

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



**PROYECTO DE GRADO**

**“SEGUIMIENTO Y CONTROL DE UNIDADES EDUCATIVAS  
CASO: DIRECCIÓN DISTRITAL DE EDUCACIÓN LA PAZ-2”**

**PARA OPTAR AL TÍTULO DE LICENCIATURA EN INFORMATICA  
MENCION: INGENIERIA DE SISTEMAS INFORMATICOS**

**POSTULANTE: Univ. LUIS ALBERTO MENDOZA MENDOZA  
TUTOR METODOLOGICO: M. Sc. FATIMA CONSUELO DOLZ DE MORENO  
ASESOR: LIC. EUFREN LLANQUE QUISPE**

**LA PAZ – BOLIVIA  
2014**



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



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

**LICENCIA DE USO**

El usuario está autorizado a:

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

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

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

## **DEDICATORIA**

A mis padres Pablo y Teodocia por el apoyo y amor incondicional brindado durante mis años de estudio y toda mi vida.

A mis hermanos Edgar, Teddy, Hugo, Ema, Nelly y Estela por la ayuda que siempre me otorgan.

## **AGRADECIMIENTOS**

Ante todo a Dios por guiar todos mis pasos y darme la fuerza necesaria para seguir adelante.

A la Universidad Mayor de San Andrés, que me permitió una formación profesional y en la cual viví momentos que me acompañaran por siempre.

A mi tutora Metodológica, M. Sc. FÁTIMA CONSUELO DOLZ DE MORENO, por su guía, constante apoyo y paciencia.

A mi asesor Lic. Eufren Llanque Quispe por brindarme sus conocimientos, su tiempo y su guía en la realización del proyecto.

A todo el personal de la Dirección Distrital de Educación La Paz-2, por la ayuda brindada durante el desarrollo y la implementación del proyecto.

A mis compañeros de carrera por todos los momentos vividos.

## **RESUMEN**

La Distrital de Educación La Paz-2, tiene como principal función la de fortalecer el accionar de los responsables de cada una de las Unidades Educativas del distrito 2, en cuanto al desarrollo curricular y administrativo.

En la dirección Distrital de Educación La Paz-2, se pudo observar el manejo poco eficiente de los registros de las Unidades Educativas, el problema principal se basa en que el control de las Unidades Educativas pertenecientes al Distrito- 2, no llena las expectativas del responsable de la Institución, debido a que los procesos se realizan de forma manual, haciendo difícil realizar el seguimiento de los registros de control dentro de la Institución.

El Sistema de Información de Seguimiento y Control para la Dirección Distrital De Educación La Paz – 2, fue precisamente desarrollado para subsanar las deficiencias en el control de las Unidades Educativas, teniendo como objetivo principal de mejorar la administración de la información.

Una vez realizado el sistema se realizaron las pruebas de funcionamiento, en las cuales se puede constatar que el sistema responde a los requerimientos, logrando de esta manera otorgar un mejor servicio a los usuarios.

# ÍNDICE

	Pág.
<b>CAPÍTULO 1. MARCO REFERENCIAL</b> .....	1
1.1 INTRODUCCIÓN .....	1
1.2 ANTECEDENTES.....	2
1.3 IDENTIFICACIÓN DEL PROBLEMA .....	4
1.4 PLANTEAMIENTO DEL PROBLEMA.....	5
1.5 OBJETIVOS .....	5
1.5.1 OBJETIVO GENERAL.....	5
1.5.2 OBJETIVOS ESPECÍFICOS.....	5
1.6 ALCANCES.....	5
1.7 JUSTIFICACIÓN .....	6
1.7.1 JUSTIFICACIÓN SOCIAL.....	6
1.7.2 JUSTIFICACIÓN TÉCNICA.....	6
1.7.3 JUSTIFICACIÓN ECONÓMICA .....	7
1.8 APORTES.....	7
1.9 METODOLOGÍA Y HERRAMIENTAS .....	7
<b>CAPÍTULO 2. MARCO TEÓRICO</b> .....	9
2.1 INTRODUCCIÓN .....	9
2.2 DIRECCIÓN DISTRITAL DE EDUCACIÓN LA PAZ-2.....	9
2.3 METODOLOGÍA DEL DESARROLLO DEL SISTEMA.....	11
2.3.1 SISTEMA.....	11
2.3.2 INFORMACIÓN.....	12
2.3.3 SISTEMA DE INFORMACIÓN .....	12
2.3.4 MÉTODO.....	13
2.3.5 DIAGRAMA.....	13
2.3.6 MODELO.....	14
2.3.7 MODELO VISUAL .....	14
2.3.8 METODOLOGÍA .....	15
2.4 LENGUAJE UNIFICADO DE MODELADO – UML.....	15
2.4.1 CARACTERÍSTICAS DE UML .....	15
2.4.1.1 DIAGRAMA DE CASOS DE USO .....	16
2.4.1.2 DIAGRAMA DE SECUENCIA.....	18
2.4.1.3 DIAGRAMA DE COLABORACIÓN.....	18

2.4.1.4	DIAGRAMA DE ACTIVIDADES .....	19
2.4.1.5	DIAGRAMA DE ESTADOS .....	19
2.4.1.6	DIAGRAMA DE COMPONENTES.....	19
2.4.1.7	DIAGRAMA DE DESPLIEGUES.....	20
2.4.1.8	DIAGRAMA DE CLASES .....	20
2.5	METODOLOGÍA ICONIX .....	20
2.5.1	CARACTERÍSTICAS DE ICONIX.....	21
2.5.2	FASES DE CICLO DE VIDA DE ICONIX.....	21
2.5.3	DESCRIPCIÓN DE LAS FASES.....	21
2.5.3.1	ANÁLISIS DE REQUERIMIENTOS.....	21
2.5.3.2	ANÁLISIS Y DISEÑO PRELIMINAR .....	22
2.5.3.3	DISEÑO .....	23
2.5.3.4	IMPLEMENTACIÓN .....	23
2.6	HERRAMIENTAS DE IMPLEMENTACIÓN.....	24
2.6.1	PHP.....	24
2.6.2	MYSQL .....	24
2.6.3	SERVIDOR APACHE.....	25
2.7	BASE DE DATOS .....	26
2.7.1	SISTEMA DE ADMINISTRACIÓN DE BASE DE DATOS.....	27
2.8	CALIDAD DE SOFTWARE .....	27
2.8.1	OPERACIONES DE UN PRODUCTO.....	28
2.8.2	REVISIÓN DEL PRODUCTO.....	28
2.8.3	TRANSICIÓN DEL PRODUCTO .....	29
2.9	FACTORES DE CALIDAD ISO 9126.....	29
2.9.1	PRUEBAS DE SOFTWARE.....	30
2.9.1.1	PRUEBAS DE CAJA BLANCA .....	30
2.9.1.2	PRUEBAS DE CAJA NEGRA.....	31
	<b>CAPÍTULO 3. MARCO APLICATIVO.....</b>	<b>32</b>
3.1	ANÁLISIS DE LA SITUACIÓN.....	32
3.2	FASE DE ANÁLISIS DE REQUERIMIENTOS .....	32
3.2.1	FUNCIONES DEL SISTEMA .....	35
3.2.2	ANÁLISIS DE RIESGOS.....	36
3.3	DESARROLLO CASOS DE USO .....	36
3.3.1	ACTORES DEL SISTEMA .....	37
3.3.2	CASOS DE USO EXPANDIDOS.....	40

3.3.3 DEFINIENDO EL MODELO CONCEPTUAL PRELIMINAR.....	41
3.3.4 DEFINIENDO LOS CASOS DE USO EXPANDIDOS .....	41
3.3.5 DEFINIENDO DIAGRAMA DE ACTIVIDADES .....	44
3.3.6 DEFINIENDO DIAGRAMA DE SECUENCIA.....	47
3.3.7 DEFINIENDO CONTRATO DE OPERACIONES .....	49
3.8.8 DEFINIENDO INTERFAZ DE USUARIO .....	52
3.4 FASES DE IMPLEMENTACIÓN.....	53
3.4.1. PERFECCIONAMIENTO DE LA ARQUITECTURA DEL SISTEMA.....	54
3.4.2. DISEÑO DE LA BASE DE DATOS .....	54
3.4.3. DEFINIENDO EL ESQUEMA DE LA BASE DE DATOS .....	55
3.4.4. IMPLEMENTACIÓN DE LOS MÉTODOS.....	61
<b>CAPÍTULO 4. PRUEBAS Y CALIDAD DE SOFTWARE .....</b>	<b>65</b>
4.1 INTRODUCCIÓN .....	65
4.2 PRUEBAS DEL SISTEMA .....	65
4.2.1 PRUEBA DE CAJA BLANCA.....	65
4.2.2 PRUEBA DE CAJA NEGRA.....	70
4.3 CALIDAD DE SOFTWARE .....	72
CAPÍTULO 5. CONCLUSIONES Y RECOMENDACIONES .....	78
BIBLIOGRAFÍA.....	80
ANEXOS .....	82
ÁRBOL DE PROBLEMAS.....	82
ÁRBOL DE OBJETIVOS .....	83
MATRIZ DE MARCO LÓGICO .....	84



## ÍNDICE DE IMÁGENES

	<b>Pág.</b>
Figura 2.1 Estructura Institucional .....	10
Figura 2.2 Enfoque Diseño de Caja Blanca .....	31
Figura 2.3 Enfoque Diseño de Caja Negra .....	31
Figura 3.1 Delimitación del Sistema en base a los actores .....	38
Figura 3.2 Diagrama de Casos de Uso 1: Operaciones del Operador .....	38
Figura 3.3 Diagrama de Casos de Uso 2: Operaciones U.E .....	39
Figura 3.4 Diagrama de Casos de Uso 3: Operaciones de Sistemas .....	39
Figura 3.5 Modelo Conceptual Preliminar .....	41
Figura 3.6 Diagrama de Actividad: Ingreso al Sistema .....	44
Figura 3.7 Diagrama de Actividad: Creación de una U.E. ....	45
Figura 3.8 Diagrama de Actividad: Edición de una U.E. ....	45
Figura 3.9 Diagrama de Actividad: Eliminar una U.E .....	46
Figura 3.10 Diagrama de Secuencia: Gestionar Unidad Educativa .....	47
Figura 3.11 Diagrama de Secuencia: Registrar Unidad Educativa .....	48
Figura 3.13 Arquitectura del Sistema de Información .....	54
Figura 3.14 Diagrama de Entidad/Relación .....	55
Figura 4.1 Grafo de Flujo-Búsqueda de Unidad Educativa .....	67

## ÍNDICE DE TABLAS

	<b>Pág.</b>
Tabla 3.1 Cronograma de actividades .....	34
Tabla 3.2 Funciones del Sistema .....	35
Tabla 3.3 Análisis de riesgos .....	36
Tabla 3.4 Actores del Sistema .....	37
Tabla 3.5 Caso de Uso Esencial: Ingresar Nuevo operador .....	44
Tabla 3.6 Contrato: Adición de una Nueva Unidad Educativa .....	49
Tabla 3.7 Contrato: Modificar Unidad Educativa .....	50
Tabla 3.8 Contrato: Buscar Unidad Educativa .....	51
Tabla 3.9 Contrato: Mensaje.....	52
Tabla 3.10 Operador .....	56
Tabla 3.11 Unidad Educativa .....	57
Tabla 3.12 Circulares e Instructivos .....	58
Tabla 3.13 Técnico .....	59
Tabla 3.14 Director .....	60
Tabla 3.15 Procedimiento de Listado de Unidades Educativas.....	64
Tabla 4.1 Prueba de Caja Negra .....	71
Tabla 4.2. Calculo Punto Función .....	73
Tabla 4.4. Ponderación de evaluación .....	75
Tabla 4.5 Cuestionario en el aspecto de usabilidad.....	76

### 1.1 INTRODUCCIÓN

Los Sistemas de Información y las Tecnologías de Información están cambiando la forma en que operan las organizaciones actuales. A través de su uso se logran importantes mejoras, tales como la automatización de los procesos operativos, tareas administrativas.

Las Tecnologías de la Información se conceptualizan como la integración y convergencia de la computación, las telecomunicaciones y la técnica para el procesamiento de datos, donde sus principales componentes son: el factor humano, los contenidos de la información, el equipamiento, la infraestructura, el software y los mecanismos de intercambio de información, los elementos de política y regulaciones, además de los recursos financieros.

Los componentes mencionados conforman los protagonistas del desarrollo informático en una sociedad, tanto para su desarrollo como para su aplicación, además se reconoce que las tecnologías de la información constituyen el núcleo central de una transformación multidimensional que experimenta la economía y la sociedad; de aquí lo importante que es el estudio y dominio de las influencias que tal transformación impone al ser humano como ente social, ya que tiende a modificar no sólo sus hábitos y patrones de conducta, sino, incluso, su forma de pensar.

