono de impresión de Hoja de Registro Personal, caso contrario aparece un mensaje de no cumplimiento de las verificaciones pidiendo llenar otra vez el formulario.
5. El sistema despliega una vista de impresión de la Hoja de Ingreso Personal con los datos del estudiante y se procede a la impresión.

Matriculación de Estudiantes

Caso de uso: Matriculación de Estudiantes	
Actores	Personal de Contabilidad (iniciador), Estudiante, Administrador del Sistema.
Propósito:	Registrar y emitir la matricula del estudiante del presente periodo y gestión.
Resumen:	Un Estudiante llega a matricularse. El personal del área de Contabilidad registra los datos especificados en el formulario. El administrador del sistema interviene en la modificación de algún dato erróneo introducido.
Tipo:	Primario y real
Referencias cruzadas:	Funciones: R3, R4, R11, R15.

Tabla 3.16. Casos de Uso Reales: Matriculación de Estudiantes

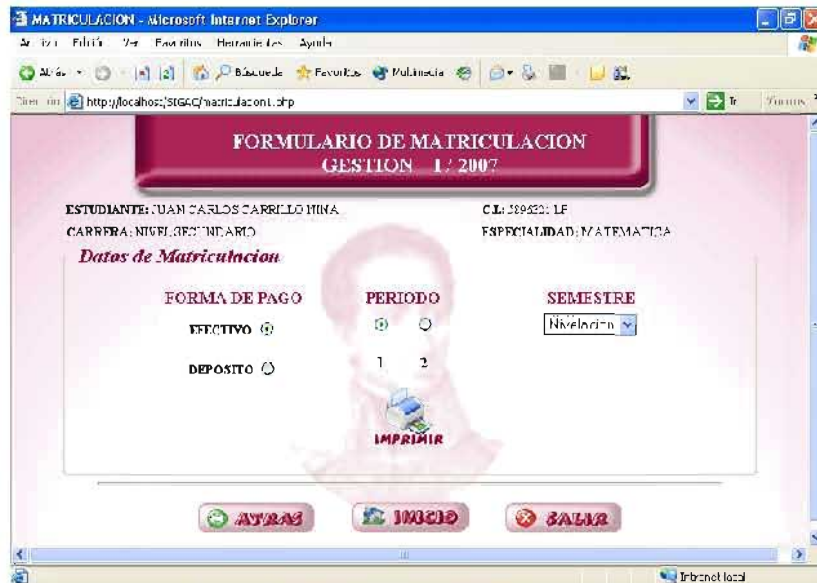


Figura 3.16. Ventana Formulario de Matriculación

Curso normal de los eventos	
Acción de los actores	Respuesta del sistema
1. Un estudiante llega a matricularse en la presente gestión	
2. El personal introduce el CI del estudiante.	3. Verifica si el CI existe, y despliega un pequeño formulario con los datos ya existente del estudiante y otros campos para ser llenados.
3. El encargado selecciona los campos requeridos en el formulario y presiona el icono IMPRIMIR matricula.	4. El sistema despliega una vista de impresión de la Matricula con los datos del estudiante y se procede a la impresión.
5. Personal encargado cierra la vista de impresión y vuelve al formulario de Matriculación.	

Inscripción de Estudiantes

Caso de uso: Inscripción de Estudiantes	
Actores	Personal de Archivo y Kardex (iniciador), Estudiante, Administrador del Sistema.
Propósito:	Inscribir en las materias correspondientes y emitir la boleta de inscripción del estudiante del presente periodo y gestión.
Resumen:	Estudiante llega a inscribirse en materias que le corresponden. El personal de archivo y kardex selecciona las materias a cursar en el presente periodo. El administrador del sistema es el único que puede modificar los datos de inscripción del estudiante.
Tipo:	Primario y real
Referencias cruzadas:	Funciones: R5, R6, R7, R8, R9, R10, R15.

Tabla 3.17. Casos de Uso Reales: Inscripción de Estudiantes

INSCRIPCIÓN
GESTIÓN I / 2007

ESTUDIANTE: JUAN CARLOS SANCHEZ MINA C.I.: 989636112
CARRERA: NIVEL SECUNDARIO ESPECIALIDAD: MATEMÁTICA

Elija 4 materias como máximo Paralelo: 1

MATERIAS A CURSAR	
CODIGO	MATERIA
<input checked="" type="checkbox"/> EU0402	TECNICAS DE ESTUDIO
<input checked="" type="checkbox"/> LFI1001	LENGUAJE
<input checked="" type="checkbox"/> MAT 003	INTRODUCCIÓN AL ALGEBRA
<input checked="" type="checkbox"/> MA1-004	FUNDAMENTOS BÁSICOS DE LA MATEMÁTICA

Aceptar

ATRAS INICIO SALIR

Figura 3.17. Ventana Formulario de Inscripción.

Curso normal de los eventos

Acción de los actores

1. Un estudiante llega a inscribirse en sus materias de la presente gestión

2. El personal introduce el CI del estudiante.

3. El personal encargado selecciona las materias a cursar y presiona el icono IMPRIMIR boleta de inscripción.

5. Personal encargado cierra la vista de impresión y vuelve al formulario de Inscripción.

Respuesta del sistema

3. Verifica si el CI existe o si ya se inscribió y despliega un pequeño formulario con los datos ya existentes del estudiante y las materias a cursar.

4. El sistema despliega una vista de impresión de las materias a cursar con los datos del estudiante y se procede a la impresión.

Registro de Notas de los Estudiantes

Caso de uso: Registro de Notas	
Actores	Docente.
Propósito:	Realizar el registro de notas del estudiante y la emisión del boletín de notas del presente periodo y gestión.
Resumen:	El docente accede Registrar Notas según la materia y paralelo asignado en su especialidad, procede al llenado de las notas de cada estudiante, una vez concluido se imprime el boletín de notas de la presente gestión.
Tipo:	Primario y real
Referencias cruzadas:	Funciones: R7, R11, R13, R17, R18.

Tabla 3.18. Casos de Uso Reales: Registro de notas



Figura 3.18. Ventana Materias a registrar notas.

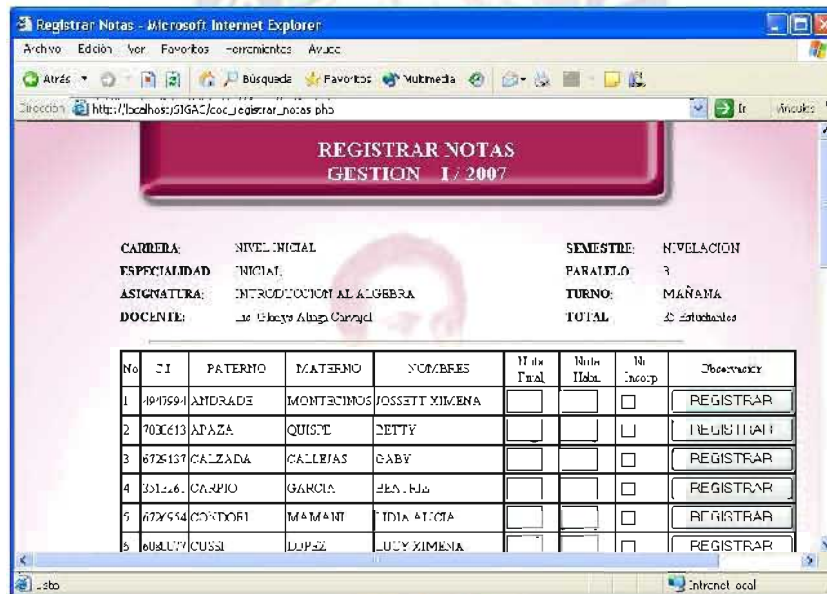


Figura 3.19. Ventana Registro de notas

Curso normal de los eventos

Acción de los actores	Respuesta del sistema
1. Docente ingresa al sistema autenticándose.	
2. Docente ingresa al modulo de registro de notas.	3. Despliega materias y paralelos asignados.
4. Docente elije materia y paralelo a registrar notas.	5. Muestra la lista de los estudiantes inscritos en su materia, junto con cajas de texto vacías donde se introducirán las notas finales.
6. Llena las notas de cada estudiante asignado en su materia.	7. Opción de imprimir boletines de notas
8. Emite el boletín de notas de esa materia.	

Administración de Usuarios

Caso de uso: Administración de Usuarios	
Actores	Administrador del Sistema(iniciador) y usuarios del sistema (personal encargado para trabajar en los procesos correspondientes).
Propósito:	Autenticar a los usuarios y asignar niveles de usuario para que puedan ingresar con su código y contraseña al sistema.
Resumen:	Un usuario llega a registrar sus datos en el sistema, El administrador del sistema registra los datos especificados en el formulario y asigna los niveles de usuario según las tareas encargadas a cumplir. También puede modificar sus datos, niveles de usuario y dar de baja a usuarios.
Tipo:	Primario y real
Referencias cruzadas:	Funciones: R11, R13.

Tabla 3.19. Casos de Uso Reales: Administración de Usuarios

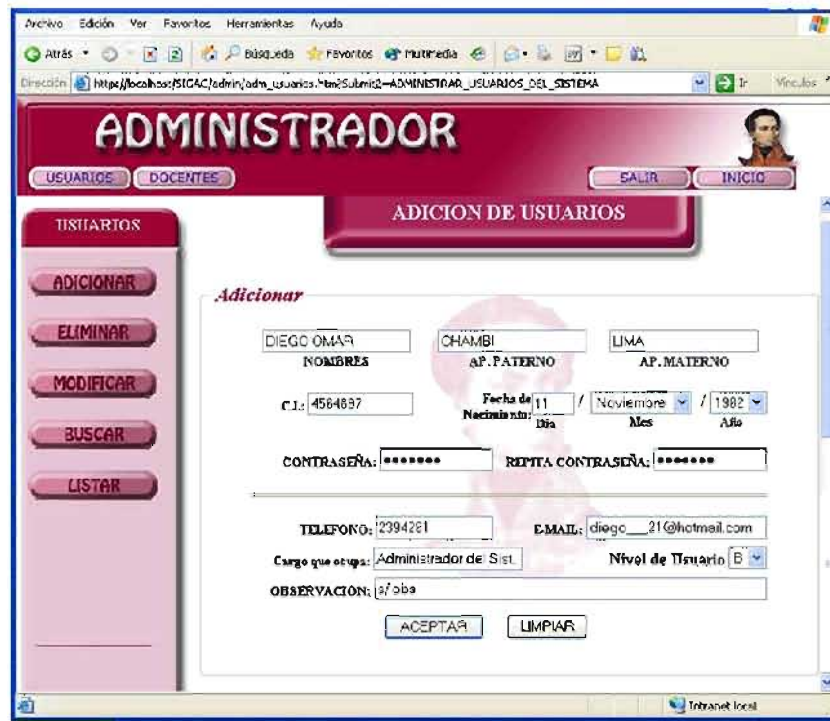


Figura 3.20. Ventana Formulario Administración de Usuarios

Curso normal de los eventos

Acción de los actores

1. Un usuario llega a registrar sus datos en el sistema.
2. El administrador del sistema referencia al usuario mediante su CI e introduce los datos especificados en el formulario, además le asigna un nivel de usuario según la tarea que le corresponda realizar.
4. El administrador del sistema cierra la ventana y vuelve al menú principal.

Respuesta del sistema

3. Verifica duplicidad del CI, integridad y formato de los datos, y despliega un mensaje de Usuario aceptado que puede ingresar al sistema.

Administración de Paralelos

Caso de uso: Administración de Paralelos	
Actores	Administrador del Sistema y personal encargado de asignar paralelos.
Propósito:	Habilitar paralelos en las diferentes especialidades que así lo necesiten.
Resumen:	El administrador junto con el personal encargado de la asignación de paralelos habilitan paralelos y cupos en el sistema para que los estudiantes puedan inscribirse.
Tipo:	Primario y real
Referencias cruzadas:	Funciones: R10, R11, R14.

Tabla 3.20. Casos de Uso Reales: Administración de Paralelos

Figura 3.21. Ventana administración de paralelos

Curso normal de los eventos

Acción de los actores

1. Personal encargado de la asignaron de paralelos llega con la lista de paralelos según aprobación de instancias superiores (Dirección Académica).

2. El administrador del sistema referencia los paralelos mediante su codificación e introduce su cupo, nombre y selecciona el turno y el semestre.

4. El administrador del sistema cierra la ventana y vuelve al menú principal.

Respuesta del sistema

3. Verifica duplicidad de la codificación, integridad y formato de los datos, y despliega un mensaje de paralelo habilitado.



3.4.2.2. DIAGRAMAS DE INTERACCIÓN

El diagrama de interacción es la realización de los diagramas de colaboración, para ello primeramente se elaborará la relación que existe entre los artefactos de la anterior fase como podemos observar en la *Figura 3.22*

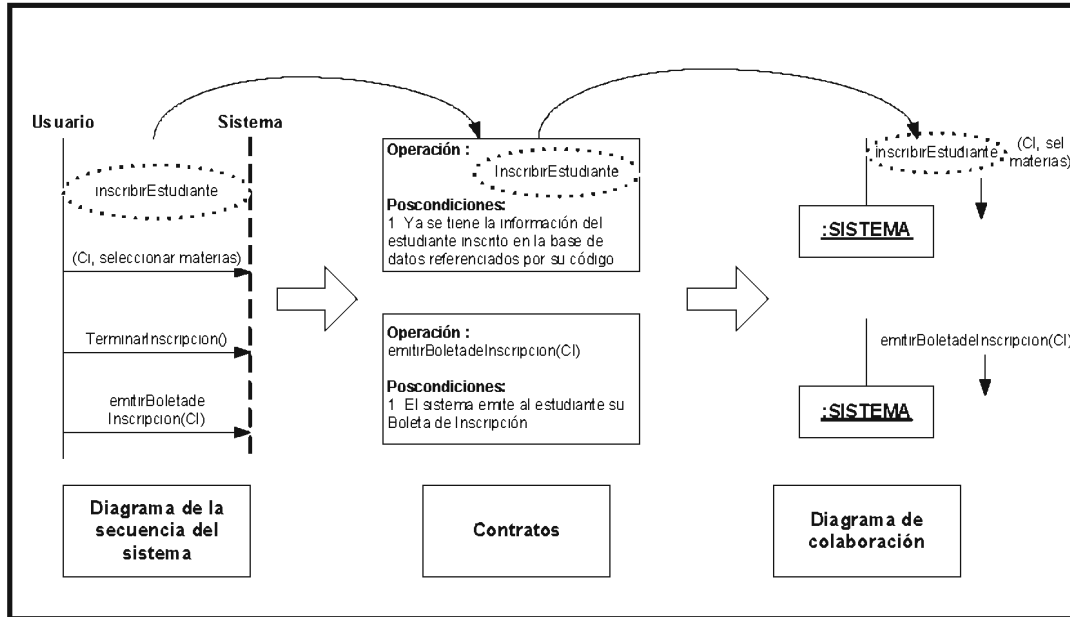


Figura 3.22. Relación entre los artefactos

3.4.2.2.1. DIAGRAMAS DE COLABORACION

El diagrama de colaboración siguiente es la representación gráfica de la **Inscripción**.



Figura 3.23. Diagrama de colaboración Registro de Estudiantes

El proceso de matriculación es representado gráficamente en la siguiente figura.

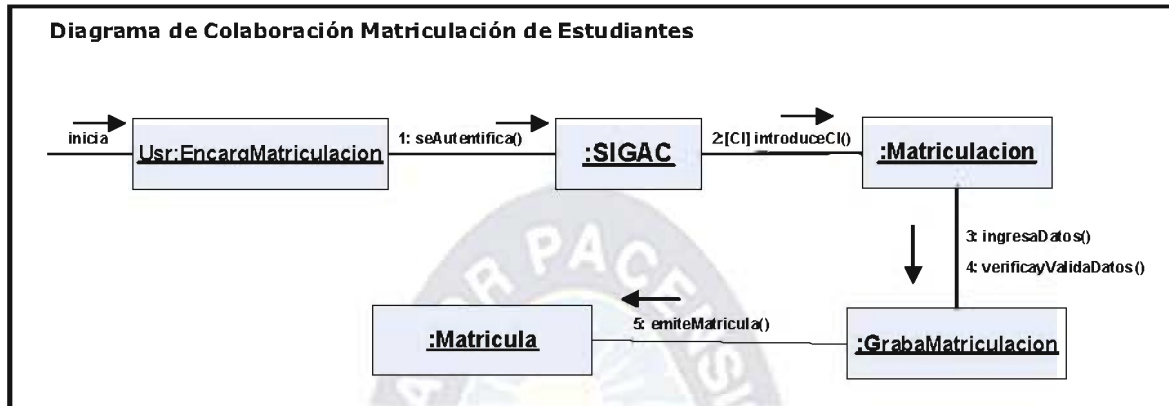


Figura 3.24. Diagrama de colaboración Matriculación de Estudiantes

La figura 3.25 muestra el diagrama de colaboración del proceso de inscripción de Estudiantes



Figura 3.25. Diagrama de colaboración Inscripción de Estudiantes

La siguiente figura especifica el comportamiento de la adición de usuarios



Figura 3.26. Diagrama de colaboración Adición de Usuarios

La siguiente figura muestra el proceso de autenticación del usuario



Figura 3.27. Diagrama de colaboración para la Autenticación

3.4.2.3. DIAGRAMA DE CLASES

En la figura 3.28 se aprecia el diagrama de Clases del Sistema.

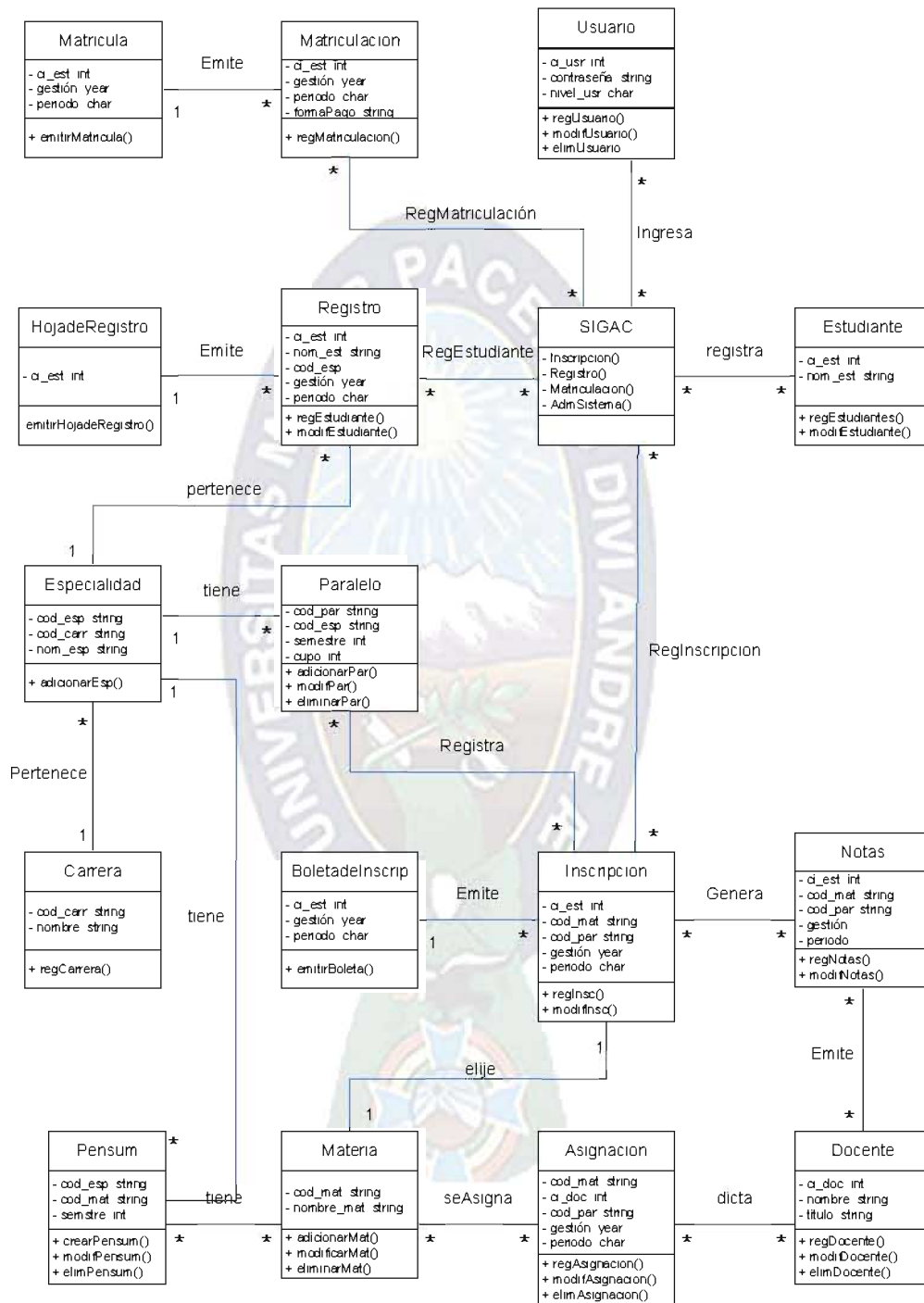


Figura 3.28. Diagrama de Clases del Sistema

Fuente: Elaboración Propia

3.4.2.4. DIAGRAMAS DE ESTADO

Un diagrama de estado en UML nos ayudará a describir visualmente los estados y eventos más interesantes de un objeto, así como su comportamiento ante un evento. De acuerdo a lo realizado en la etapa de análisis se obtiene los siguientes diagramas de estado. En la Figura 3.29 nos muestra el diagrama de estado correspondiente a la verificación de código y password del Usuario, para que este pueda acceder al sistema, siempre y cuando los datos introducidos sean correctos.