La Dirección Distrital de Educación de La Paz - 2 es una institución dedicada a diseñar paradigmas para el cumplimiento de las Bases, fines, objetivos y políticas educativas coherentes del Sistema Educativo Nacional con una administración y gestión colegiada por redes articuladas con compromiso de excelencia para asegurar y lograr procesos educativos con calidad y eficiencia.

La Dirección Distrital, cuya actividad principal es garantizar la administración de la educación pública y privada, planificando, organizando, orientando y evaluando los procesos educativos. Actualmente aún cuenta con alguna información, registrada y almacenada

manualmente lo cual implica una pérdida de tiempo, además de no hacer conocer rápidamente los circulares e instructivos.

La finalidad principal del trabajo es solucionar el problema del manejo y control de información de las distintas unidades educativas implementando un Sistema de Seguimiento de Control con el cuál la información será almacenada eficientemente.

El presente proyecto pretende optimizar el manejo, control y seguimiento en la Dirección Distrital de Educación La Paz - 2, a fin de proporcionar información correcta, confiable y adecuada a las necesidades de los altos directivos de la institución.

## **1.2 ANTECEDENTES**

En la actualidad la Dirección Distrital de Educación La Paz - 2, cuenta con una unidad de cómputo que realiza reportes mensuales de las actividades que realizan las distintas unidades educativas pertenecientes al distrito - 2, los reportes están realizados en aplicaciones de Microsoft Office, lo cual significa pérdida de tiempo en la búsqueda de información.

El propósito del proyecto es mejorar procesos manuales, complejos, morosos por otros procesos totalmente automatizados que devuelvan resultados de alta confiabilidad y en un corto periodo de tiempo.

El objetivo de la Dirección Distrital de Educación La Paz - 2 es mejorar la Administración Institucional, Técnico-Pedagógico y Comunidad Educativa, por medio de diferentes actividades programadas en el Plan para mejorar la calidad de Recursos Humanos durante la gestión.

Además de garantizar la administración de la educación pública y privada, planificando, organizando, orientando y evaluando los procesos educativos con un equipo Técnico Distrital cualificado y Consejo Técnico Distrital comprometidos en el cumplimiento de objetivos propuestos.

La Dirección Distrital de Educación La Paz - 2 fortalece los sistemas de gestión técnica administrativa y financiera del SEDUCA, a través de la optimización de los recursos (humanos, materiales, infraestructura) y tecnológica para mejorar la calidad educativa, a su

vez fortalecer el accionar de los responsables de las Unidades Educativas y su personal, durante la gestión escolar en cuanto al desarrollo curricular y administrativo.

De acuerdo a los aspectos administrativo, técnico-pedagógico y escuela-comunidad, a partir de los programas de descentralización administrativa, definidas en la ley 1178 y 1565, y sus decretos complementarios, y de acuerdo a la Ley U.S.P.A. (Ley de Uso de Suelos y Patrones de Asentamiento) Nro. 557 Donde se establece que geográficamente se encuentran los Macro Distritos Municipales 7 (Centro), 5 (Sur) de la ciudad de La Paz, Provincia Murillo, del Departamento de La Paz, por lo que sus habitantes de las diferentes zonas del Distrito por sus características pertenecen a la clase alta media y media-baja, por lo que la población escolar está constituido por los hijos de padres de ingresos económicos variables.

La Dirección Distrital de Educación La Paz - 2 mantiene independencia administrativa con iniciativa y responsabilidad, en las Unidades Educativas del Distrito, implementa en las Unidades Educativas del Distrito el apoyo permanente del Equipo Técnico Distrital en los diferentes aspectos como: Administrativos, Técnico Pedagógico Cultural y la Comunicación Interinstitucional.

En la Carrera de Informática perteneciente a la Facultad de Ciencias Puras y Naturales de la Universidad Mayor de San Andrés se han desarrollado similares proyectos de grado:

- “SISTEMA DE INFORMACIÓN WEB PARA EL CONTROL Y SEGUIMIENTO DE LAS ACTIVIDADES EDUCATIVAS DE LOS ESTUDIANTES” [QUENTA.2008], trabajo que tiene como objetivo brindar información acerca de los estudiantes y sus calificaciones así como también información de los docentes y sus materias.
- “SISTEMA DE INFORMACIÓN DE SEGUIMIENTO Y CONTROL DE HISTORIALES CLÍNICOS DEL CONSULTORIO PSICOLÓGICO DEL CEMSE” [CHIPANA.2009], que tiene como objetivo desarrollar un sistema que contempla consulta, tratamientos, exámenes, seguimiento y control interno de consultorio, inventarios y estadísticas.

Como se puede notar no son muchos los proyectos desarrollados. En nuestro caso el Sistema de Información de Seguimiento y Control de la Dirección Distrital de Educación La Paz - 2 pretende llevar un control eficaz de la información de las unidades educativas, reportando circulares e instructivos los cuales proporcionarían datos para medir la eficacia de este servicio, además agilice y procese la información adecuadamente.

### **1.3 IDENTIFICACIÓN DEL PROBLEMA**

Después de haber realizado el análisis correspondiente a la Dirección Distrital de Educación La Paz – 2, se observa que presenta problemas en su funcionamiento.

En una primera instancia al realizar la recopilación de la información, se identifican los siguientes problemas:

- La búsqueda de información de las unidades educativas pertenecientes a la Dirección Distrital de Educación se realiza de forma manual.
- Registro de datos de las unidades educativas con errores en el RUE (Sic y Programa).
- Inexistencia del seguimiento de instructivos y circulares.
- Inexistencia del seguimiento adecuado de las unidades educativas
- La recepción y seguimiento de informes diarios es de forma manual

Los problemas mencionados tienen una relación respectiva de causa y efecto, conformando una diversidad de otros problemas secundarios. Por tanto los problemas se originan porque los procesos que se realizan manualmente tienen como consecuencia la falta de una administración adecuada de la información, caracterizando esta por no ser precisa ni oportuna debido a que la información es almacenada en archivos registrados manualmente.

## **1.4 PLANTEAMIENTO DEL PROBLEMA**

El manejo y control de la información de las unidades educativas pertenecientes a la Dirección Distrital de Educación La Paz - 2 requiere mayor eficiencia ya que no permite optimizar el registro y control de la información de las Unidades Educativas

## **1.5 OBJETIVOS**

### **1.5.1 OBJETIVO GENERAL**

Desarrollar e implementar un sistema de información y control bajo medidas de seguridad, confiabilidad y rapidez en la administración de los procedimientos de información que cubra el funcionamiento, control y registro de toda la información referente a las unidades educativas pertenecientes a la Dirección Distrital de Educación La Paz - 2.

### **1.5.2 OBJETIVOS ESPECÍFICOS**

- Proporcionar una interfaz amigable para el usuario, fácil y entendible para uso del personal.
- Proporcionar una información confiable de las distintas unidades educativas del distrito - 2, mediante una base de datos que contenga los datos más sobresalientes de la unidad educativa, tratamiento y otros.
- Disposición inmediata de la información para el personal autorizado.
- Elaboración de instructivos y circulares.
- Uso de normas de seguridad y calidad para el diseño de sistemas de información.
- Presentación acorde con el estándar de las aplicaciones del cliente en cuanto a tipo de letras, colores, logos, mensajes, usabilidad.

## **1.6 ALCANCES**

El alcance del proyecto pretende los siguientes procesos:

El Sistema de Seguimiento y Control para la Dirección Distrital de La Paz - 2 pretende lograr un manejo eficaz y eficiente de los registros de información de las unidades educativas, evitando procesos manuales que producen pérdida de tiempo.

El proyecto contempla las siguientes actividades:

- Modificar o actualizar datos de alguna unidad educativa.
- Consultar información de una unidad educativa.
- Imprimir los datos de la unidad educativa.
- Consultar y editar datos de las unidades educativas.
- Colocar circulares a través del portal

## **1.7 JUSTIFICACIÓN**

### **1.7.1 JUSTIFICACIÓN SOCIAL**

El presente proyecto beneficia a los funcionarios de la Dirección Distrital de Educación La Paz – 2, ya que se tendrá un adecuado seguimiento y control de información de las unidades educativas, que a su vez podrán obtener la información oportuna de los instructivos y los estudiantes que deben realizar trámites administrativos.

### **1.7.2 JUSTIFICACIÓN TÉCNICA**

Es necesario instrumentar un mecanismo que automatice los procesos administrativos de la Dirección Distrital de Educación La Paz - 2, el proyecto se justifica por qué se hace posible la aplicación de tecnología informática a un problema real observado en la Dirección Distrital de manera que la manipulación de la información de las unidades educativas sean segura y confiable dentro de la Dirección Distrital de Educación La Paz - 2.



### **1.7.3 JUSTIFICACIÓN ECONÓMICA**

El Sistema de Información de Seguimiento y control de las Unidades Educativas permite a la Dirección Distrital de Educación La Paz – 2 optimizar sus principales tareas, mejorando el tiempo de servicio por el sistema que será implementado en red, que permite al personal de la Dirección realizar consultas.

### **1.8 APORTES**

El proyecto tiene como aporte principal el desarrollo de un sistema de seguimiento y control de los registros de información que realicen las tareas de:

- Circulares que emite el sistema con información acertada y oportuna.
- Automatización de los procesos rutinarios.
- Minimizar y optimizar tiempos de ejecución.
- Seguridad, control y registro de cualquier acceso de usuarios al sistema, además de proveer códigos de acceso y mantenimiento de los mismos.

### **1.9 METODOLOGÍA Y HERRAMIENTAS**

La metodología que se utiliza para el desarrollo del presente proyecto se enfoca al paradigma orientado a objetos, utilizando ICONIX para el modelado del sistema. Las razones de elección del método es que es lo suficientemente flexible para realizar cambios como introducir o eliminar elementos de la notación UML además proporciona un ciclo de vida iterativo e incremental lo cual es muy apropiado para el desarrollo del sistema.

A continuación se detalla la metodología en las siguientes etapas:

- Etapa de recopilación de información.
  - Entrevistas
  - Consultas y estado de cada servicio de apoyo.
- Etapa de análisis, diseño y desarrollo del sistema

- Se aplica la metodología ICONIX (*Use case driven object modeling with UML. A practical approach*).
- Herramienta Lenguaje Unificado de Modelado (*Unified Modeling Language*) que permite visualizar, construir, y documentar los elementos de un sistema.
- Etapa de programación
  - PHP. (*Hipertext preprocessor*), es un lenguaje de programación de estilo clásico de código abierto muy popular, adecuado para desarrollo web y que puede ser incrustado en HTML. Es popular porque un gran número de páginas y portales web están creadas con PHP. Código abierto significa que es de uso libre y gratuito para todos los programadores que quieran usarlo.
  - MySQL. *Database Server*, es un sistema de administración relacional de bases de datos. Una base de datos relacional archiva datos en tablas separadas en vez de colocar todos los datos en un gran archivo. Esto permite velocidad y flexibilidad. Las tablas están conectadas por relaciones definidas que hacen posible combinar datos de diferentes tablas sobre pedido.
  - APACHE, es el servidor web hecho por excelencia, su configurabilidad, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa. El servidor Apache se desarrolla dentro del proyecto HTTP Server (*httpd*) de la *Apache Software Foundation*.
- Etapa de prueba de calidad de software
  - Pruebas de caja blanca.
  - Pruebas de caja negra.
  - Muestras de calidad.

## 2.1 INTRODUCCIÓN

El presente capítulo señala las herramientas necesarias para el desarrollo y la implementación del Sistema de Información de Seguimiento y Control para la Dirección Distrital de Educación La Paz -2 mediante una sustentación teórica en referencia a los objetivos propuestos.