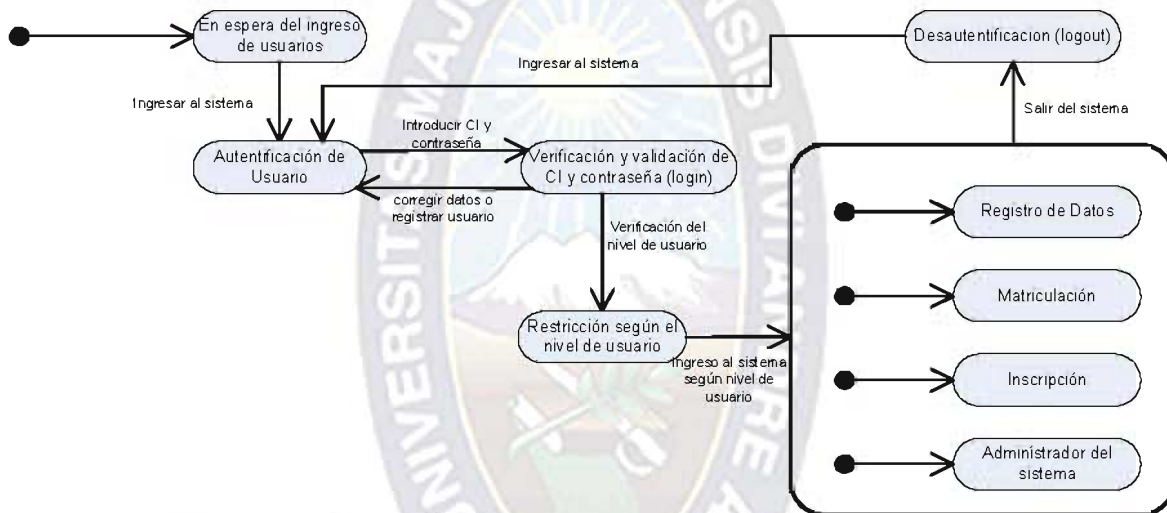


Figura 3.29. Diagrama de Estado correspondiente a la autenticación e ingreso al sistema



Figura 3.30. Diagrama de Estado para el registro de estudiantes nuevos

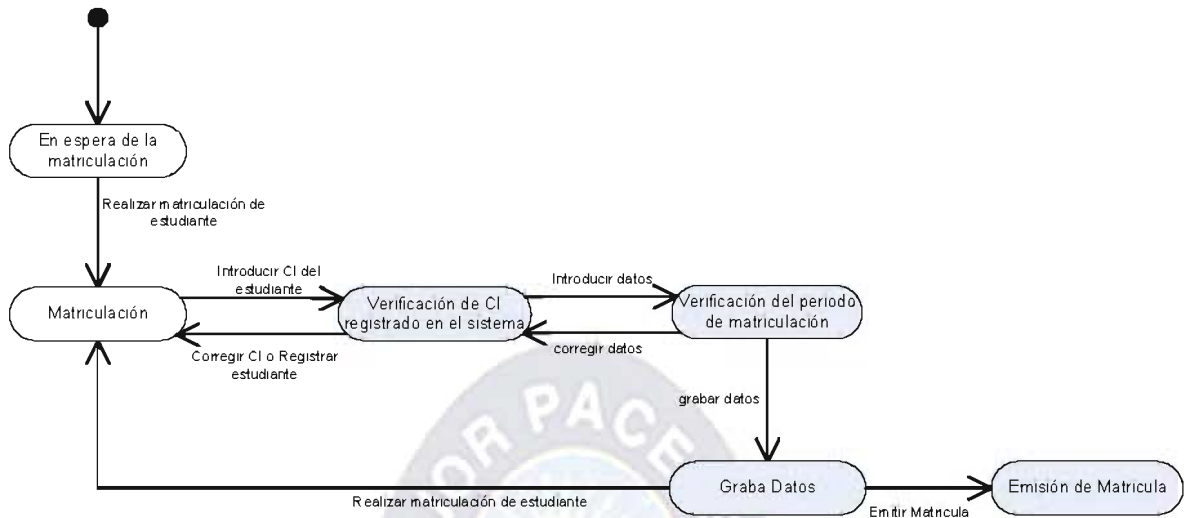


Figura 3.31. Diagrama de Estado para la matriculación de estudiantes

Luego veremos en el diagrama de estados de la Figura 3.32, observamos como es que se presenta los pasos para el proceso de **Inscripción**.

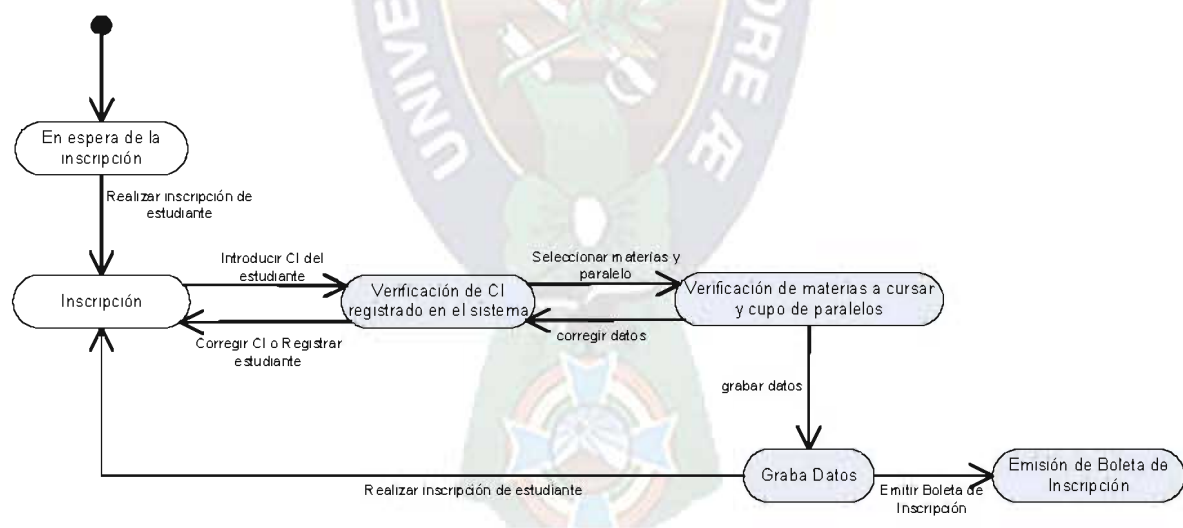


Figura 3.32. Diagrama de Estado correspondiente a la Inscripción

3.4.3. DISEÑO DE LA INTERFAZ DE USUARIO.

La interfaz del usuario será mediante un menú con imágenes que le ayudaran la inscripción, consulta y manipulación de la información de forma confiable e integra.

Para el ingreso a los diferentes módulos y la correspondiente introducción de datos, las interfaces contienen cuadros de texto, zonas de selección (checkbox, radiobutton), botones, iconos, menús desplegables, etc. Para esto el uso del ratón es de vital importancia para facilitar la interacción entre el usuario y el sistema.

A continuación se muestran algunas vistas de pantalla, donde se aprecian los módulos principales que tiene el sistema, además de presentar las opciones que contiene cada módulo.

La primera pantalla del sistema es la de ingreso de usuarios, que se muestra a continuación:



Figura 3.33. Ventana principal de ingreso al sistema

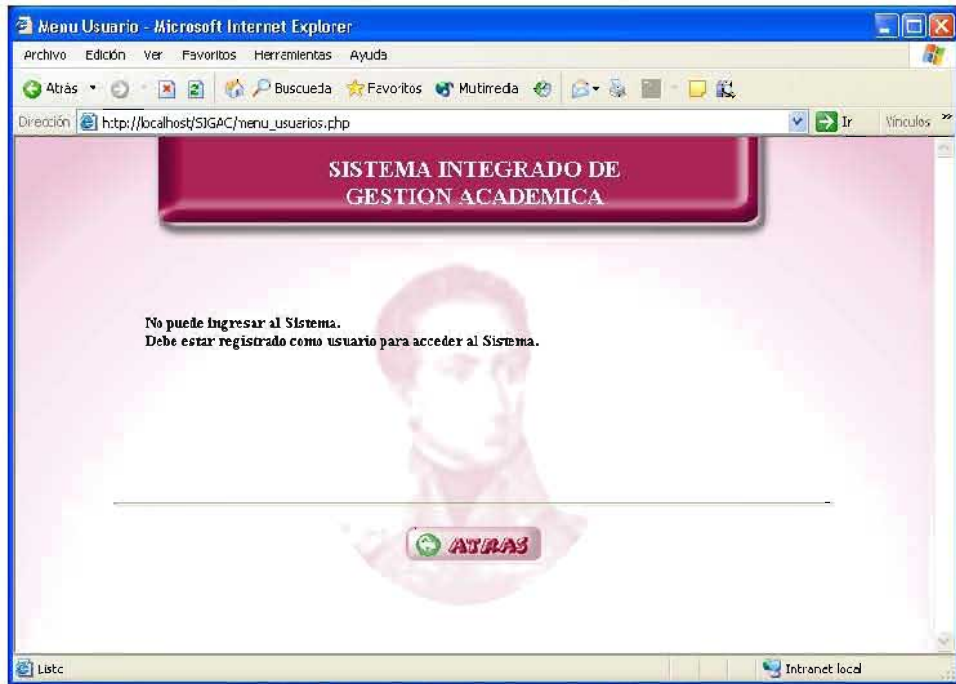


Figura 3.34. Ventana de autenticación incorrecta

Una vez ingresado al sistema, según el nivel de usuario que tenga podrá acceder a los módulos de: Registro de estudiantes nuevos, matriculación de estudiantes, inscripción de estudiantes y administración del sistema.

La pantalla del modulo Registro de estudiantes nuevos se muestra a continuación en la siguiente figura.

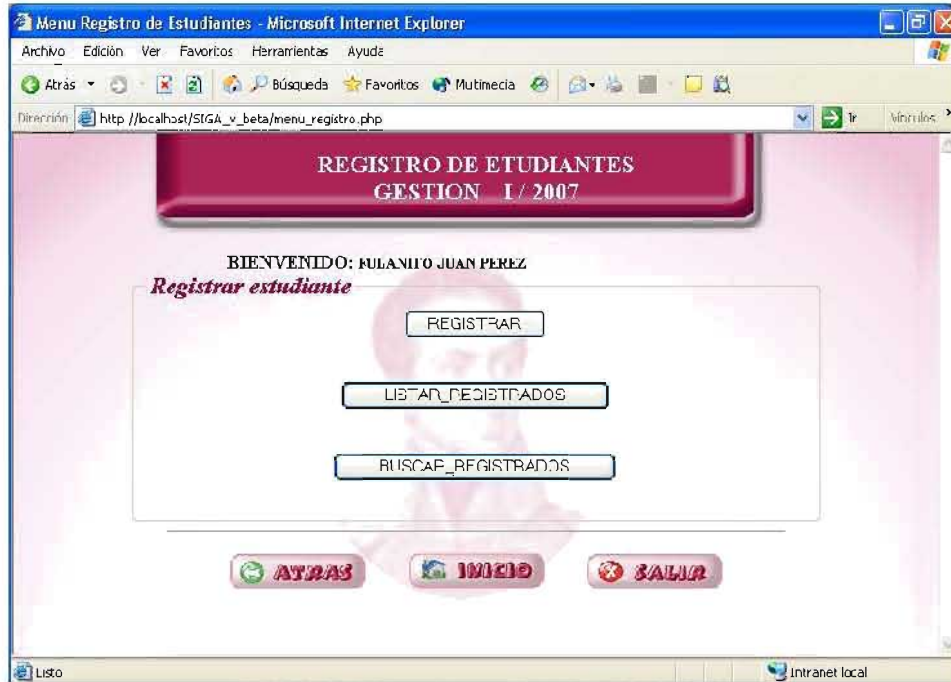


Figura 3.35. Ventana con menú para el proceso de registro



Figura 3.36. Diagrama de Estado para el registro de estudiantes nuevos

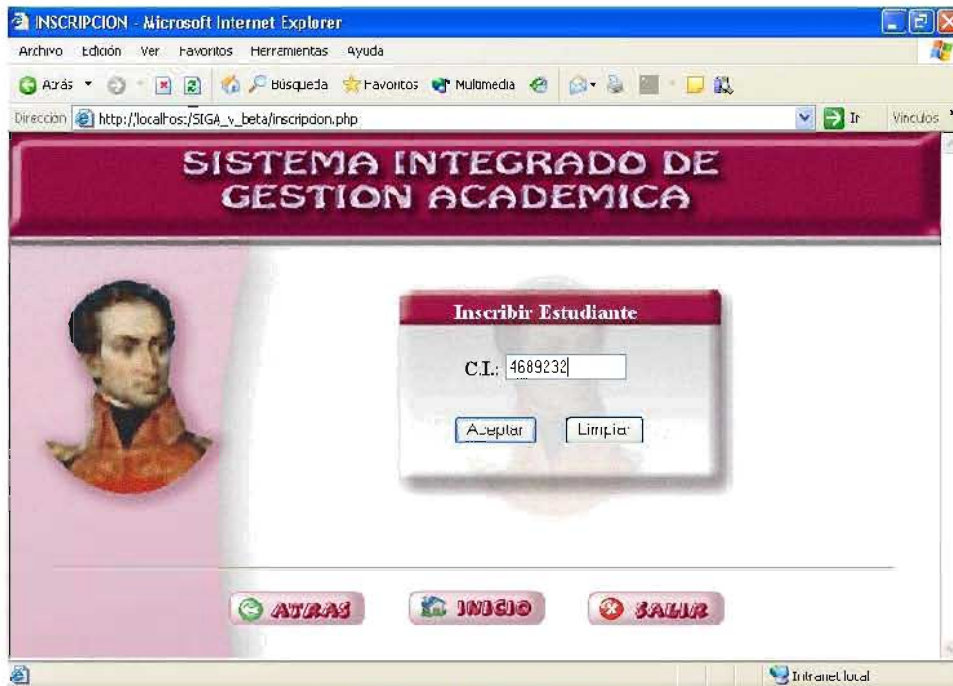


Figura 3.37. Ventana de Inscripción de estudiante



Figura 3.38. Ventana salir del sistema (finalización de Sesión)

3.5. IMPLEMENTACIÓN.

La implementación del Portal de Gestión Académica se basa en los siguientes puntos.