## 2.2 DIRECCIÓN DISTRITAL DE EDUCACIÓN LA PAZ-2

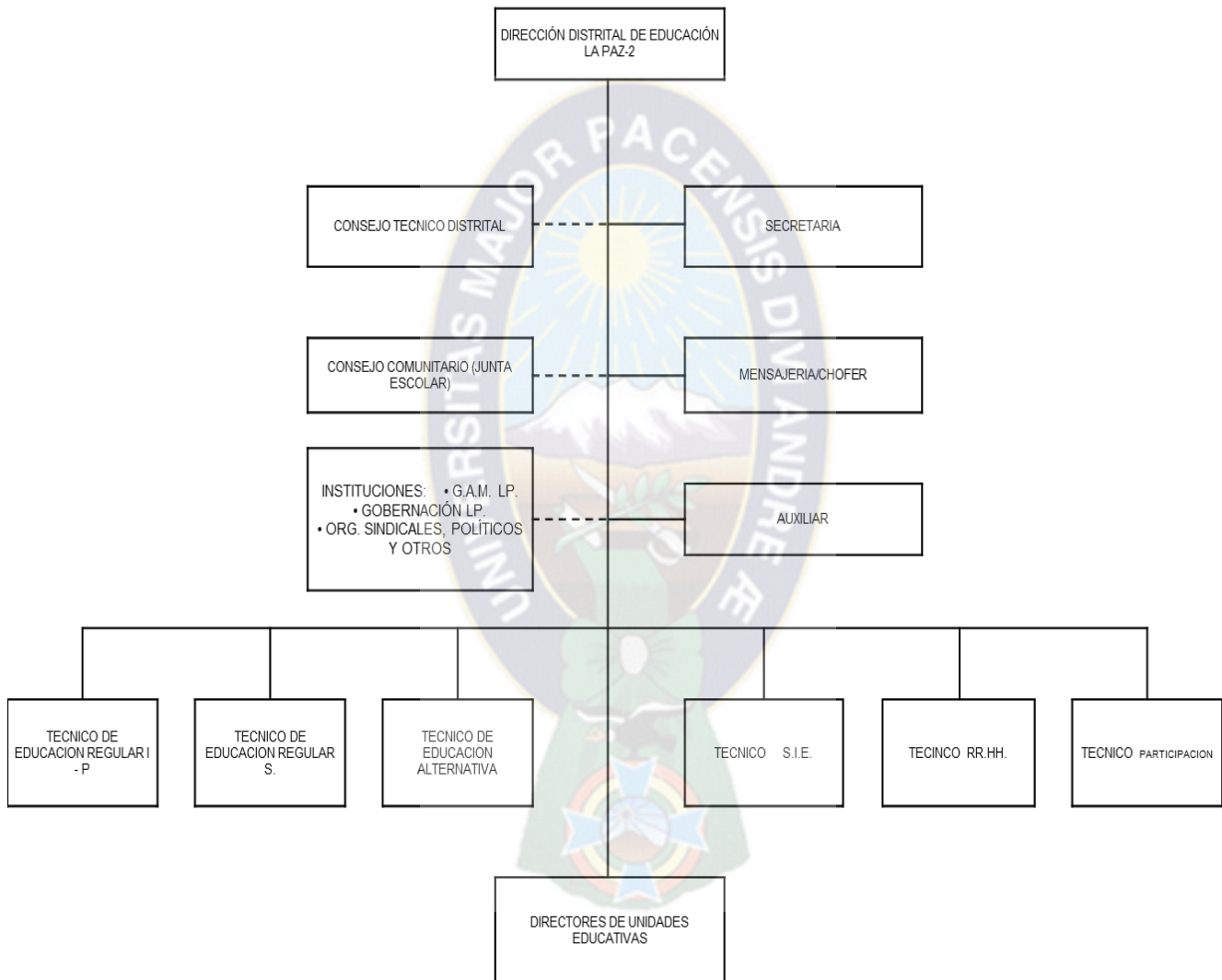
La Dirección Distrital de Educación La Paz-2, en estricto cumplimiento de las normas establecidas, es una Unidad Operativa encargada de hacer cumplir las normas y reglamentaciones de las Políticas Educativas del Ministerio de Educación del Estado Plurinacional de Bolivia.

La finalidad de la Institución es la de diseñar, implementar y ejecutar políticas, estrategias educativas inclusivas, equitativas, interculturales, plurilingües, científicas, técnica – tecnología de calidad, con participación social desde el ámbito territorial, comunitario productivo y descolonizador a través del Sistema Educativo Plurinacional.

A su vez la Dirección Distrital de Educación La Paz-2 tiene la visión de diseñar paradigmas para el cumplimiento de las bases, fines, objetivos y políticas educativas coherentes del Sistema Educativo Plurinacional, con una administración y gestión colegiada a través de redes articuladas, comprometidas con los procesos educativos para alcanzar la calidad, eficiencia en beneficio de los niños y jóvenes educandos.

A continuación se muestra la Estructura Organizacional de la Institución (Ver figura 2.1)

## DIRECCIÓN DISTRITAL DE EDUCACIÓN LA PAZ-2



**Figura 2.1 Estructura Institucional**

**Fuente: Dirección Distrital de Educación La Paz - 2**

## **2.3 METODOLOGÍA DEL DESARROLLO DEL SISTEMA**

Una Metodología para el Desarrollo de Sistemas de Información es un conjunto de actividades llevadas a cabo para desarrollar y poner en marcha un Sistema de Información.

Los Objetivos de las Metodologías de Desarrollo de Sistemas de Información son:

- Definir actividades a llevarse a cabo en un Proyecto de S.I.
- Unificar criterios en la organización para el desarrollo de S.I.
- Proporcionar puntos de control y revisión.

A su vez la metodología de desarrollo del Sistema consiste en las herramientas, modelos y métodos para asistir al proceso de desarrollo de software que la desarrollamos a continuación.

### **2.3.1 SISTEMA**

Un Sistema es un grupo de componentes que pueden funcionar recíprocamente para lograr un propósito común. Son capaces de reaccionar juntos al ser estimulados por influencias externas, el sistema no está afectado por sus propios egresos y tiene límites específicos en base de todos los mecanismos de retroalimentación significativos [SPEDDING, 1979].

Según O'Brien (2001) en un sentido más amplio un sistema es un grupo de componentes interrelacionados que trabajan en conjunto hacia una meta en común mediante la aceptación de entradas y generando salidas en un proceso de transformación organizado [O' BRIEN, 2001].

De lo mencionado un Sistema es un conjunto de partes coordinadas y en interacción para alcanzar un conjunto de actividades. Los sistemas reciben (entrada) datos, energía o materia del ambiente y proveen (salida) información, energía o materia.

### **2.3.2 INFORMACIÓN**

Información es un conjunto de datos con un significado, o sea, que reduce la incertidumbre o que aumenta el conocimiento de algo. En verdad, la información es un mensaje con significado en un determinado contexto, disponible para uso inmediato y que proporciona orientación a las acciones por el hecho de reducir el margen de incertidumbre con respecto a nuestras decisiones [CHIAVENATO, 2006].

Son datos evaluados para individuos concretos que trabajan en un momento dado sobre un problema especial para alcanzar un objetivo [ALVAREZ, 2002].

Teniendo en cuenta las anteriores ideas y definiciones la información es un conjunto de datos acerca de algún suceso, que organizados tienen su significado, cuyo propósito poder ser el de reducir la incertidumbre o incrementar el conocimiento de algo.

### **2.3.3 SISTEMA DE INFORMACIÓN**

Un sistema de información es un conjunto de elementos interrelacionados con el propósito de prestar atención a las demandas de información de una organización, para elevar el nivel de conocimientos que permitan un mejor apoyo a la toma de decisiones y desarrollo de acciones [PEÑA, 2006].

El sistema que examina y recupera los datos provenientes del ambiente que captura los datos a partir de las transacciones y operaciones efectuadas dentro de la empresa que filtra, organiza y selecciona los datos y los representa en forma de información a los gerentes, proporcionándoles los medios para generar la información [MURDICK, 1998]

Un sistema de información realiza cuatro actividades básicas: entrada, almacenamiento, procesamiento y salida de información.

Entrada de información, es un proceso mediante el cual el sistema de información toma los datos que requiere para procesar la información. Las entradas pueden ser manuales o automáticas. Las manuales son aquellas que se proporcionan en forma directa por el usuario, mientras que las automáticas son datos o información que provienen o son tomados de otros sistemas o módulos; llamado interfaces automáticas.

Almacenamiento de información, el almacenamiento es una de las actividades o la capacidad más importante que tiene una computadora, ya que a través de esta propiedad el sistema puede recordar la información guardada en la sección o proceso anterior. La unidad típica de almacenamiento son los discos magnéticos o discos duros, los discos flexibles y los discos compactos.

Procesamiento de Información, es la capacidad del sistema de información para efectuar cálculos de acuerdo con una secuencia de operaciones preestablecidas. Los cálculos pueden efectuarse con datos introducidos recientemente en el sistema o bien con datos que están almacenados.

Salida de información, la salida es la apreciada de un sistema de información para sacar la información procesada o bien de entrada al exterior. Las unidades típicas de salida son las impresoras, terminales, discos magnéticos, la voz y los plotters, entre otros [KENDALL & KENDALL, 2005].

#### **2.3.4 MÉTODO**

Método es una palabra que proviene del término griego *methodos* (“camino” o “vía”) y que se refiere al medio utilizado para llegar a un fin. Su significado original señala el camino que conduce a un lugar.

Método es un conjunto de tareas técnicas y herramientas que, de acuerdo a un enfoque metodológico, se aplican para la resolución de un problema [RENDON, 2000].

#### **2.3.5 DIAGRAMA**

Un diagrama es una representación gráfica en la que se muestran las relaciones entre las diferentes partes de un conjunto o sistema o los cambios de un determinado fenómeno,

Es una representación gráfica de una colección de elementos del modelo, construida a menudo como un gráfico conexo de arcos (relaciones) y de vértices (otros elementos modelo) [BOOCH, 2000].

### 2.3.6 MODELO

El modelo es una representación parcial de la realidad; se refiere a que no es posible explicar una totalidad, ni incluir todas las variables que esta pueda tener, por lo que se refiere más bien a la explicación de un fenómeno o proceso específico, visto siempre desde el punto de vista de su autor [AGUILERA, 2000].

Así un modelo es la representación de la realidad por medio de las abstracciones. Los modelos enfocan ciertas partes importantes de un sistema (por lo menos, aquella que le interesan a un tipo de modelo específico), restándole importancia a otras.

Todos los modelos incluyen:

- **Notación:** El lenguaje para expresar cada modelo.
- **Proceso:** Las actividades que encaminan a la construcción ordenada de los modelos del sistema.
- **Herramientas:** Los artefactos que eliminan la labor de construir el modelo y reafirmar las reglas sobre los propios modelos, de forma que sea posible revelar los errores e inconsistencias.

### 2.3.7 MODELO VISUAL

El objetivo del modelado de un sistema es capturar las partes esenciales del sistema, plasmando de manera gráfica lo más importante. Esto se conoce como modelado visual.

El modelado visual permite capturar las partes más importantes del sistema, el cual permite manejar la complejidad del sistema, modelar el sistema independientemente del lenguaje de implementación [VALENCIA, 2000].

La importancia del modelo visual de un software es aquella que ayuda a recopilar las partes principales o esenciales de un sistema, como pueden ser:

- Perspectivas del usuario.



- Análisis y diseño de una aplicación.
- Minimiza la complejidad del programa.
- Reusabilidad de los componentes del sistema.

### **2.3.8 METODOLOGÍA**

Conjunto de métodos que siguen una investigación científica o una exposición doctrinal, se puede decir un enfoque, un manera de interpretar la realidad o la disciplina en cuestión [VALENCIA, 2000]. La metodología de la investigación proporciona tanto al estudiante como a los profesionales una serie de herramientas teórico-prácticas para la solución de problemas mediante el método científico. Estos conocimientos representan una actividad de racionalización del entorno académico y profesional fomentando el desarrollo intelectual a través de la investigación sistemática de la realidad [SAMPIERI, 2003].

### **2.4 LENGUAJE UNIFICADO DE MODELADO – UML**

UML (*Unified Modeling Language*) es la sucesión de una serie de métodos de análisis y diseño orientadas a objetos que aparecen a fines de los 80's y principios de los 90s. UML es llamado un lenguaje de modelado, no un método. UML prescribe una notación estándar y semánticas esenciales para el modelado de un sistema.

El UML fusiona los conceptos de la orientación a objetos aportados por Booch, OMT y OOSE [Booch, 1999]. Permite visualizar, construir y documentar los artefactos o elementos de un sistema.

UML permite ver el sistema de distintos puntos de vista [GRAU, 2001].

#### **2.4.1 CARACTERÍSTICAS DE UML**

El UML permite al desarrollador del sistema de software ver un sistema de distintos puntos de vista, además aleja al diseñador del código fuente y por último logran un aumento de abstracción de enfoque del sistema sin perder la información para que se pueda comprender claramente las características de la aplicación.

UML permite desarrollar diagramas, que es una representación gráfica de una colección de elementos de modelado, que muestra los distintos aspectos del sistema.

Se mencionan los diagramas para el modelado de un sistema. [GRAU, 2001].

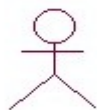
- Diagrama de Casos de Uso.
- Diagrama de Secuencia.
- Diagrama de Actividades.
- Diagrama de Estados.
- Diagrama de Colaboración.
- Diagrama de Componentes.
- Diagrama Conceptual.
- Diagramas de Clases.
- Diagramas de Despliegue.

#### **2.4.1.1 DIAGRAMA DE CASOS DE USO**

El diagrama de casos de uso representa la forma en como un Cliente (Actor) opera con el sistema en desarrollo, además de la forma, tipo y orden en como los elementos interactúan (operaciones o casos de uso), es decir describe el comportamiento del sistema.

Un diagrama de casos de uso consta de los siguientes elementos:

- **Actor**



Una definición previa, es que un Actor es un rol que un usuario juega con respecto al sistema. Es importante destacar el uso de la palabra rol, pues con esto se especifica que un Actor no necesariamente representa un apersona en particular, sino más bien la labor que realiza frente al sistema.

- **Casos de Uso**



Es una operación/tarea específica que se realiza tras una orden de algún agente externo, sea desde una petición de un actor o bien desde la invocación desde otro caso de uso. Representan la funcionalidad completa de un sistema mostrando su interacción con los agentes, la representación se hace a través de las relaciones de los actores y los casos de uso (acciones) dentro del sistema.

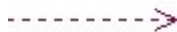
- **Relaciones**

- **Asociación**



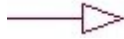
Es el tipo de relación más básica que indica la invocación desde un actor o caso de uso a otra operación (caso de uso). Dicha relación se denota con una flecha simple.

- **Dependencia o Instanciación**



Es una forma muy particular de relación entre clases, en la cual una clase depende de otra, es decir, se instancia (se crea). Dicha relación se denota con una flecha punteada.

- **Generalización**



Este tipo de relación es uno de los más utilizados, cumple una doble función dependiendo de su estereotipo, que puede ser de Uso (<< uses >>) o de Herencia (<< extends >>).

Este tipo de relación está orientado exclusivamente para casos de uso (y no para actores).

**extends:** se recomienda utilizar cuando un caso de uso es similar a otro (características).

**uses:** Se recomienda utilizar cuando se tiene un conjunto de características que son similares en más de un caso de uso y no se desea mantener copiada la descripción de la característica.

De lo anterior cabe mencionar que tiene el mismo paradigma en diseño y modelamiento de clases, en donde está la duda clásica de usar o heredar.

#### **2.4.1.2 DIAGRAMA DE SECUENCIA**

El diagrama de secuencia muestra la forma en que los objetos se comunican entre sí al transcurrir el tiempo. Consta de objetos que se representan de modo usual: rectángulo con nombre, mensajes entre los objetos representados por líneas continuas con una punta de flecha y el tiempo representado como una progresión vertical. Los objetos se colocan cerca de la parte superior del diagrama de izquierda a derecha y se acomodan de manera que simplifiquen el diagrama.

#### **2.4.1.3 DIAGRAMA DE COLABORACIÓN**

El diagrama de colaboración muestra una interacción organizada basándose en los objetos que toman parte en la interacción y los enlaces entre los mismos [PRESSMAN, 2003].

Presenta una alternativa al diagrama de secuencia para modelar interacciones entre objetos en el sistema, mientras que el diagrama de secuencia se centra en la secuencia cronológica del escenario que estamos modelando, el diagrama de colaboración se centra en estudiar todos los efectos de un objeto dado durante un escenario.

#### **2.4.1.4 DIAGRAMA DE ACTIVIDADES**

El diagrama de actividades describe el comportamiento del Sistema, representado como un conjunto de flujo secuencial de las actividades [PRESSMAN, 2003].

Un diagrama de actividades muestra el orden en el cual se va realizando las tareas dentro de un sistema, por lo tanto es el flujo de control de las actividades, las actividades se enlazan por transiciones automáticas, cuando una actividad termina se desencadena el lazo a la siguiente actividad, el cual muestra desde otro punto de vista.

#### **2.4.1.5 DIAGRAMA DE ESTADOS**

El diagrama de estado muestra la secuencia de estados por los que pasa un caso de uso o un objeto a lo largo de su vida, indicando que eventos hacen que se pase de un estado a otro y cuáles son las respuestas y acciones que se generan.

Se representa principalmente por: estado, elemento y transición.

#### **2.4.1.6 DIAGRAMA DE COMPONENTES**

El diagrama de componentes ilustra las piezas del software, controladores embebidos, que conforman un sistema. Un diagrama de componentes tiene un nivel más alto de abstracción que un diagrama de clase, usualmente un componente se implementa por una o más clases (u objetos) en tiempo de ejecución. Estos son bloques de construcción, como eventualmente un componente puede comprender una gran porción de un sistema.

#### **2.4.1.7 DIAGRAMA DE DESPLIEGUES**

El diagrama de despliegue es un diagrama estructurado que muestra la arquitectura del sistema desde el punto de vista del despliegue (distribución) de los artefactos del software en los destinos del despliegue.

Los artefactos representan elementos concretos en el mundo físico que son el resultado de un proceso de desarrollo, en cambio el destino de despliegue está generalmente representado por un nodo que es o bien de los dispositivos de hardware o bien algún entorno de ejecución de software.

#### **2.4.1.8 DIAGRAMA DE CLASES**

Un diagrama de clases es una representación gráfica de la vista estática, que muestra una colección de elementos declarativos (estáticos) del modelo, como clases, tipos y sus contenidos y sus relaciones.

### **2.5 METODOLOGÍA ICONIX**

ICONIX es un proceso de la ingeniería de software donde se especifica una serie de actividades que se debe realizar, con la finalidad de proporcionar un producto de software de calidad hacia los usuarios, es decir, que cumpla con los requerimientos de acuerdo a una planificación establecida.

Se menciona el enfoque de la metodología Iconix:

- Modelado de objetos conducido por casos de uso.
- Centrado en datos: se descompone en fronteras de datos.
- Basado en escenarios que descomponen los casos de uso.
- Enfoque iterativo e incremental.
- Ofrece trazabilidad.
- Uso directo de UML.

## **2.5.1 CARACTERÍSTICAS DE ICONIX**

- Flexible para diferentes estilos y clases de problemas
- Apoyo a la manera de trabajo de la gente
- Guía para los menos experimentados
- Expone los productos anteriores al código de manera estándar y comprensible.

## **2.5.2 FASES DE CICLO DE VIDA DE ICONIX**

ICONIX permite agilizar el desarrollo de software utilizando herramientas para cubrir todo el ciclo de vida del desarrollo de sistemas orientados a objetos, mediante una serie de etapas donde se especifican los requerimientos y se modela el comportamiento del sistema utilizando el lenguaje de modelamiento unificado.

- Análisis de Requerimientos.
- Análisis y Diseño Preliminar.
- Diseño.
- Implementación.

## **2.5.3 DESCRIPCIÓN DE LAS FASES**

### **2.5.3.1 ANÁLISIS DE REQUERIMIENTOS**

ICONIX cubre todas las fases del ciclo del desarrollo del software, empezando con un análisis de requisitos dividido en cuatro actividades:

- a) Abstracción de los objetos y las relaciones de agregación y generalización que existen entre ellos. Utilizar un diagrama de clases de alto nivel definido como modelo de dominio. Se refiere a hacer un relevamiento de todos los requisitos que deberían ser parte del sistema y con estos construir el diagrama de clases, que representa las agrupaciones funcionales que estructurarán el sistema en desarrollo.

- b) Se intenta en la medida de lo posible presentar una prototipo rápido de las interfaces del sistema, diagramas de navegación, entre otros para dar a los clientes una mayor comprensión del sistema propuesto.

Se intenta establecer las especificaciones iniciales que no se hayan contemplado y hacer que los usuarios puedan comenzar a evaluar la aplicación para percatarse de los que gusta o no a los mismos y poder hacer los cambios necesarios. El proceso se repite hasta que usuarios y analistas están de acuerdo en que el sistema cumple todas las necesidades y características solicitadas.

Los prototipos implican la creación de un modelo operativo de trabajo de un sistema o subsistema. Estos logran que los usuarios se hagan participantes activos en el desarrollo del sistema, que la definición de necesidades se simplifique, que la probabilidad de rechazo de implantación disminuya, la disminución del tiempo de desarrollo y en las fases de análisis y diseño volviéndolo un proceso ágil.

- c) Identificar casos de uso del sistema y sus actores involucrados utilizando el modelo de casos de uso para su representación.

Donde los casos de uso describen en forma de acciones el comportamiento del sistema desde el punto de vista del usuario, esto es como los usuarios interactúan con el sistema y que debe hacer para obtener el resultado esperado, definen los límites del sistema y relaciones entre el sistema y el entorno comprendiendo a los actores, el sistema y los propios casos de uso.

- d) Por último se hace una revisión de los requisitos funcionales con los casos de uso y con los objetos del dominio (trazabilidad).

### **2.5.3.2 ANÁLISIS Y DISEÑO PRELIMINAR**

Se contemplan 3 actividades:



- a) Se describen los casos de uso con un flujo principal de acciones y posibles flujos alternos y de excepción.
- b) Se realiza un diagrama de robustez, en donde se debe ilustrar las interacciones existentes entre los objetos participantes de un caso de uso.

El análisis de robustez ayuda a identificar los objetos que participan en cada caso de uso, sirve para saber si las especificaciones del sistema son razonables.

- c) Se debe actualizar el diagrama de clases definido en el modelo de dominio con las clases y atributos encontrados en los diagramas de robustez.

### **2.5.3.3 DISEÑO**

- a) Se especifica el comportamiento por medio de un diagrama de secuencia. Identificar mensajes entre los objetos de cada caso de uso y utilizar los diagramas de colaboración para mostrar la interacción entre objetos.

El diagrama de secuencia muestra interacciones entre objetos como una vista temporal y es el núcleo del modelo dinámico y muestra los cursos alternos que pueden tomar los casos de uso. A partir de la realización del diagrama de secuencia se empiezan a notar los métodos de las clases del sistema debido a que hasta este punto se ve la interacción entre los objetos con los actores y con los otros objetos dinámicamente.

- b) Se termina el modelo estático, añadiendo los detalles del diseño en el diagrama de clases.
- c) Verificar si el diseño satisface todos los requisitos identificados.

### **2.5.3.4 IMPLEMENTACIÓN**

- a) Se utiliza un diagrama de componentes de ser necesario, con el que se muestra la distribución física de los elementos que componen la estructura interna del sistema.

El diagrama de componentes describe los elementos físicos y sus relaciones en el entorno de realización y sus opciones.

c) Escribir el código.

d) Se realizan las pruebas.

En la fase de la implementación se realizan test de unidades, de casos, datos y resultados, de integración con los usuarios para verificar la aceptación de los resultados.

## **2.6 HERRAMIENTAS DE IMPLEMENTACIÓN**

Las herramientas de implementación proporcionan componentes u objetos que ayudan a la creación de una interfaz de usuario adecuado con las necesidades, se especifican herramientas para el desarrollo del sistema.

### **2.6.1 PHP**

PHP (*Hypertext Preprocessor*) es un lenguaje de programación de código abierto con el cuál se pueden crear páginas dinámicas, es decir, que son interactivas con el usuario, y permite mayor rapidez en el manejo de datos; una de las mayores ventajas de este lenguaje es su integración con la base de datos MySQL. Lo que permite, manejar una base de datos desde una simple página PHP, añadiendo, modificando, eliminando, y realizando todas las operaciones posibles con una base de datos.

Las principales características del lenguaje de programación PHP son: su rapidez, facilidad de aprendizaje, soporta muchas plataformas, diversos sistemas operativos, servidores http y de base de datos, y por último que este es un software libre, que no implica gasto alguno para su aplicación [CASTRO, 2007].

### **2.6.2 MySQL**

MySQL es un sistema de administración de bases de datos (*Database Management System, DBMS*) para bases de datos relacionales. Así, MySQL no es más que una aplicación que permite gestionar archivos llamados de bases de datos.

MySQL es la base de datos de código de fuente abierta más usada, proporciona un administración de la base de datos incomparable en velocidad, estabilidad y facilidad de

despliegue, el cuál otorga a nuestras aplicaciones la capacidad de guardar y acceder a la información en forma rápida y precisa, ofreciendo una gestión de usuarios y password para la seguridad de los datos.

MySQL Server fue desarrollado inicialmente para manejar grandes bases de datos mucho más rápidamente que las soluciones existentes, además que la conectividad, velocidad y seguridad hacen que sea altamente satisfactorio para acceder a la base de datos [Castro, 2007].

### **2.6.3 SERVIDOR APACHE**

APACHE es el servidor de páginas web. Un servidor de páginas web es un programa que permite acceder a páginas web alojadas en un ordenador; es el servidor web hecho por excelencia, su configurabilidad, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa.

El servidor Apache es principalmente usado para servir páginas web estáticas y dinámicas en la WWW, es una tecnología gratuita de código fuente abierta, además que trabaja con una gran cantidad de lenguajes (PHP, Java Script, etc.).