3.5.1 HERRAMIENTAS PARA EL DESARROLLO.

Las herramientas de desarrollo utilizadas en el presente proyecto son multiplataforma, es decir que pueden operar en cualquier sistema operativo.

3.5.1.1. PLATAFORMA DEL SISTEMA

Para un correcto funcionamiento del Sistema, el Servidor Web donde estará instalado el Sistema de Información, debe tener instalado un Sistema Operativo compatible, por ejemplo Linux Red Hat o Fedora, o Windows NT, 2000, XP o Vista.

3.5.1.2. SISTEMA GESTOR DE BASE DE DATOS

El gestor de Base de Datos para el desarrollo del presente Proyecto fue elaborado entorno a Mysql, ya que su aplicación nos facilita el manejo de grandes cantidades de información con rapidez y confiabilidad.

3.5.1.3. HERRAMIENTA DE PROGRAMACIÓN

Para la programación del sistema se utilizó a PHP, como lenguaje de programación que nos ayuda a construir paginas dinámicas que interactúen con bases de datos confiables, además de Javascript para la interacción de elementos de la pagina con los diferentes eventos.

3.5.1.4. HERRAMIENTA DE DISEÑO

Para el diseño de las interfaces se utilizó la aplicación Adobe Photoshop CS, además de la implementación de Hojas de Estilo CSS, que optimiza los recursos del sistema en cuanto a estilos se refiere.

3.5.2. INSTALACIÓN

Todo Software o Sistema Informático para su implantación requiere de cierto tipo de Hardware como de Software, para que su rendimiento sea el óptimo.

Los requerimientos de rendimiento son:

- Tecnología Cliente/Servidor
 - Servidor de Aplicaciones Web (Apache (recomendado), JSP, otros).
 - Servidor de Base de Datos (Postgres, Mysql (recomendado), Oracle, otros).
- La capacidad del sistema se limita a la cantidad de información almacenada.
- Se espera que una consulta no demore mas de 5 segundos

Los requerimientos de software para el servidor son:

- Tener instalado un Sistema Operativo Cliente /Servidor.
- Tener instalado un servidor web.
- Tener instalado un motor de base de datos.
- Tener instalado un navegador de Internet.

Los requerimientos de Hardware para el servidor son:

Procesador:	Pentium IV o D
Disco Duro:	80 Gb o superior
Memoria:	512 Mb o superior
Monitor:	SVGA
Tarjeta de Red:	PCI para RJ-45 o Cable Coaxial
Cable de red:	UTP

Los requerimientos de software para el cliente son:

- Tener instalado un Sistema Operativo Cliente /Servidor.
- Tener instalado un navegador de Internet.

Los requerimientos de Hardware para el cliente son:

Procesador:	Pentium II, III, IV o D
Disco Duro:	40 Gb o superior
Memoria:	256 Mb o superior
Monitor:	VGA o SVGA
Tarjeta de Red:	PCI para RJ-45 o Cable Coaxial
Cable de red:	UTP

3.5.3. PORTABILIDAD

El Sistema Integrado de Gestión Académica: Modulo Registro de Estudiantes esta diseñado para funcionar en distintas plataformas. Por ser una aplicación Web se analiza desde el punto de vista del Cliente y del Servidor.

3.5.3.1. SERVIDOR

El sistema puede funcionar a partir de un equipo Pentium 4 con 512 Mb en memoria RAM y 40 Gb en disco duro.

El sistema operativo del Servidor puede correr en las siguientes plataformas Linux Red Hat, Linux Fedora, Linux SUSE, Linux Knoppix, Windows NT, Windows 2000, Windows XP, Windows Vista.

3.5.3.2. CLIENTE

El sistema puede funcionar a nivel cliente sin inconvenientes, desde un equipo Pentium III con memoria RAM de 128 Mb.

El sistema operativo del Cliente puede ser navegado desde Windows98, Windows Me, Windows NT, Windows XP, Windows Vista y Sistemas Linux.

3.5.4. ESTRUCTURA OPERACIONAL DEL SISTEMA

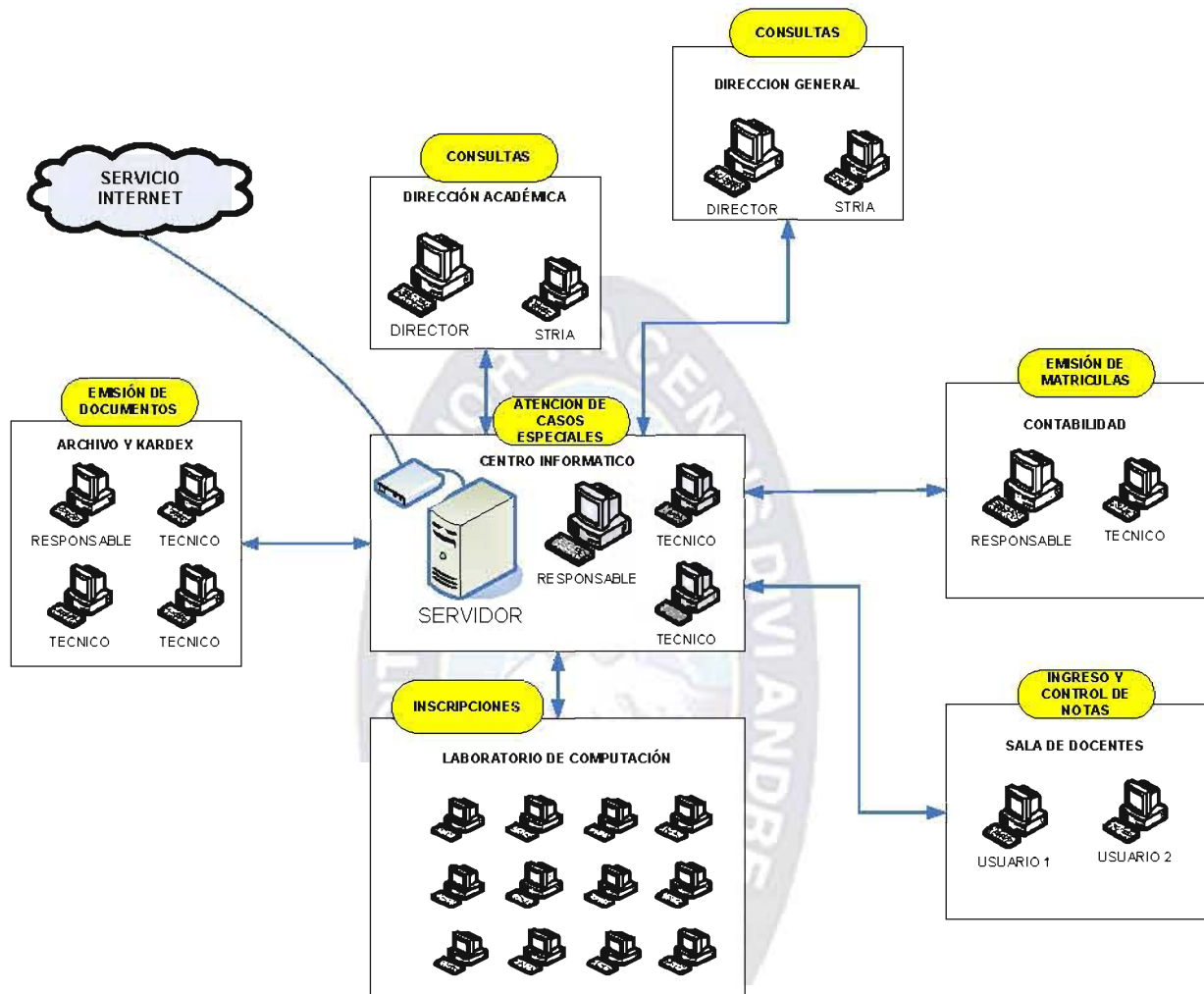


Figura 3.39. Estructura Operacional del Sistema

Fuente: Elaboración propia

En esta figura se puede observar que en la Unidad de Centro Informático es donde se encontrará instalado el Sistema de Gestión Académica, toda la información se encontrará en el servidor desde el cual se podrá solicitar información autorizada a cada Unidad que este interconectada, también se muestra como el Sistema interactúa con las diferentes Unidades, además de los proceso que en cada Departamento son realizados.

CAPÍTULO IV



CALIDAD DEL SISTEMA

CAPÍTULO IV

CALIDAD DEL SISTEMA

4.1. INTRODUCCIÓN

Los desarrollos Web son cada vez más complejos y, además, están creciendo rápidamente, entre ellos las aplicaciones de software centrados en la Web. Por lo tanto la utilización sistemática y disciplinada de métodos, modelos, y técnicas de Ingeniería de Software para el desarrollo, el mantenimiento, y la evaluación de la calidad de los sitios Web debiera ser un requerimiento obligatorio, principalmente en los proyectos de mediana o gran escala. Una de las metas principales de la evaluación y comparación de calidad de artefactos Web, radica en medir, analizar y comprender el grado de cumplimiento de un conjunto de características y atributos con respecto a los requerimientos de calidad establecidos, para un perfil de usuario y dominio de aplicación dados.

La calidad de un producto software debe evaluarse usando un modelo de calidad, midiendo atributos internos (típicamente, medidas estáticas de productos intermedios), o puede ser evaluada midiendo atributos externos (típicamente, medidas del comportamiento del código cuando se está ejecutando). Además, el objetivo de un producto es que tenga el efecto requerido en un contexto de uso particular.

Los modelos de calidad pueden ser utilizados para construir mejores productos y asegurar su calidad. Sin embargo, construir un modelo de calidad es bastante complejo y es usual que estos modelos descompongan la calidad del

producto software jerárquicamente en una serie de características y subcaracterísticas que pueden usarse como una lista de comprobación de aspectos relacionados con la calidad.

En el proceso de evaluación de requerimientos de calidad de artefactos Web complejos, se observa la necesidad de contar con una metodología cuantitativa, integrada, flexible y robusta, que se apoye en principios y prácticas de Ingeniería de Software para la evaluación y comparación de características y atributos, con el fin de obtener resultados objetivos y justificables.