La historia de Apache se remonta a febrero de 1995, donde empieza el proyecto del grupo Apache, el cual está basado en el servidor Apache httpd de la aplicación original de NCSA. El desarrollo de esta aplicación original se estancó por algún tiempo tras la marcha de Rob McCool por lo que varios webmaster siguieron creando sus parches para sus servidores web hasta que se contactaron vía email para seguir en conjunto el mantenimiento del servidor web, fue ahí cuando formaron el grupo Apache.

Fueron Brian Behlendorf y Cliff Skolnick quienes a través de una lista de correo coordinaron el trabajo y lograron establecer un espacio compartido de libre acceso para los desarrolladores.

Fue así como fue creciendo el grupo Apache, hasta lo que es hoy. Aquella primera versión y sus sucesivas evoluciones y mejoras alcanzaron una gran implantación como software de

servidor inicialmente solo para sistemas operativos UNIX y fruto de esa evolución es la versión para Windows.

## 2.7 BASE DE DATOS

Una base de datos es una colección de información organizada de forma que un programa de ordenador pueda seleccionar rápidamente los fragmentos de datos que necesite. Una base de datos es un sistema de archivos electrónico. Base de datos es un conjunto de datos persistentes que es utilizado por los sistemas de aplicación de alguna empresa dada [ J.C. DATE, 2001].

Las bases de datos tradicionales se organizan por campos, registros y archivos. Un campo es una pieza única de información; un registro es un sistema completo de campos; y un archivo es una colección de registros.

Entre las principales características de los sistemas de base de datos podemos mencionar:

- Independencia lógica y física de los datos.
- Redundancia mínima.
- Acceso concurrente por parte de múltiples usuarios.
- Integridad de los datos.
- Consultas complejas optimizadas.
- Seguridad de acceso y auditoría.
- Respaldo y recuperación.
- Acceso a través de lenguajes de programación estándar.

### 2.7.1 SISTEMA DE ADMINISTRACIÓN DE BASE DE DATOS

Un sistema de ambiente de bases de datos (SABD) es sencillamente el software que permite que una institución centralice sus datos, los administre eficientemente y proporcione acceso a los datos almacenados mediante programas de aplicación.

El sistema de administración de base de datos actúa como un interface entre los programas de aplicación y los archivos de datos físicos. Este sistema consta de tres elementos:

- Lenguaje de definición de datos
- Lenguaje de manejo de datos
- Diccionario de datos

**El lenguaje de definición de datos**, pone a disposición instrumentos para definir con facilidad y precisión, especificando la estructura lógica, interna y externa de la base de datos.

**El lenguaje de manejo de datos**, se usa en conjunción de algún lenguaje de programación, este lenguaje permite a los usuarios finales o administrador satisfacer solicitudes y/o desarrollar aplicaciones mediante comandos, como el SQL (*Structured Query Language*).

**El diccionario de datos**, es un archivo automatizado o manual de almacenamiento de definiciones de los elementos de datos y características de los mismos, como su uso, presentación física, propiedades, autorización y seguridad.

### 2.8 CALIDAD DE SOFTWARE

Es la concordancia entre el estado funcional y los requerimientos del funcionamiento, explícitamente a los estándares de documentación de desarrollo e implícitamente, características que son expectativas de todos los profesionales de software [GROSSO, 2002].

Es necesario realizar la evaluación de calidad de software porque permite llegar a construir un sistema de alta calidad y que soporten las necesidades de los usuarios finales. Para poder llegar al software de calidad, las métricas son un buen medio para poder entender, monitorear, controlar y probar el desarrollo del software.

Las métricas que afectan a la calidad de software se dividen en dos grupos:

- a) Factores que pueden ser medidos directamente (Costo, unidad de tiempo, etc.)
- b) Factores que sólo pueden ser medidos indirectamente (Facilidad de uso, mantenimiento, complejidad, eficiencia, etc.)

### **2.8.1 OPERACIONES DE UN PRODUCTO**

- Corrección, hasta donde satisface un programa su especificación y logra los objetivos propuestos por el usuario.
- Fiabilidad, hasta dónde se puede esperar que un programa lleve a cabo su función con la exactitud requerida.
- Eficiencia, es la cantidad de recursos de hardware y software que necesita una aplicación para realizar operaciones con los tiempos de respuestas adecuadas.
- Facilidad de Uso, es el esfuerzo para aprender el manejo de una aplicación, trabajar con ella, introducir datos y conseguir resultados.

### **2.8.2 REVISIÓN DEL PRODUCTO**

- Facilidad de Mantenimiento, es el esfuerzo requerido para poder localizar y reparar errores.
- Flexibilidad, es el esfuerzo requerido para modificar una aplicación en funcionamiento.
- Facilidad de Prueba, es el esfuerzo requerido para probar una aplicación de forma que cumpla con lo especificado en los requisitos.

### 2.8.3. TRANSICIÓN DEL PRODUCTO

- Portabilidad, es el esfuerzo requerido para transferir la aplicación a otro hardware o a otro sistema operativo.
- Reusabilidad, hasta donde se puede volver a emplear un programa.
- Interoperabilidad, es el esfuerzo necesario para comunicar la aplicación con otras b aplicaciones o sistemas informáticos.

### 2.9 FACTORES DE CALIDAD ISO 9126

El estándar ISO 9126 ha sido desarrollado en un intento de identificar los atributos clave de calidad para el software. El estándar identifica los atributos clave de calidad:

- Funcionalidad, hasta donde un software satisface las necesidades, indicadas por los siguientes subatributos: idoneidad, corrección, interoperabilidad, conformidad y seguridad.
- Confiabilidad, cantidad de tiempo que el software está disponible para su uso. Está referido por los siguientes subatributos: madurez, tolerancia a fallos y factibilidad de recuperación.
- Usabilidad, esfuerzo necesario para aprender a operar un sistema. Viene reflejado por los siguientes subatributos: facilidad de comprensión, facilidad de aprendizaje y operatividad.
- Eficiencia, la cantidad de recursos informáticos y de códigos necesarios para que un programa realice su función. Está indicado por los siguientes subatributos: tiempo de uso y recursos utilizados.
- Facilidad de mantenimiento, la facilidad con que una modificación puede ser realizada. Está indicada por los siguientes subatributos: facilidad de análisis, facilidad de cambio, estabilidad y facilidad de prueba.

- Portabilidad, la facilidad con que el software puede ser llevado de un entorno a otro. Está referido por los siguientes subatributos: facilidad de instalación, facilidad de ajuste, facilidad de adaptación al cambio.

El objetivo no es necesariamente alcanzar una calidad perfecta, sino la necesaria, suficiente para cada contexto, uso a la hora de la entrega y uso por parte de los usuarios.

### **2.9.1 PRUEBAS DE SOFTWARE**

La prueba de software es el conjunto de actividades que se realizan sistemáticamente, que pueda planificarse por adelantado y ejecutar una vez construido el código para la revisión final de las especificaciones, del diseño y de la codificación del software [Vasquez, 2001].

Las pruebas tienen como objetivo:

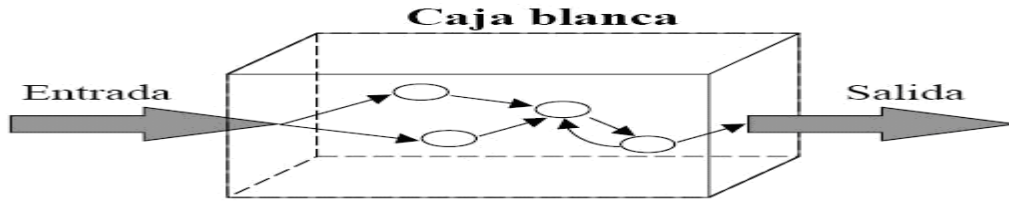
- Detectar errores, dar a la luz el máximo número de errores, para así reducir el riesgo e impacto que pueda tener.
- Calidad, se obtiene cuando el producto cumple con los requerimientos y las expectativas del cliente.
- Reducir el mantenimiento, cuando los errores se descubren muy temprano y se evita que ocurra en la realización del sistema, existe una reducción en el mantenimiento del sistema.

#### **2.9.1.1 PRUEBAS DE CAJA BLANCA**

La prueba de Caja Blanca o también denominada prueba de cristal, permiten examinar la estructura interna del programa, mediante métodos que garanticen que se ejecuten por lo menos una vez todos los caminos independientes de cada módulo, ejecutar las decisiones lógicas en sus caminos de verdadero y falso, ejecutar las estructuras de datos para asegurar su validez. [PRESSMAN, 2003].

En las pruebas de caja blanca se consideran puntos importantes, primero que las pruebas se deben realizar con el conocimiento del funcionamiento interno del código del sistema y segundo que las pruebas solo pueden ser realizados por los programadores.





**Figura 2.2 Enfoque Diseño de Caja Blanca**

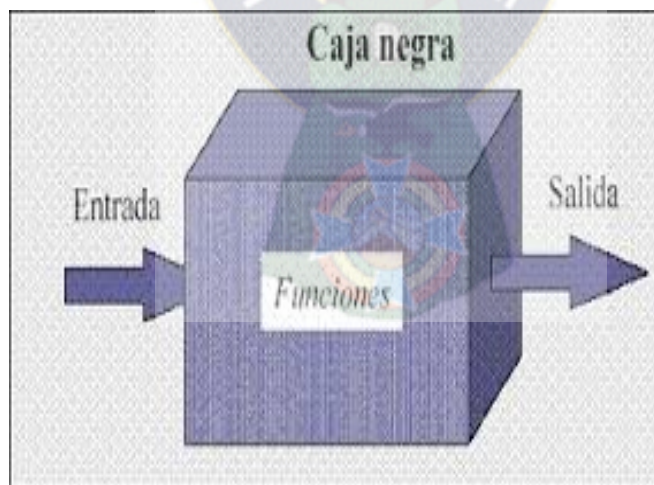
Fuente: Ingeniería de Software-pruebas del Software [PRESSMAN, 2003]

### 2.9.1.2 PRUEBAS DE CAJA NEGRA

La prueba de la Caja Negra pretende demostrar que las entradas se realizan y se aceptan de una forma adecuada, se produce una salida correcta del sistema, y además que la integridad de la información externa se mantiene.

Para realizar la prueba de caja negra se tomará en cuenta aspectos importantes; primero que no se presta atención al código del programa sino al aspecto fundamental del sistema y segundo que las pruebas son realizadas por el mismo usuario, como una prueba de aceptación.

Es importante resaltar que la prueba de Caja Negra es realizada por los usuarios finales [PRESSMAN, 2003].



**Figura 2.3 Enfoque Diseño de Caja Negra**

Fuente: Ingeniería de Software-pruebas del Software [PRESSMAN, 2003]

### 3.1 ANÁLISIS DE LA SITUACIÓN

Antes de comenzar con la captura de requerimientos se realiza una revisión de los procesos actuales para así conocer las debilidades del sistema actual y poder fortalecer con el Sistema de Información de Seguimiento y Control de la Dirección Distrital de Educación La Paz-2.

La Dirección Distrital de Educación La Paz-2 perteneciente a la Dirección Departamental de Educación La Paz, es la encargada de fortalecer el accionar de los responsables de las Unidades Educativas, y su personal, durante la gestión escolar en cuanto al desarrollo curricular y administrativo, a través de información confiable, eficaz y oportuna de las Unidades Educativas.

El personal de servicio de la Dirección Distrital de Educación La Paz-2 es el responsable del llenado y actualización de datos de las unidades educativas, cada persona emite los informes actualizados con el fin de proporcionar la información al responsable de la Dirección Distrital de Educación La Paz-2, que se encarga de la revisión de dichos informes.

Los datos de cada unidad educativa son distintos, el responsable del Sistema de Información tiene la tarea de revisar manualmente los informes, es decir, verificar mediante los registros diarios; aspecto que dificulta la realización de un reporte general para la Institución. Si bien, la información generada se encuentra en un libro de registros no prevé aspectos como la pérdida, extravío de alguna información. Por todos los aspectos mencionados anteriormente es conveniente resguardar la información mediante un sistema que satisfaga la seguridad de la información y que solo personal autorizado tenga acceso al mismo.

### 3.2 FASE DE ANÁLISIS DE REQUERIMIENTOS

En la fase de Análisis de Requerimientos se debe responder a una pregunta fundamental: ¿Qué es lo que quiere el Cliente? Y para ello, se debe diagnosticar la Situación Actual, recopilar los requerimientos del Cliente, tanto en relación al Sistema, como generales respecto al área de Informática, es decir la situación ideal, para así poder definir Alternativas

de Solución, según las cuales podemos avanzar desde lo que hoy se posee, hacia el punto que se pretende llegar. Los requerimientos se señalan a continuación:

- Proporcionar una interfaz amigable para el usuario, fácil y entendible para uso del personal.
- Proporcionar una información confiable de las distintas unidades educativas del distrito - 2, mediante una base de datos que contenga los datos más sobresalientes de la unidad educativa, tratamiento y otros.
- Disposición inmediata de la información para el personal autorizado.
- Elaboración de circulares e instructivos.
- Uso de normas de seguridad y calidad para el diseño de sistemas de información.
- Presentación acorde con el estándar de las aplicaciones del cliente en cuanto a tipo de letras, colores, logos, mensajes, usabilidad.

Debe seguirse un plan de elaboración que vaya en función del tiempo que se estima la conclusión del software, de tal manera que el software construido podrá ser modificable, fiable y eficiente dando una mayor probabilidad de obtener un sistema de mejor calidad al final del proceso. Se identifica y define los casos de uso del sistema, se organiza los casos de uso en grupos (paquetes) y se asignan requerimientos funcionales a casos de uso.

<b>N°</b>	<b>ACTIVIDADES</b>	<b>DURACIÓN</b>
1	Análisis de requerimientos	4 Semanas
2	Modelado del Negocio	
3	Modelado de Casos de Uso	
4	Diagrama de Casos de Uso	
5	Diagrama de Colaboración	5 Semanas
6	Diagrama de Secuencias	
7	Diagrama de Secuencias	
8	Diagrama de Actividades	
9	Diagrama de Clases	
10	Construcción de prototipos de interfaces de usuario	6 Semanas
11	Implementación de interfaz de usuario	
12	Implementación y puestas en marcha del sistema	
13	Planificación de pruebas	
14	Realización de pruebas	
15	Evaluación de pruebas	

**Tabla 3.1 Cronograma de actividades**

Fuente: Elaboración propia

### 3.2.1 FUNCIONES DEL SISTEMA

Se describe de manera general las funciones del Sistema de Información de Seguimiento y Control para la Dirección Distrital de Educación La Paz -2.

En la Tabla 3.2 se hace una descripción general sobre las funcionalidades del Sistema, donde las funciones están orientadas a la construcción del sistema

Ref #	Función	Categoría	Atributo	Detalles y restricciones	Categoría
R1.	Verificar el permiso al Operador para que este tenga acceso al Sistema	Evidente	Tiempo de respuesta	5 segundos como máximo	Obligatorio
			Metáfora de interfaz	Pantalla con colorido agradable a la vista	Opcional
R2.	El sistema debe capturar los datos introducidos por el Operador, cuando se registra o actualiza datos de alguna unidad educativa	Oculto	Tolerancia a fallas	La información debe guardarse aunque este produzca fallas	Obligatorio
			Tiempo de respuesta	El tiempo de respuesta debe ser 5 segundos como máximo	Obligatorio
R3.	El sistema debe generar circulares e instructivos	Evidente	Tiempo de respuesta	10 segundos como máximo	Obligatorio
			Metáfora de interfaz	Pantalla con colorido agradable a la vista	Obligatorio
R4.	El sistema debe generar la información general de cada unidad educativa	Evidente	Tiempo de respuesta	10 segundos como máximo	Obligatorio
			Metáfora de interfaz	Pantalla con colorido agradable al operador	Opcional
R5.	El sistema deberá ser capaz de cumplir con las búsquedas necesarias que así se requieran	Evidente	Tiempo de respuesta	20 segundos como máximo	Opcional
			Metáfora de interfaz	Pantallas con colorido agradable al Operador	Opcional

**Tabla 3.2 Funciones del Sistema**

Fuente: Elaboración propia

### 3.2.2 ANÁLISIS DE RIESGOS

Se describe los posibles riesgos potenciales y las estrategias para solucionarlos que pudieran surgir en un futuro en la elaboración del producto (Tabla 3.3).

<b>Tipo de Riesgo</b>	<b>Descripción</b>	<b>Problema que ocurra</b>	<b>Efecto</b>	<b>Estrategia</b>
<b>Riesgo de estimación</b>	Tiempo de entrega de software.	Posible retraso con la entrega, si es que las partes involucradas en el proyecto no dieran continuidad de trabajo	Retraso en la entrega del producto de software, restando credibilidad y compromiso en las partes que incumplan.	Acelerar la entrega con el cumplimiento de las partes para la culminación del proyecto.
<b>Riesgo de personal</b>	Cambio de autoridades en la Dirección Distrital.	Posible abandono del proyecto si las nuevas autoridades no estén de acuerdo.	Se perjudicará el proyectista y se cancelará el proyecto.	Realizar acuerdos serios en el cambio de autoridad entre los salientes y entrantes.
<b>Riesgo Tecnológico</b>	Que no se cuente con conexión a internet en un futuro.	Posible corte de internet o cortes momentáneos.	Una parte de producto de software no tendrá funcionalidad	Realizar la planificación de posible ejecución sin internet

**Tabla 3.3 Análisis de riesgos**

Fuente: Elaboración propia

### 3.3 DESARROLLO CASOS DE USO

Los casos de uso describen de forma narrativa los eventos que siguen los actores para completar un evento del sistema; a continuación se describe el caso de uso de más alto nivel ya que da una breve descripción del funcionamiento.

### 3.3.1 ACTORES DEL SISTEMA

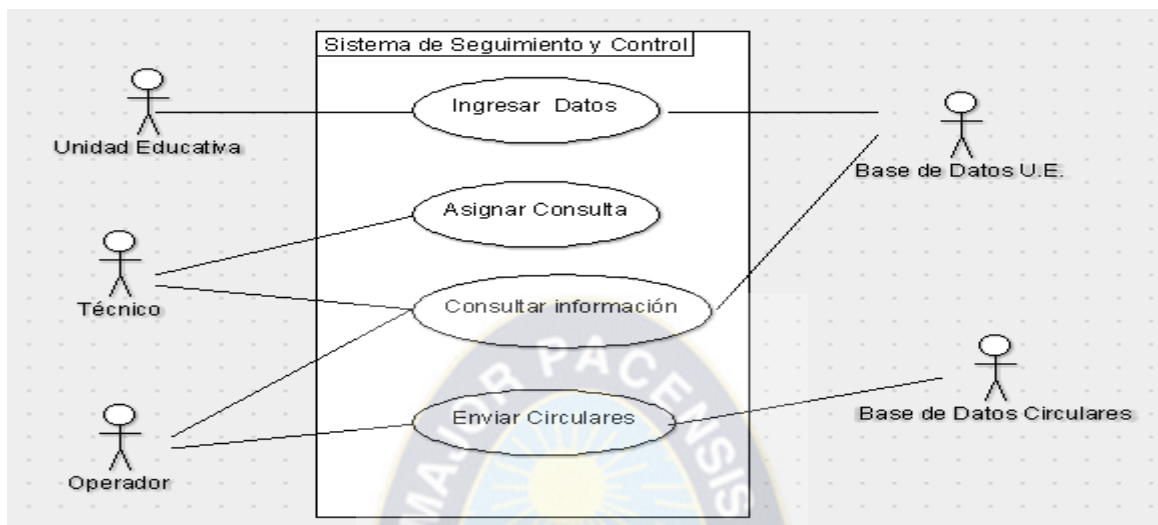
Se muestra la descripción de las personas que se encuentran involucradas en los distintos procesos de seguimiento y control de las actividades de las Unidades Educativas. (Tabla 3.4)

<b>ACTOR</b>	<b>DESCRIPCIÓN</b>
<b>Operador</b>	Persona encargada de registrar, modificar los datos de las Unidades Educativas y enviar instructivos y circulares.
<b>Técnico</b>	Encargado de realizar el control y seguimiento de las Unidades Educativas
<b>Director</b>	Responsable de justificar el cumplimiento correcto de la Unidad Educativa
<b>Unidad Educativa</b>	Es el actor central del Sistema sobre la que ejerce el seguimiento

**Tabla 3.4 Actores del Sistema**

Fuente: Elaboración propia

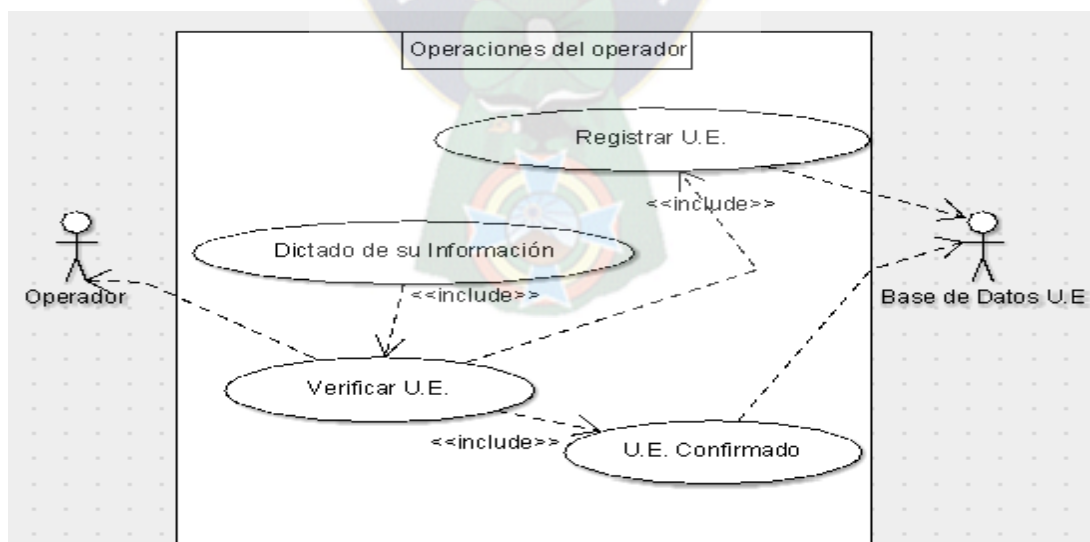
El siguiente diagrama de casos de uso es la representación del Sistema de Seguimiento y Control para la Dirección Distrital de Educación La Paz – 2 ( Figura 3.1).



**Figura 3.1 Delimitación del Sistema en base a los actores**

Fuente: Elaboración propia

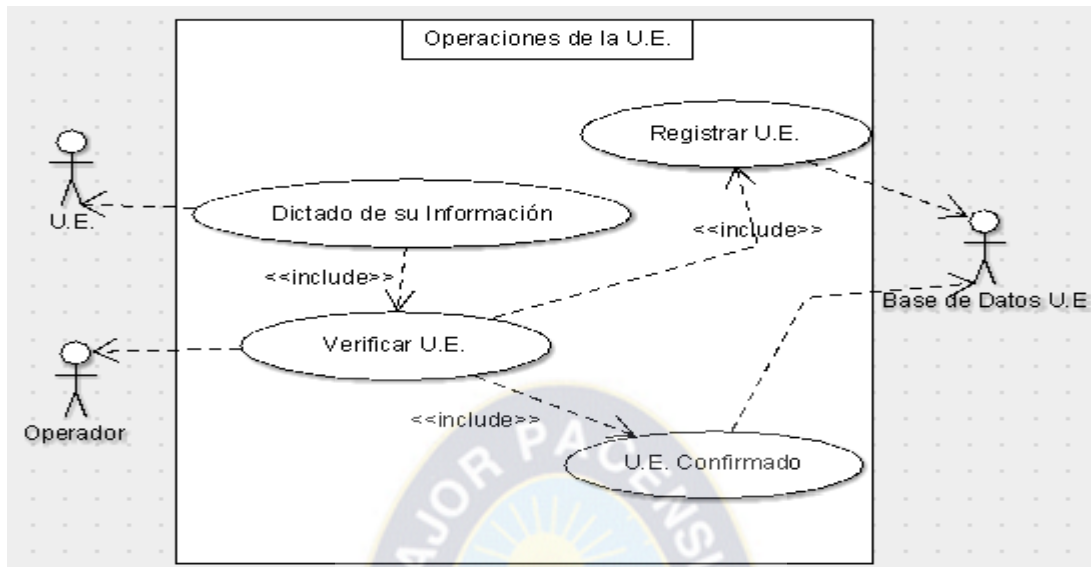
También se da una descripción más detallada de los casos de uso más importantes, que ayuda a comprender de mejor manera el funcionamiento del sistema ( Figura 3.2)