En el presente proyecto de grado el Sistema de Gestión Académica para el Instituto Normal Superior Simón Bolívar utilizará la Metodología de Evaluación de Calidad de Sitios Web (Web Site QEM) propuesta por Luís Antonio Olsina, la cual se basa en el estándar ISO 9126. Posteriormente se efectúa el análisis Costo/Beneficio del producto.

4.2. METODOLOGÍA DE EVALUACIÓN DE CALIDAD DE SITIOS WEB (WEB SITE QEM)

Para la aplicación de esta metodología se deben realizar las tareas propias de cada fase, las cuales son especificadas a continuación:

1. Definiendo el Dominio y Ente para la evaluación de de la calidad
2. Definiendo metas de Evaluación y Seleccionando el perfil de Usuario.
3. Especificando Requerimientos de Calidad para artefactos Web.
4. Definiendo Criterios Elementales e Implementando Procedimientos de Medición
5. Definiendo las estructuras de agregación e implementando la Evaluación de Calidad Global.
6. Analizando y comparando los Resultados parciales y globales

4.2.1. DEFINIENDO EL DOMINIO Y ENTE PARA LA EVALUACIÓN DE LA CALIDAD

Podemos definir al dominio de la aplicación, desde el punto de vista de la evaluación, como a un sistema real o abstracto del universo que existe independientemente del sistema de evaluación. Consiste de un conjunto de entes a los que se le atribuyen propiedades (*atributos, características*), manifiestan un comportamiento y se relacionan.

Por una parte, tenemos a los entes y relaciones propios del dominio de la aplicación; por ejemplo, para este sistema académico podemos considerar entes o clases como *Carrera, Especialidad, Estudiante, Materias*, entre otras, y relaciones entre ellas, como *inscripción, asignación*, etc. Esto se puede observar en la figura 3.27 donde se muestra en el Diagrama de Clases, los entes y relaciones que a su vez contienen métodos y atributos.

4.2.2. DEFINIENDO METAS DE EVALUACIÓN Y SELECCIONANDO EL PERFIL DE USUARIO

4.2.2.1. METAS DE EVALUACION

“Los proyectos sin metas claras no alcanzarán sus metas claramente” [Gilb 88]. En este proceso se deben definir y especificar claramente las metas y el alcance del proceso.

Los objetivos y metas del proceso de evaluación pueden ser muy variados y específicos del tipo de proyecto. Se debe evaluar la calidad de las características de un ente; por ejemplo, la confiabilidad de un artefacto o la calidad global de un sistema completo, en donde intervienen todas las características esenciales. Los resultados podrán ser utilizados para comprender, controlar, mejorar o predecir a la calidad del ente.

Por lo tanto, la pregunta es, ¿a partir de qué enfoque o estrategia seleccionamos las características observables (por ejemplo, navegabilidad,

performancia, usabilidad, etc.)?, ¿Cómo seleccionamos las métricas para esas características y atributos y en qué contexto las analizamos e interpretamos?

Por lo tanto podemos decir que la meta consistió en "*Comprender a la calidad global de un sitio Web desde el punto de vista del visitante general*".

4.2.2.2. PERFILES DE USUARIO CONSIDERANDO LAS METAS Y EL DOMINIO DE APLICACIÓN EN LA WEB.

El estándar mencionado afirma que la relativa importancia de las características de calidad (como usabilidad, funcionalidad, confiabilidad, eficiencia, portabilidad, y mantenibilidad) varían dependiendo del punto de vista considerado y de la criticidad de los componentes del artefacto a evaluar.

Se consideran tres perfiles de usuario, según el estándar ISO [ISO/IEC 9126], para dominios Web, como ser: visitantes, desarrolladores, y gerenciadore. Los visitantes generales, pueden ser descompuestos en dos audiencias más específicas: visitantes casuales y visitantes intencionales. Un *visitante intencional* se define como a la persona que tiene al menos algún conocimiento o manifiesta algún interés en el uso del sistema, y desea probablemente informarse o aprender más acerca del mismo.

Por lo tanto el perfil de usuario seleccionado para este estudio fue el del visitante intencional. En cuanto al punto de vista del visitante, se han identificado a estudiantes actuales y futuros; personal académico, personal administrativo y docentes, donde la visión de estos usuarios, esta centrado en su performancia, su eficiencia, su facilidad de uso, entre otros aspectos. Los usuarios no están interesados en características internas o de desarrollo de los artefactos.

4.2.3. ESPECIFICANDO REQUERIMIENTOS DE CALIDAD PARA ARTEFACTOS WEB

En este paso los se deben acordar y especificar las características y atributos de calidad y agruparlas en un árbol de requerimientos, conforme a las metas, el dominio, y el perfil de usuario seleccionados. Para tal efecto se utilizara el estándar ISO9126 e IEEE 1061.

4.2.3.1. MODELOS DE CALIDAD DE PRODUCTO PRESCRITOS EN LOS ESTÁNDARES ISO 9126 E IEEE 1061

Las ventajas de contar con un modelo de calidad universal serían varias, pero esencialmente nos permitiría evaluar y comparar productos, potencialmente, sobre la misma base. Este estándar describe a la calidad del software, con mínimo solapamiento, a partir de seis características generales.

- ✓ **Funcionalidad**
- ✓ **Confiabilidad**
- ✓ **Usabilidad**
- ✓ **Eficiencia**
- ✓ **Mantenibilidad**
- ✓ **Portabilidad**

Como idea central podemos extraer que esta versión del estándar ISO de características de calidad de producto software, provee una base conceptual muy importante al prescribir dichas características a un alto nivel de abstracción, pero si bien sugiere subcaracterísticas de un modo informativo, *no provee subcaracterísticas y métricas ni métodos para la medición del puntaje y evaluación*”

4.2.3.2. SELECCIONANDO EL ENFOQUE DE MODELO DE CALIDAD

En lo que respecta al enfoque, se adoptara. La noción de un *enfoque de modelo de calidad mixto* que es a la vez, pragmático y flexible. Parte de un modelo fijo (en nuestro caso, a partir del modelo de calidad ISO 9126) en la que

se asume que todas las características necesarias para monitorear un proyecto de evaluación de calidad de producto, son un subconjunto de las seis características publicadas en el estándar. Por otra parte, a nivel de subcaracterísticas (segundo nivel) se definen por consenso entre evaluadores y demás partes intervinientes (posiblemente construido en consideración de subcaracterísticas prescriptas en el estándar para cada característica).

En conjunto, se define el mecanismo de descomposición restante, y se acuerdan las subcaracterísticas de niveles inferiores, los atributos, (los criterios de medición) y las relaciones entre atributos, subcaracterísticas y características., conforme a las metas y el perfil de usuario seleccionado.

4.2.3.3. ÁRBOL DE REQUERIMIENTOS DE CALIDAD GENERAL

Este paso reside en determinar un árbol estándar que pueda ser reusado y personalizado para distintos subdominios de aplicaciones. Contar con un árbol de requerimientos general pero a su vez fácilmente adaptable a diversos dominios, sería de gran beneficio para tareas de aseguramiento y control de calidad

En la siguiente descripción del árbol de requerimientos de calidad se consideran subcaracterísticas y atributos derivados de las características de más alto nivel denominadas: *Usabilidad, Funcionalidad, Confiabilidad, Eficiencia, Portabilidad y Mantenibilidad.*

1. Usabilidad

1.1 Comprensibilidad Global del Sitio

1.2 Mecanismos de Ayuda y Retroalimentación en línea

- 1.2.1 Calidad de la Ayuda
- 1.2.2 Indicador de Última Actualización
- 1.2.3 Directorio de Direcciones
- 1.2.4 *Facilidad FAQ*
- 1.2.5 Retroalimentación

1.3 Aspectos de Interfaces y Estéticos

- 1.3.1 *Cohesividad al Agrupar los Objetos de Control Principales*
- 1.3.2 Permanencia y Estabilidad en la Presentación de los Controles Principales
- 1.3.3 Aspectos de Estilo
 - 1.3.3.1 *Uniformidad en el Color de Enlaces*
 - 1.3.3.2 *Uniformidad en el Estilo Global*
 - 1.3.3.3 *Guía de Estilo Global*
- 1.3.4 *Preferencia Estética*

2. Funcionalidad

2.1 Aspectos de Búsqueda y Recuperación

2.1.1 Mecanismo de Búsqueda en el Sitio Web

2.1.1.1 Búsqueda Restringida

2.1.1.1.1 *de Personas*

2.1.1.1.2 *de Cursos*

2.1.1.1.3 *de Unidades Académicas*

2.1.1.2 Búsqueda Global

2.1.2 Mecanismos de Recuperación

2.2 Aspectos de Navegación y Exploración

2.2.1 Navegabilidad

2.2.1.1 Orientación

2.2.1.1.1 *Indicador del Camino*

2.2.1.1.2 *Etiqueta de la Posición Actual*

2.2.1.2 Promedio de Enlaces por Página

2.2.2 Objetos de Control Navegacional

2.2.2.1 Permanencia y Estabilidad en la Presentación de los Controles Contextuales (Subsitio)

2.2.2.2 Nivel de Desplazamiento

2.2.2.2.1 *Desplazamiento Vertical*

2.2.2.2.2 *Desplazamiento Horizontal*

2.3 Aspectos del Dominio orientados al Estudiante

2.3.1 Relevancia de Contenido

2.3.1.1 Información de Unidad Académica

2.3.1.2 Información de Inscripción

2.3.1.3 Información de Carreras

2.3.1.4 Información de Seguimiento Académico

2.3.2. Servicios Online

3. Confiabilidad

3.1 No Deficiencia

3.1.1 Errores de Enlaces

3.1.1.1 *Enlaces Rotos*

3.1.1.2 *Enlaces Inválidos*

3.1.1.3 *Enlaces no Implementados*

3.1.2 Errores o Deficiencias Varias

3.1.2.1 *Deficiencias o cualidades ausentes debido a diferentes navegadores (browsers)*

3.1.2.2 *Deficiencias o resultados inesperados independientes de browsers*

3.1.2.3 *Nodos Destinos (inesperadamente) en Construcción*

3.1.2.4 *Nodos Web Muertos (sin enlaces de retorno)*

4. Eficiencia

4.1 Performancia

4.2 Accesibilidad

4.2.1 Accesibilidad de Información

4.2.1.1 *Soporte a Versión sólo Texto*

4.2.1.2 Legibilidad al desactivar la Propiedad Imagen del Browser

4.2.1.2.1 *Imagen con Título*

4.2.1.2.2 *Legibilidad Global*

4.2.2 Accesibilidad de Ventanas

5. Portabilidad

4.1 Independencia de Hardware

4.2 Independencia de Software

4.2 Instalabilidad

6. Mantenibilidad

4.1 Expandibilidad

4.2 Estabilidad

4.2.4. DEFINIENDO LOS CRITERIOS DE EVALUACIÓN ELEMENTALES E IMPLEMENTANDO PROCEDIMIENTOS DE MEDICIÓN.

En este proceso, se consideran diferentes tipos de criterios de calidad elemental, escalas (y representación gráfica como escala de preferencia), valores y rangos críticos, y funciones para determinar la preferencia elemental, entre otros asuntos. Una vez definidos y consensuados los criterios para medir cada atributo, se debe ejecutar el proceso de medición, es decir, la recolección de datos, el cómputo de las variables y las preferencias elementales, y la documentación de los resultados.

A partir del árbol de requerimientos, para cada atributo cuantificable A_i (u hoja del árbol) debemos asociar y determinar una variable X_i , que tomará un valor real a partir de un proceso de. Además, para cada variable X_i computada, por medio de un criterio elemental, producirá una preferencia elemental IE_i . Este resultado final, elemental, se puede interpretar como el grado o porcentaje del requerimiento del usuario satisfecho para el atributo A_i .