**Figura 3.2 Diagrama de Casos de Uso 1: Operaciones del Operador**

Fuente: Elaboración Propia

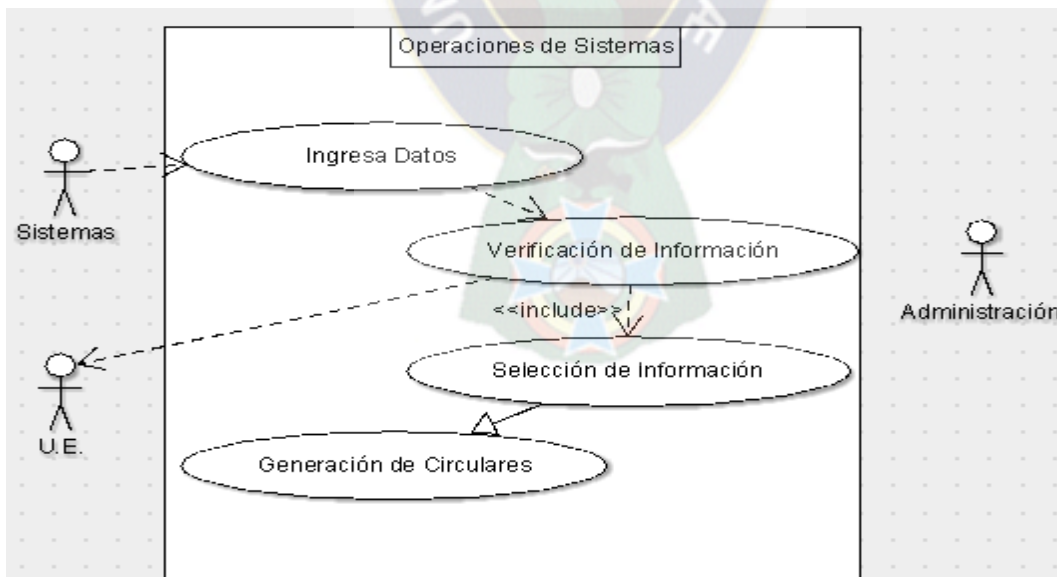




**Figura 3.3 Diagrama de Casos de Uso 2: Operaciones U.E**

Fuente: Elaboración Propia

Ahora se describe mediante casos de uso las operaciones que realiza el sistema (figura 3.4)



**Figura 3.4 Diagrama de Casos de Uso 3: Operaciones de Sistemas**

Fuente: Elaboración Propia

### 3.3.2 CASOS DE USO EXPANDIDOS

Los casos de uso expandidos del sistema permiten entender de mejor manera la relación que existe entre el sistema y su entorno, esos diagramas de referencia definirán lo que se espera que realice el sistema, sin definir su funcionamiento interno.

Antes de describir los casos de uso, debe tomarse en cuenta que el actor Operador representa a toda aquella persona, que pertenezca a la institución y que tenga tuición de administrar el sistema. Los casos de uso de alto nivel y los casos expandidos se describen a continuación.

a) **Casos de Uso 1:** Gestionar Unidad Educativa

Ofrece la funcionalidad para crear, editar y eliminar el registro de una Unidad Educativa, además permite vincularlo con su respectivo Director y Técnico asignado.

b) **Casos de Uso 2:** Gestionar Director de la Unidad Educativa

Ofrece la funcionalidad para crear, editar y eliminar el registro de un Director encargado de la Unidad Educativa.

c) **Casos de Uso 3:** Asignar Consulta

Permite la asignación de consulta a alguna Unidad Educativa.

d) **Casos de Uso 4:** Ingresar Nuevo Operador

Permite que otras personas, que cuenten con la autorización respectiva puedan acceder a la manipulación del sistema.

e) **Casos de Uso 5:** Validar Operador:

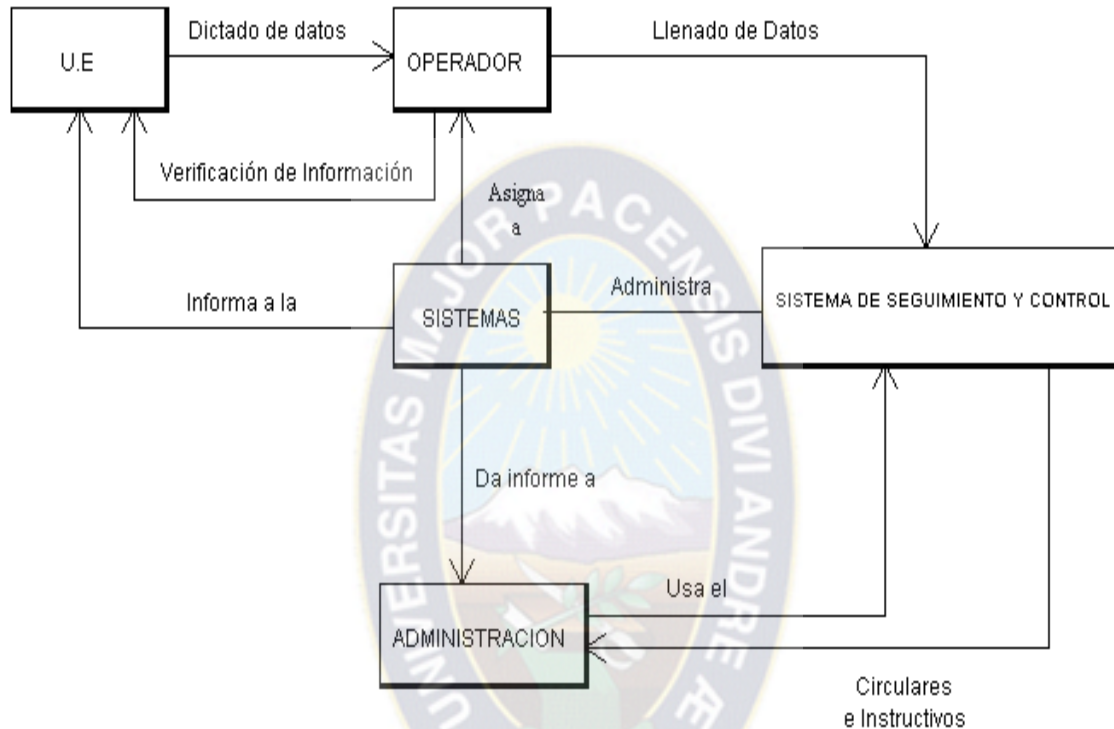
Validar a un Operador ya registrado para el uso del sistema, mediante un login y password para que pueda hacer uso del sistema.

f) **Casos de Uso 6:** Enviar reportes:

Tiene como propósito permitir el registro y envío de reportes.

### 3.3.3 DEFINIENDO EL MODELO CONCEPTUAL PRELIMINAR

Para definir el diagrama de clases (Figura 3.5), es necesario definir el modelo conceptual el cual muestra los conceptos presentes en el dominio del problema.



**Figura 3.5 Modelo Conceptual Preliminar**

Fuente: Elaboración Propia

### 3.3.4 DEFINIENDO LOS CASOS DE USO EXPANDIDOS

Los casos de uso expandidos detallan la conformidad de un proceso paso a paso, suelen ser útiles para alcanzar un conocimiento más profundo de los procesos y de los requerimientos. Los casos de uso esenciales, que integran el sistema, son los siguientes:

Casos de uso expandido de ingreso de nuevo operador del sistema (Ver tabla 3.5)

Casos de uso:	Ingresar Nuevo Operador	
Actores:	Operador, Técnico	
Propósito:	Permite que otras personas, que cuenten con la autorización respectiva puedan acceder a la manipulación del sistema	
Resumen:	Se inicia cuando el Departamento de Sistemas necesita adicionar a un nuevo usuario del sistema este puede ser un Técnico u Operador, los cuales se encargan del mantenimiento y actualización de datos	
Curso normal de eventos		
Acción del actor	Respuesta del Sistema	
1. El caso de uso comienza cuando un Operador ingresa su Login y password al sistema	2. El sistema verifica el Login y el Password del Operador, si está registrado, le permite el ingreso al sistema.  3. El sistema se desplaza en la interfaz Operador el menú de actualización de datos	

<p>4. El operador debe ingresar a la opción de nuevo operador.</p> <p>6. El operador ingresa los datos del nuevo Operador y puede modificar sus permisos y privilegios</p>	<p>5 El sistema verifica si el password es correcto e identifica al operador en esta instancia.</p> <p>7. Una vez realizada esta operación, el sistema supervisa si ya se encuentra, en tal caso enviará un mensaje de no admitido dentro el sistema al operador</p>
<p>Alternativas</p> <p>En línea 2: El login y el password son incorrectos, el sistema rechaza al operador y este no puede utilizarlo.</p> <p>En línea 6: El password es incorrecto, el sistema envía un mensaje de no admitido al</p>	

operador.

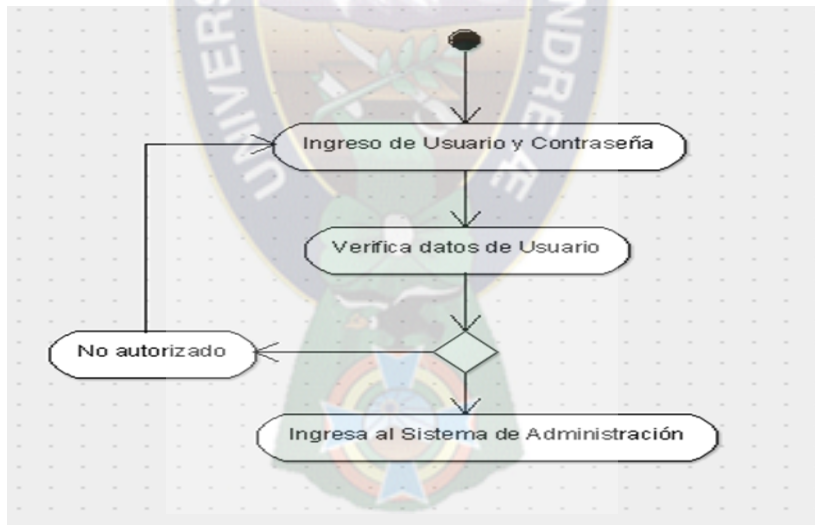
En línea 7: Verificar los datos que se desean guardar.

**Tabla 5Tabla 3.5 Caso de Uso Esencial: Ingresar Nuevo operador**

Fuente: Elaboración Propia

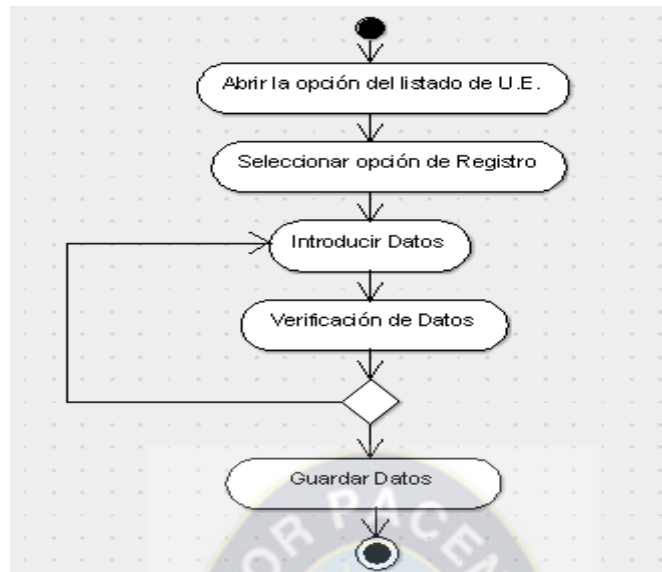
**3.3.5 DEFINIENDO DIAGRAMA DE ACTIVIDADES**

Se muestran diagramas de actividades para conceptualizar los procesos vistos anteriormente y poder describirlos de una manera explícita para el desarrollo del sistema.