4.2.5. DEFINIENDO LAS ESTRUCTURAS DE AGREGACIÓN E IMPLEMENTANDO LA EVALUACIÓN DE CALIDAD GLOBAL.

4.2.5.1. EMPLEO DE LOS OPERADORES DE LSP PARA MODELAR RELACIONES LÓGICAS ENTRE CARACTERÍSTICAS Y ATRIBUTOS

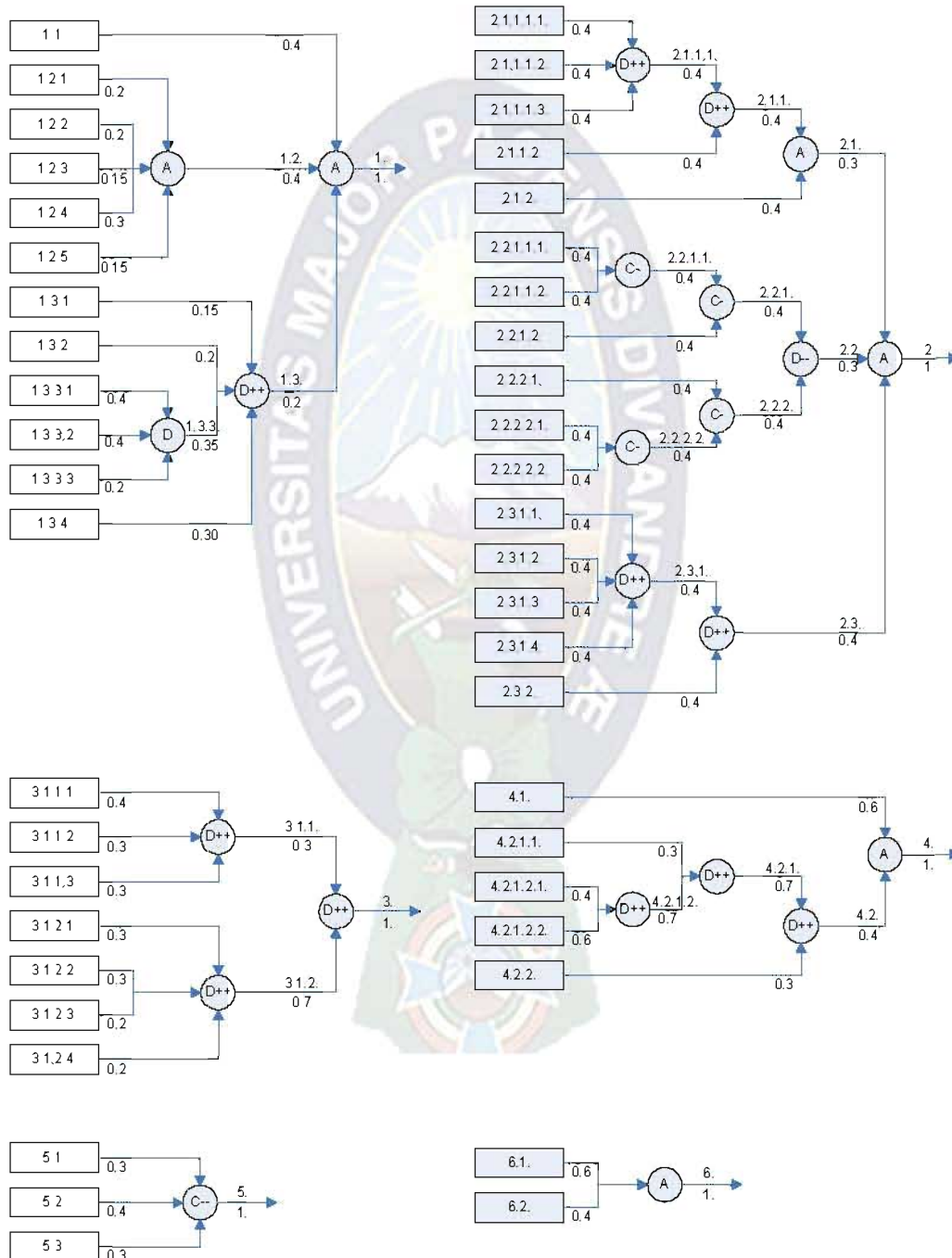


Figura 4.1. Operadores de LSP

4.2.5.2. COMPUTO DE LAS PREFERENCIAS PARCIALES Y GLOBALES

CARACTERÍSTICA	X	IE	R	IE^R	PESO	PESO*IE	TOTAL
1. Usabilidad	78,403	0,7840	1	0,78403	0,3	0,2352	
1 1 Comprensibilidad Global del Sitio	100	1	1	1,00000	0,4	0,4000	
1 2 Mecanismos de Ayuda y Retroalimentación en línea	71,096	0,7110	1	0,71096	0,4	0,2844	
1 2 1 Calidad de la Ayuda	60	0,6	0,526	0,76438	0,2	0,1529	
1 2 2 Indicador de Última Actualización	0	0	0,526	0,00000	0,2	0,0000	
1 2 3 Directorio de Direcciones	0	0	0,526	0,00000	0,15	0,0000	
1 2 4 Facilidad FAQ	50	0,5	0,526	0,69448	0,3	0,2083	
1 2 5 Retroalimentación	500	5	0,526	2,33162	0,15	0,3497	
1 3 Aspectos de Interfaces y Estéticos	49,821	0,4982	1	0,49821	0,2	0,0996	
1 3 1 Cohesividad al Agrupar los Objetos de Control Principales	50	0,5	1,565	0,33798	0,15	0,0507	
1 3 2 Permanencia y Estabilidad en la Presentación de los Controles Pncipales	80	0,8	1,565	0,70524	0,2	0,1410	
1 3 3 Aspectos de Estilo	43,433	0,43433	1,565	0,27114	0,35	0,0949	
1 3 3 1 Uniformidad en el Color de Enlaces	80	0,8	11,095	0,08410	0,4	0,0336	
1 3 3 2 Uniformidad en el Estilo Global	100	1	11,095	1,00000	0,4	0,4000	
1 3 3 3 Guía de Estilo Global	60	0,6	11,095	0,00346	0,2	0,0007	
1 3 4 Preferencia Estética	80	0,8	1,565	0,70524	0,3	0,2116	
2. Funcionalidad	68,655	0,6865	1	0,68655	0,3	0,2060	
2 1 Aspectos de Búsqueda y Recuperación	82,281	0,8228	1	0,82281	0,3	0,2468	
2 1 1 Mecanismo de Búsqueda en el Sitio Web	97,135	0,9714	1	0,97135	0,6	0,5828	
2 1 1 1 Búsqueda Restringida	99,724	0,9972	0,261	0,99928	0,5	0,4996	
2 1 1 1 1 de Personas	60	0,6	0,192	0,90658	0,4	0,3626	
2 1 1 1 2 de Cursos	60	0,6	0,192	0,90658	0,4	0,3626	
2 1 1 1 3 de Unidades Académicas	60	0,6	0,192	0,90658	0,3	0,2720	
2 1 1 2 Búsqueda Global	80	0,8	0,261	0,94342	0,5	0,4717	
2 1 2 Mecanismos de Recuperación	60	0,6	1	0,60000	0,4	0,2400	
2 2 Aspectos de Navegación y Exploración	90,299	0,9030	1	0,90299	0,3	0,2709	
2 2 1 Navegabilidad	90,743	0,9074	1,449	0,86870	0,6	0,5212	
2 2 1 1 Orientación	78,791	0,7879	0,261	0,93968	0,5	0,4698	
2 2 1 1 1 Indicador del Camino	20	0,2	0,261	0,65701	0,4	0,2628	
2 2 1 1 2 Etiqueta de la Posición Actual	60	0,6	0,261	0,87518	0,6	0,5251	
2 2 1 2 Promedio de Enlaces por Página	60	0,6	0,261	0,87518	0,5	0,4376	
2 2 2 Objetos de Control Navegacional	96,832	0,9683	1,449	0,95442	0,4	0,3818	
2 2 2 1 Permanencia y Estabilidad en la Presentación de los	80	0,8	0,261	0,94342	0,4	0,3774	
2 2 2 2 Nivel de Desplazamiento	94,342	0,9434	0,261	0,98491	0,6	0,5909	
2 2 2 2 1 Desplazamiento Vertical	80	0,8	0,261	0,94342	0,5	0,4717	0,9434
2 2 2 2 2 Desplazamiento	80	0,8	0,261	0,94342	0,5	0,4717	

Horizontal							
2 3 Aspectos del Dominio orientados al Estudiante	42,202	0,4220	1	0,42202	0,4	0,1688	
2 3 1 Relevancia de Contenido	70,524	0,7052	1,449	0,60289	0,7	0,4220	
2 3 1 1 Información de Unidad Académica	80	0,8	1,565	0,70524	0,2	0,1410	0,7052
2 3 1 2 Información de Inscripción	80	0,8	1,565	0,70524	0,3	0,2116	
2 3 1 3 Información de Carreras	80	0,8	1,565	0,70524	0,3	0,2116	
2 3 1 4 Información de Seguimiento Académico	80	0,8	1,565	0,70524	0,2	0,1410	
2 3 2 Servicios Online	0	0	1,449	0,00000	0,3	0,0000	
3. Confiabilidad	82,86	0,8286	1	0,82860	0,2	0,1657	
3 1 No Deficiencia	0	0		#NUM!		#NUM!	
3 1 1 Errores de Enlaces	88,621	0,8862	1,449	0,83942	0,3	0,2518	0,8286
3 1 1 1 Enlaces Rotos	60	0,6	0,192	0,90658	0,4	0,3626	0,8862
3 1 1 2 Enlaces Inválidos	40	0,4	0,192	0,83868	0,3	0,2516	
3 1 1 3 Enlaces no Implementados	60	0,6	0,192	0,90658	0,3	0,2720	
3 1 2 Errores o Deficiencias Varias	87,491	0,8749	1,449	0,82396	0,7	0,5768	
3 1 2 1 Deficiencias o cualidades ausentes debido a diferentes	20	0,2	0,153	0,78173	0,3	0,2345	0,8749
3 1 2 2 Deficiencias o resultados inesperados independientes de	30	0,3	0,153	0,83176	0,3	0,2495	
3 1 2 3 Nodos Destinos (inesperadamente) en Construcción	20	0,2	0,153	0,78173	0,2	0,1563	
3.1.2.4 Nodos Web Muertos (sin enlaces de retorno)	20	0,2	0,153	0,78173	0,3	0,2345	
4. Eficiencia	75,63	0,7563	1	0,75630	0,2	0,1513	
4.1 Performancia	80	0,8	1	0,80000	0,6	0,4800	0,7563
4.2 Accesibilidad	69,074	0,6907	1	0,69074	0,4	0,2763	
4.2.1 Accesibilidad de Información	64,391	0,6439	1	0,64391	0,7	0,4507	0,6907
4.2.1.1 Soporte a Versión sólo Texto	10	0,1	0,261	0,54828	0,6	0,3290	0,6439
4.2.1.2 Legibilidad al desactivar la Propiedad Imagen del Browser	40,014	0,4001	0,261	0,78737	0,4	0,3149	0,4001
4.2.1.2.1 Imagen con Título	30	0,3	3,929	0,00882	0,4	0,0035	
4.2.1.2.2 Legibilidad Global	90	0,9	3,929	0,66103	0,6	0,3966	
4.2.2 Accesibilidad de Ventanas	80	0,8	1	0,80000	0,3	0,2400	
5. Portabilidad	83,009	0,8301	3,929	0,48111	0,5	0,2406	
5.1 Independencia de Hardware	80	0,8	0,573	0,87998	0,3	0,2640	0,8301
5.2 Independencia de Software	50	0,5	0,573	0,67222	0,3	0,2017	
5.3 Instalabilidad	85	0,85	0,573	0,91108	0,4	0,3644	
6. Mantenibilidad	72	0,7200	3,929	0,27508	0,5	0,1375	
6. Mantenibilidad	80	0,8000	1	0,80000	0,6	0,4800	0,7200
6. Mantenibilidad	60	0,6000	1	0,60000	0,4	0,2400	

Tabla 4.1. Cómputo de las preferencias parciales

4.2.6. ANÁLISIS Y COMPARACIÓN DE RESULTADOS PARCIALES Y GLOBALES.

Característica	X	IE	R	IE∧R	PESO	PESO*IE	TOTAL
1. Usabilidad	78,403	0,7840	1,565	0,68333	0,3	0,2050	0,7566
2. Funcionalidad	68,655	0,6865	1,565	0,55513	0,3	0,1665	
3. Confiabilidad	82,86	0,8286	1,565	0,74509	0,2	0,1490	
4. Eficiencia	75,63	0,7563	1,565	0,64588	0,2	0,1292	
5. Portabilidad	83,009	0,8301	9,521	0,16982	0,5	0,0849	
6. Mantenibilidad	72	0,7200	9,521	0,04382	0,5	0,0219	

Tabla 4.2. Cómputo de las preferencias globales

De acuerdo a la valoración de la evaluación de la calidad global realizada del “Sistema de Gestión Académica para el Instituto Normal Simón Bolívar” aplicando el modelo Web-Site QEM, se puede concluir que el usuario visitante tendrá una satisfacción de 76 % al utilizar el Sistema. Este valor esta dentro de los márgenes de satisfacción definidos de acuerdo al modelo Web-Site QEM de nivel de puntuación para las métricas.

4.3. BENEFICIO/COSTO

Como su nombre lo sugiere, el método de análisis Beneficio/Costo está basado en la razón de los beneficios a los costos asociada con un proyecto en particular. Se considera que un proyecto es atractivo, cuando los beneficios derivados de su implementación exceden sus costos asociados. Por tanto, el primer paso en un análisis Beneficio/Costo es determinar cuales de los elementos son beneficios y cuales costos.

4.3.1 ANÁLISIS DE COSTOS

Para determinar el costo total del proyecto se tomará en cuenta los siguientes costos:

- Costo del Software desarrollado.
- Costo de implementación del Sistema. conformado por:
 - Costo de software propietario
 - Costo de hardware.
- Costo de elaboración del proyecto.

4.3.1.1 COSTO DEL SOFTWARE DESARROLLADO

Para la determinación del costo del Software desarrollado, se hará uso del Modelo Constructivo de Costo COCOMO II, orientado a los Puntos de Función.

4.3.1.1.1 ESTIMACIÓN DE PUNTOS DE FUNCIÓN

a) Cálculo de Puntos de Función no ajustados

Parámetros de Medición	Cuenta	Factor de Ponderación	Total
Número de entradas de usuario	40	4	160
Número de salidas de usuario	25	5	125
Número de peticiones de usuario	15	4	60
Número de archivos en operación	22	10	220
Número de interfaces externas	0	7	0
Cuenta Total			565

Tabla 4.3. Cálculo de Puntos de Función no ajustados. [PRESSMAN, 2005]

b) Cálculo de valores de ajuste de la complejidad.

Los valores son respondidos usando una escala desde 0 (no importante o aplicable) hasta 5 (absolutamente esencial)

Factor	Valor
Copia de seguridad y recuperación	5
Comunicación de datos	2
Proceso distribuido	3
Rendimiento crítico	3
Entorno operativo existente	4
Entrada de datos en línea (on-line)	4
Transacciones de entrada en múltiples pantallas	4
Archivos maestros actualizados en línea (on-line)	3
Complejidad de valores del dominio de información	2
Complejidad del procesamiento interno	4
Código diseñado para la reutilización	2
Conversión/instalación en diseño	2
Instalaciones múltiples	3
Aplicación diseñada para el cambio	4
$\Sigma (F_i)$	45

Tabla 4.4. Determinación de Complejidad. [PRESSMAN, 2005]

$$\text{Factor de Ajuste} = 0.65 + 0.01 * \sum (F_i)$$

$$\text{Factor de Ajuste} = 0.65 + 0.01 * 45$$

$$\text{Factor de Ajuste} = 1.1$$

c) Cálculo de Puntos de Función

El cálculo de Puntos de Función se basa en la fórmula:

$$\text{PF} = \text{Cuenta Total} * \text{Factor de Ajuste}$$

$$\text{PF} = 565 * 1.1$$

$$\text{PF} = 621.5$$

d) Conversión de los Puntos de Función a KLDC

Ahora se debe convertir los Puntos de Función a miles de líneas de código. Tomar en cuenta la siguiente tabla.

LENGUAJE	NIVEL	Factor LDC / PF
C	2.5	128
Ansi Basic	5	64
Java	6	53
PL/I	4	80
Ansi Cobol 74	3	107
Visual Basic	7.00	46
ASP	9.00	36
<u>PHP</u>	<u>11.00</u>	<u>29</u>
Visual C++	9.50	34

Tabla 4.5. Conversión de Puntos de Función a KLDC

$$\begin{aligned} \text{LDC} &= \text{PF} * \text{Factor LDC/PF} \\ \text{LDC} &= 621.5 * 29 \\ \text{LDC} &= 18023.5 \\ \text{KLDC} &= (18023.5/1000) = 18.02 \end{aligned}$$

e) Aplicación de las fórmulas básicas de esfuerzo, tiempo calendario y personal requerido. Las ecuaciones del COCOMO básico tienen la siguiente forma:

$$E = a_b (KLDC)^{b_b} \dots\dots\dots (\text{Ecuación } 1)$$

$$D = c_b (E)^{d_b} \dots\dots\dots (\text{Ecuación } 2)$$

Donde:

E: Esfuerzo aplicado en personas por mes

D: Tiempo de desarrollo en meses cronológicos

KLDC: Número estimado de líneas de código distribuidas (en miles)

Proyecto de Software	a_b	c_b	b_b	d_b
Orgánico	2.4	1.05	2.5	0.38
Semi - acoplado	3.0	1.12	2.5	0.35
Empotrado	3.6	1.20	2.5	0.32

Tabla 4.6. Coeficientes a_b y c_b y los exponentes b_b y d_b. [PRESSMAN, 2005]

Son proyectos de software semi-acoplados, los proyectos intermedios (en tamaño y complejidad) en los que equipos, con variados niveles de experiencia, deben satisfacer requisitos poco o medio rígidos, tal es el caso del software desarrollado.

$$E = 3.0 * (18.02)^{1.12}$$

$$D = 2.5 * (56.5)^{0.35}$$

$$E = 56.5$$

$$D = 11.4$$

El personal requerido, en este caso el número de programadores se obtiene con la siguiente fórmula:

Número de programadores = E / D

Número de programadores = $56.5 / 11.4$

Número de programadores = $4.95 \approx 5$

El salario de un programador puede oscilar entre los \$us 150, cifra que será tomada en cuenta para la estimación siguiente:

Costo del software desarrollado = Número de programadores * salario de un programador

Costo del software desarrollado por persona = $5 * 150$
= 750 \$us

Costo total del software desarrollado = $750 * 11.4$
= 8550 \$us

Costo del Software desarrollado = \$us 8550

4.3.1.2 COSTO DE IMPLEMENTACIÓN DEL SISTEMA

Descripción	Costo Total (\$us)
Windows XP SP2	150
Windows Me	100
TOTAL	250

Tabla 4.7. Costo de Software Propietario

Cantidad	Descripción	Costo Unitario (\$us)	Costo Total (\$us)
25	Terminal Pentium IV, 450 Mhz, 128 RAM, HD 20Gb, Color SVGA	500	12500
1	Servidor Pentium D, 3.0 Ghz, 1024 RAM, HD 120 Gb, lector CD, Color SVGA	6000	6000
3	Hub inteligentes de 8 puertos	130	390
2	Hub inteligentes de 16 puertos	218	436
1	Match Panel de 24 puertos	35	35
1	Rack Armario metálico (1.8 x 0.74 x 0.52 m.)	94	94
1	UPS (15min - 20min)	500	500
26	Tarjeta NE2000 Ethernet	17	442
500mts.	Cable UTP cat. 5e	0.4	200
26	Patch cords de 2m.	3.5	91
26	Jacks RJ45	5	130
26	Placas para Jack RJ45 de un conector	3.5	91
26	Cortapicos	3	78
4	Impresora matricial	150	600
6	Impresora a chorro	60	360
Total			21947

Tabla 4.8. Costo de Hardware requerido

Detalle	Importe (\$us)
Costo de Software Propietario	250
Costo de hardware requerido	21947
Total	22197

Tabla 4.9. Costo de implementación del Sistema

4.3.1.3 COSTO DE ELABORACIÓN DEL PROYECTO

Los costos de elaboración del proyecto se refieren a los costos del estudio del sistema, en la etapa de recopilación y análisis principalmente, estos costos se presentan en la siguiente tabla.

DETALLE	IMPORTE (\$us)
Análisis y diseño del proyecto	300
Bibliografía	50
Material de Escritorio	30
Internet	40
Otros	20
TOTAL	440

Tabla 4.10. Costo de elaboración del proyecto

Fuente.- Elaboración propia

4.3.1.4 COSTO TOTAL

El costo total es la sumatoria del costo del Software desarrollado, costo de implementación del Sistema y costo de elaboración del proyecto, detallados en la siguiente tabla 4.11.

Detalle	Importe (\$us)
Costo del Software desarrollado	8550
Costo de implementación del Sistema	22197
Costo de elaboración del proyecto	440
Total	31187

Tabla 4.11. Costo Total

Fuente: Elaboración propia

4.3.2 ANÁLISIS DE BENEFICIOS

Los beneficios para el presente proyecto son de tipo intangibles. Para realizar el análisis de beneficios se utilizaron cinco criterios de evaluación, los cuales permiten evaluar el sistema diseñado respecto al sistema anterior. Ellos son:

- Incremento de velocidad en los procesos.
- Capacidad en el volumen de información.
- Control de procesos.
- Integración de la información.
- Información para la toma de decisiones.

Entre los retornos de inversión, podemos citar los siguientes:

- Reducción de costos de transacción, administración y mantenimiento.
- Mayor productividad.
- Mejor calidad en la información para la toma de decisiones.
- Optimización de procesos.
- Reducción de tiempos de resolución por disponibilidad inmediata de información consolidada de usuarios.
- Mejora de calidad en atención al Estudiante, Apoderado.

4.3.2.1 INCREMENTO DE VELOCIDAD EN LOS PROCESOS

La tabla 4.12 muestra la relación de tiempos de ejecución entre el sistema diseñado y el antiguo obtenido durante la etapa de prueba del sistema. La misma que demuestra el incremento de velocidad de procesos, aliviando en gran medida la realización de tareas repetitivas, lo que permite la obtención de información rápida y oportuna.

4.3.2.2 CAPACIDAD EN EL VOLUMEN DE INFORMACIÓN

El Sistema de Gestión Académica permite almacenar grandes volúmenes de información académica como financiera y personal de los usuarios, empleando medios magnéticos de gran capacidad de almacenamiento (unidades en Gigabytes) para el registro académico estudiantil. El almacenamiento de 3000

registros se la efectúa a una razón de 200 Mb por año, por lo que se garantiza el espacio suficiente para el almacenamiento de la información. El sistema cuenta también con un componente de resguardo para salvaguardar la información.

4.3.2.3 CONTROL DE PROCESOS

Los beneficios que se plasman en la tabla 4.12, demuestran que los procesos que conforma el Sistema de Gestión Académica son exactos, consistentes y efectivos. Esto se evidencia más en los procesos de: resguardo de la información y en el procesamiento de cuadros estadísticos.

4.3.2.4 INTEGRACIÓN DE LA INFORMACIÓN

La red institucional de información, implementado conjuntamente al Sistema de Gestión Académica asegura la integridad de la información permitiendo mayor coordinación entre departamentos involucrados con las tareas de seguimiento y control de Estudiantes, Padres de Familias (Apoderados), docentes y Administrativos. Al mismo tiempo se garantiza la oportunidad y veracidad de la información generada por el sistema, puesto que se registran en el lugar de origen y en tiempo real.

4.3.2.5 INFORMACIÓN PARA LA TOMA DE DECISIONES

El sistema implementado, cuenta con una variedad de informes y cuadros estadísticos a nivel general y de detalle, los mismos que son producidos en tiempos oportunos para la previsión y planificación de planes académicos.

Criterios de eval.	1.1.2 Beneficios	Tiempo promedio de Procesamiento	
		Sistema antiguo	POGAW
Incremento de velocidad	<ul style="list-style-type: none"> Localización de estudiantes Atención a consultas de Docentes. Atención a consultas de Estudiantes. Atención a consultas de Personal Administrativo. 	Trabajo manual y por lotes 5 - 10 min. (procesos por lote) 25 - 30 min. Información ambigua	Trabajo automático en línea 0.5 - 1 min. (Proceso autom.) 1 - 5 min. Información fidedigna

Capacidad en el volumen de información	<ul style="list-style-type: none"> • 50 atenciones diarias/promedio, en consulta a docentes • 150 atenciones diarias/promedio, acceso a registros de estudiantes y otros servicios. 	<p>50 hojas entre consultas y solicitudes</p> <p>100 hojas entre registros, consultas y solicitudes</p>	La información se almacena en un disco magnético, de 80 Gb, de capacidad, que es capaz de almacenar toda la información generada.
Control de procesos	<ul style="list-style-type: none"> • Registro de información • Procesamiento de cuadros estadísticos 	<p>Existe un error del 30% por no contar con un registro adecuado en cada uno de los procesos que involucra al sistema actual</p> <p>El 30% de los cuadros procesados presentan errores de cálculo de datos estadísticos</p>	<p>Disminuye el error, el sistema permite el registro de la información de manera confiable y ordenada de tal manera que no existan confusiones</p> <p>No existen errores, el cálculo de los datos estadísticos se realizan de manera automática</p>
Integración de información	<ul style="list-style-type: none"> • Coordinación entre los procesos que están involucrados con el sistema de gestión académica. 	La coordinación no es la mas apropiada, ya que la misma se efectúa a través de personas que podrían involuntariamente extravíar documentación.	No existe error, porque se cuenta con una conexión de red de área local, la cual permite integrar toda la información generada por los diferentes usuarios en tiempos oportunos y el almacenamiento de información para consultas posteriores.

<p>Información para la toma de decisiones</p>	<ul style="list-style-type: none"> • Información para fuentes externas • Información para directivos 	<p>El tratamiento y procesamiento de la información necesaria para estas actividades suele demorar tiempos prolongados que perjudican a la toma de decisiones.</p>	<p>El sistema actual facilita información necesaria y requerida por el usuario de forma rápida y eficiente.</p>
---	--	--	---

Tabla 4.12. Cuadro de Beneficios

Fuente: Elaboración propia



CAPÍTULO V



CONCLUSIONES Y RECOMENDACIONES

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

Terminado el proceso de construcción del sistema, se concluye que:

A la culminación del análisis, diseño y desarrollo, se cumplió satisfactoriamente con la etapa de implementación, luego de realizar la capacitación al personal encargado del manejo del Sistema, llegando a cumplir con los objetivos establecidos inicialmente, obteniendo así un producto con las particularidades de un sistema académico.

El Sistema desarrollado gestiona en forma automática los procesos académicos que permitan agilizar las tareas realizadas en el Área Académica del Instituto Normal Superior Simón Bolívar, proporcionando información confiable, eficiente y oportuna, logrando cumplir adecuadamente con los requisitos impuestos por los usuarios finales.

El Sistema de Gestión realiza eficientemente los procesos de admisión de estudiantes semestralmente (registro de estudiantes nuevos a la base de datos, inscripción, matriculación) emitiendo la Hoja de Ingreso Personal, Boleta de Inscripción e impresión de la Matrícula de estudiantes.

La implementación del módulo Registro y Control de notas permite realizar el registro de Notas de manera automática y confiable, permitiendo que el área de Archivo y Kardex pueda emitir documentos: certificado de notas, historial de

notas, globales, listas oficiales y boletines de notas de forma eficiente y rápida, facilitando la organización de la información y su manejo, permitiendo realizar un mejor trabajo,.

La implementación del módulo de consultas de estudiantes permite a los estudiantes realizar su seguimiento académico (materias inscritas, notas del último periodo, historial de notas, pensum y registro personal), mostrando transparencia en el manejo de la información, evitando susceptibilidades y realizando el prestigio del INSSB.

Los reportes y cuadros comparativos (detalle de estudiantes por especialidad, sexo, edad, idioma, tipo de colegio, etc.), permite conocer la situación general de las diferentes carreras y especialidades, ayudando a la toma de decisiones a nivel ejecutivo.

De acuerdo a la utilización de la Metodología Craig Larman apropiada para el análisis y diseño orientado a objetos, la utilización de estrategias y guías de desarrollo para aplicaciones cliente/servidor; además de la formación profesional adquirida a lo largo de cinco años en la carrera de Informática de la UMSA de la ciudad de La Paz, se concluye que el desarrollo del presente proyecto de grado tiene consistencia técnica y asegura la obtención de los resultados planteados inicialmente.

5.2. RECOMENDACIONES

En cuanto al análisis y diseño del sistema, cuando la institución requiera incluir un nuevo módulo, se recomienda primero revisar el documento para poder tomar una buena decisión, ya que el diseño del sistema representa componentes reutilizables que pueden ayudar a reutilizar el diseño del sistema actual.

Luego de realizar la observación y análisis en la Institución se recomienda la integración de otros departamentos a la red para conformar una red institucional completa y por consiguiente la expansión y aprovechamiento de los servicios de Red Local, como ser:

La integración del Sistema de Gestión Académica con la sección de contabilidad, para determinar los estados financieros de esta área, permitiendo llevar a cabo un manejo más eficiente de la parte contable del INSSB.

Realizar un sistema de control de personal biométrico, que agilice y automatice este proceso, ya que actualmente el INSSB cuenta solo con un marcador de tarjetas, el cual debe ser registrado manualmente cada mes.

Implementar un sistema de consultas en la Biblioteca, que ayude a realizar rápidamente la búsqueda de material bibliográfico a los estudiantes del INSSB.

Diseñar y desarrollar un portal web que permita informar no solo a la comunidad normalista de las actividades y publicaciones del INSSB, sino a la sociedad en general, en la que tanto estudiantes como docentes puedan expresar las diferentes inquietudes que deseen expresar a través de este portal.

BIBLIOGRAFÍA

- LIBROS

- [LARMAN, 1999] Larman. Craig. *UML y Patrones*, Primera Edición. Prentice Hall. Mexico 1999.
- [PRESSMAN, 2005] Pressman, R. *Ingeniería de Software*, Quinta edición. Editorial Mc Graw Hill. Mexico. 2004.
- [JACOBSON, 1999] Jacobson I. *El proceso unificado de desarrollo de software*. Mc. Graw Hill Interamericana de España 2000.
- [BUNGE, 1979] Bunge, Mario. *La investigación científica*. Editorial Ariel de Barcelona 1969.
- [KENDALL, 1997] Kendall, Keneth E.; Kendall, Julie E. *"Análisis y Diseño de Sistemas"*. Tercera Edición. Prentice Hall Hispanoamericana S.A. 1997.
- [HERNANDEZ, 2003] Hernandez Sampier, Roberto, *Metodología de la Investigación*, Tercera edición. Editorial Mc Graw Hill. México. 2003.

- PUBLICACIONES

- [RIVERO, DURAN, COCA & BELLIDO, 2005] Rivero E., Duran R., Coca N., Bellido L., *Guia para Proyectos de Grado de la Carrera de Ingeniería de Sistemas*, Guía de Proyecto de Grado, Sucre-Bolivia. 2005.
- [LOZADA, 2003] Lozada Pereira, Blithz, *Manual de Funciones Instituto Normal Superior Simón Bolívar*, Impresión Sigla SRL
- [LOZADA, 2003] Lozada Pereira, Blithz, *Manual de Procedimientos Instituto Normal Superior Simón Bolívar*, Impresión Sigla SRL

- ARTICULOS WEB

- [OLSINA, 1999] OLSINA, Luis Antonio, 1999: *Metodología Cuantitativa para la Evaluación y Comparación de la Calidad de Sitios Web*. Tesis Doctoral. La Plata - Argentina

- **[CAMACHO, 2001]** Camacho, Hugo y otros. *El enfoque del Marco Lógico: 10 casos prácticos*. Cuaderno para la identificación y diseño de proyectos de desarrollo. <http://www.umsanet.edu.bo/docentes/gchoque>
- **[MOLPECERES, 2003]** Molpeceres, Alberto, Procesos de desarrollo: RUP, XP y FDD <http://www.iavahispano.org/licencias/>
- **[LETELIER, 2003]** P. Letelier, *Rational Unified Process (RUP)*. Departamento de Sistemas Informáticos y Computación. Universidad Politécnica de Valencia. <https://pid.dsic.upv.es>. 2003.

- REFERENCIA WEB

- **[BID]** Banco Interamericano de Desarrollo, *Marco Lógico para el Diseño de Proyectos*, Accedido en septiembre 2006. <http://www.iadb.org/int/rtc/ecourses/esp>

- PROYECTOS DE GRADO

- **[PEREZ Y LAURA, 2001]** Perez Elizabeth y Laura Wendy, 2001: *Sistema de Seguimiento Académico Instituto Normal Superior Simón Bolívar*, T358, Proyecto de Grado, La Paz-Bolivia.
- **[CHACOLLA, 2005]** Chacolla Quispe, Víctor, 2003: *Sistema de Gestión Académica Vía Intranet: Inscripciones y Consultas para el Instituto Normal Superior Simón Bolívar*, T169, Proyecto de Grado, La Paz-Bolivia.
- **[TARQUI, 2003]** Tarqui Carpio, Nelson Egberto, 2003: *Sistema de Seguimiento Académico Universidad Tecnológica Boliviana (UTB)*, T169, Proyecto de Grado, La Paz-Bolivia.
- **[FLORES, 2005]** Flores, Jesús Lalo, 2005: *Portal de Gestión Académica Estudiantil vía Web*, T1129 Proyecto de Grado, La Paz-Bolivia.
- **[SIRPA, 2005]** Sirpa Caceres, Rocio de la Azucena 2005: *Sistema de Gestión Académica Universitaria Caso: Facultad de Ingeniería*, T1241, Proyecto de Grado, La Paz-Bolivia.
- **[FLORES, 2003]** Flores Condori, Freddy 2003: *Control y Seguimiento Académico Web para la Empresa Educomser SRL*, T1879, Proyecto de Grado, La Paz-Bolivia