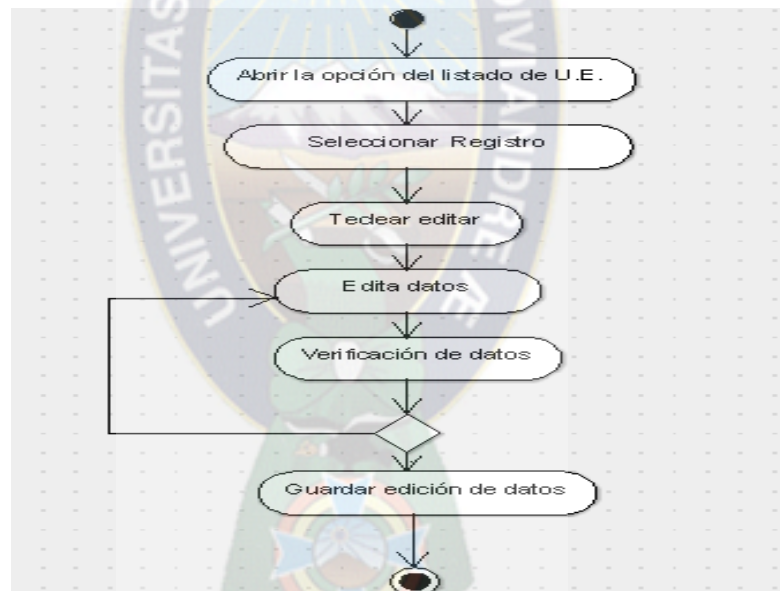
**Figura 3.6 Diagrama de Actividad: Ingreso al Sistema**

Fuente: Elaboración Propia



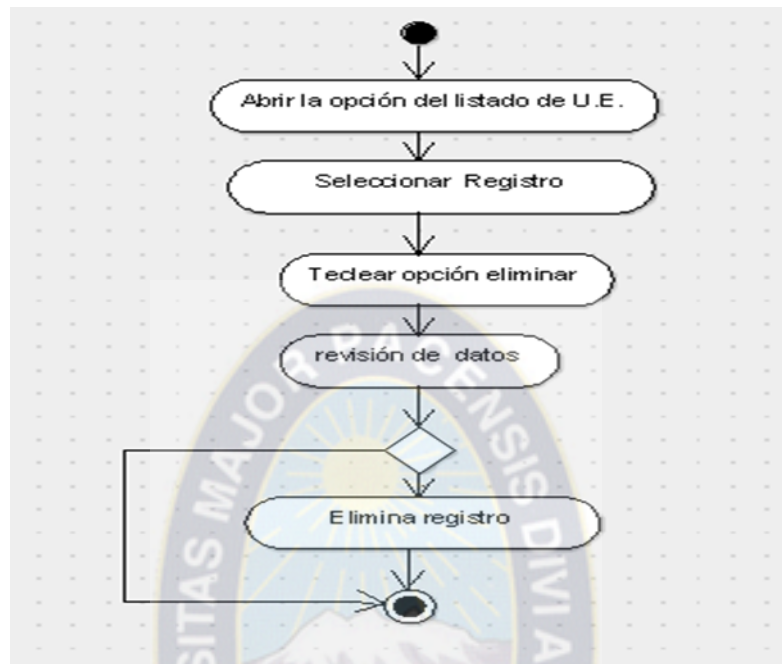
**Figura 3.7 Diagrama de Actividad: Creación de una U.E.**

Fuente: Elaboración Propia



**Figura 3.8 Diagrama de Actividad: Edición de una U.E.**

Fuente: Elaboración Propia



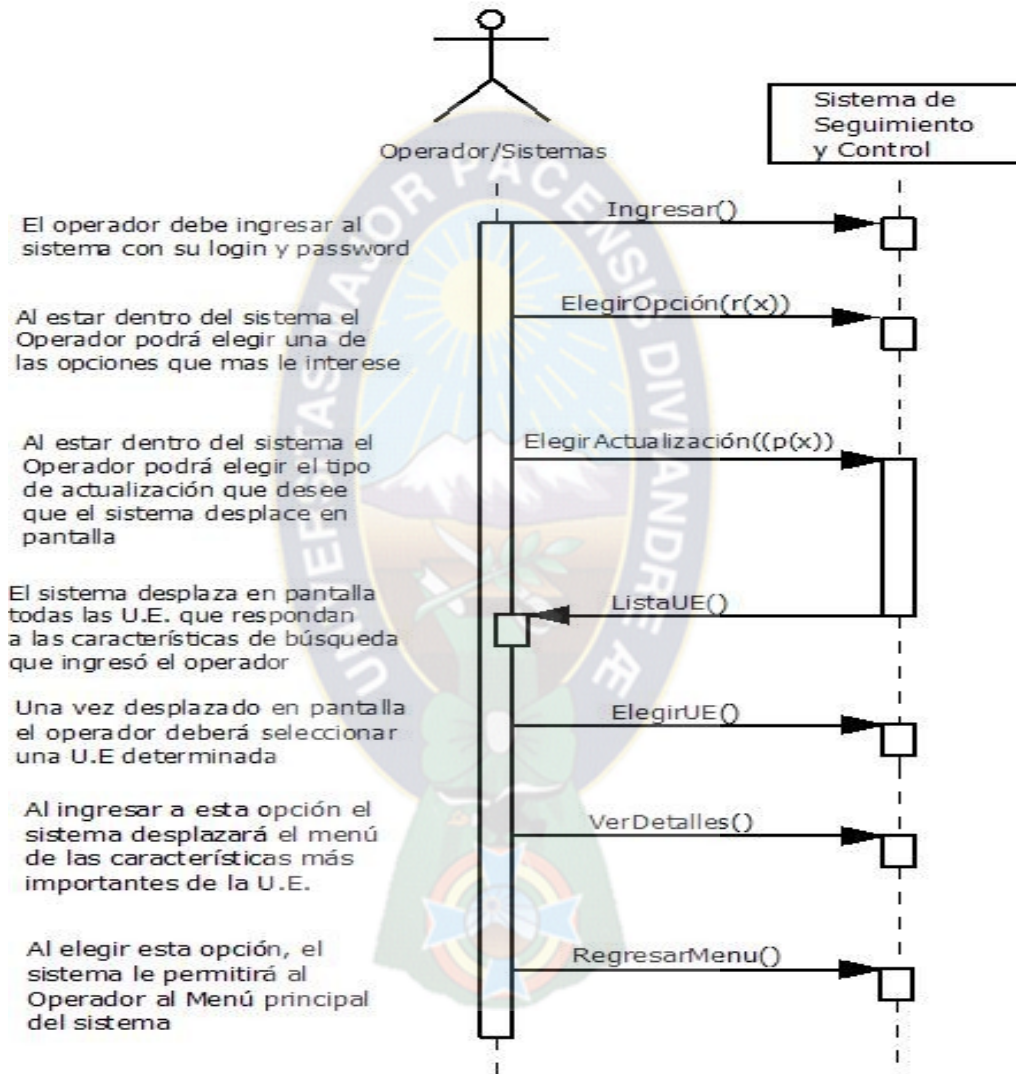
**Figura 3.9 Diagrama de Actividad: Eliminar una U.E**

Fuente: Elaboración Propia



### 3.3.6 DEFINIENDO DIAGRAMA DE SECUENCIA

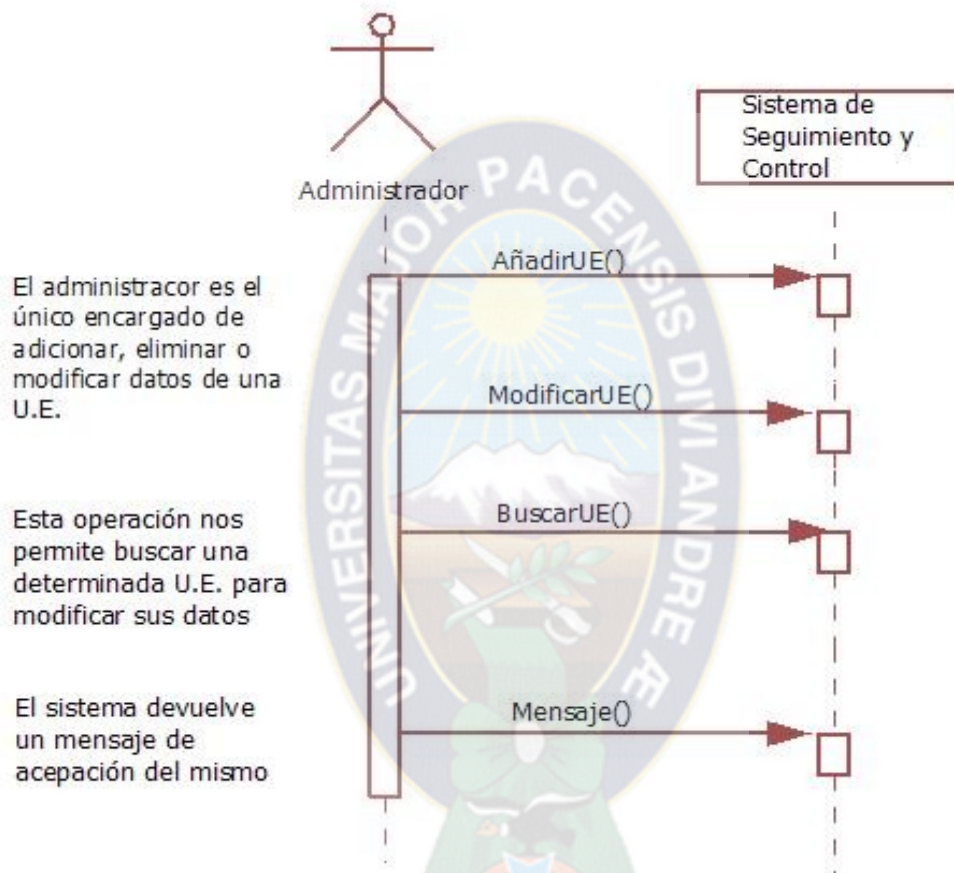
El diagrama de secuencia (figura 3.10), muestra la secuencia de mensajes entre objetos durante un escenario concreto según la secuencia temporal de los eventos, también muestra las interacciones entre objetos organizados en una secuencia temporal.



**Figura 3.10 Diagrama de Secuencia: Gestionar Unidad Educativa**

Fuente: Elaboración Propia

**Diagrama de Secuencia del Sistema de Seguimiento y Control de las Unidades Educativas (Registrar Unidad Educativa)**



**Figura 3.11 Diagrama de Secuencia: Registrar Unidad Educativa**

Fuente: Elaboración Propia

### 3.3.7 DEFINIENDO CONTRATO DE OPERACIONES

Los contratos de operaciones (ver Tabla 3.6), definen o expresan el comportamiento del sistema, las operaciones definidas en los diagramas de secuencia proponen describir el efecto que tendrán las operaciones dentro del sistema de seguimiento y control de las Unidades Educativas de la Dirección Distrital de Educación La Paz – 2. Se muestran a continuación los contratos más relevantes del sistema.

Se define el contrato: Adicionar Unidad Educativa ( Tabla 3.7)

<b>Contrato para Añadir Unidad Educativa</b>	
<b>Nombre:</b>	Añadir Unidad Educativa (Datos de la Unidad Educativa)
<b>Responsabilidades:</b>	Debe capturar los datos de la Unidad Educativa, introducidos por el Operador y guardado en la Base de Datos de la U.E.
<b>Tipo:</b>	Sistema
<b>Referencia:</b>	Caso de Uso : Registrar
<b>Excepciones.</b>	El Operador debe primero haber ingresado al sistema para registrar a una U.E.
<b>Precondiciones:</b>	Se requiere haber elegido la opción Añadir, además el servidor de la BD debe estar activo.
<b>Precondición</b>	El Operador debe ser verificado por el sistema.
<b>Poscondiciones</b>	Fue registrado una U.E. en la base de datos.

**Tabla 3.6 Contrato: Adición de una Nueva Unidad Educativa**

Fuente: Elaboración Propia

Se define el contrato: Modificar Unidad Educativa (ver Tabla 3.7)

<b>Contrato para Modificar Unidad Educativa</b>	
Nombre:	Modificar Unidad Educativa ()
Responsabilidades:	Debe modificar los datos de una Unidad Educativa identificado por el sistema
Tipo:	Sistema
Referencia:	Casos de Uso : Registrar Unidad Educativa
Notas:	El registro de cualquier cambio debe ser flexible y permitir que algunos campos puedan quedar vacíos.
Excepciones.	Antes de modificar los datos de una Unidad Educativa se debe buscar la información que este dentro del sistema, indicar si los datos son correctos.
Salida:	Datos de la Unidad Educativa modificado
Precondiciones:	Se requiere encontrar los datos de la Unidad Educativa del cual se desea modificar los datos
Precondición	El Operador debe ser verificado por el sistema
Poscondiciones	Los datos de la Unidad Educativa fueron modificados.

**Tabla 3.7 Contrato: Modificar Unidad Educativa**

Fuente: Elaboración Propia

En la Tabla 3.8 se define el contrato de buscar Unidad Educativa.

<b>Contrato para Buscar Unidad Educativa</b>	
Nombre:	Buscar Unidad Educativa ()
Responsabilidades:	Debe buscar los datos de la Unidad Educativa que ha sido introducido por el Operador
Tipo:	Sistema
Referencia:	Casos de Uso : Registrar Unidad Educativa Nueva
Cruzadas:	
Notas:	
Excepciones.	Antes de buscar a una Unidad Educativa, el Operador debe introducir el nombre de la Unidad Educativa a buscar
Salida:	Desplaza los datos de la Unidad Educativa encontrada
Precondiciones:	Se requiere introducir el nombre de la Unidad Educativa a buscar
Poscondiciones	Se encontró los datos de la Unidad Educativa buscada.

**Tabla 3.8 Contrato: Buscar Unidad Educativa**

Fuente: Elaboración Propia

En tanto que en la tabla 3.9 se define el contrato de mensaje

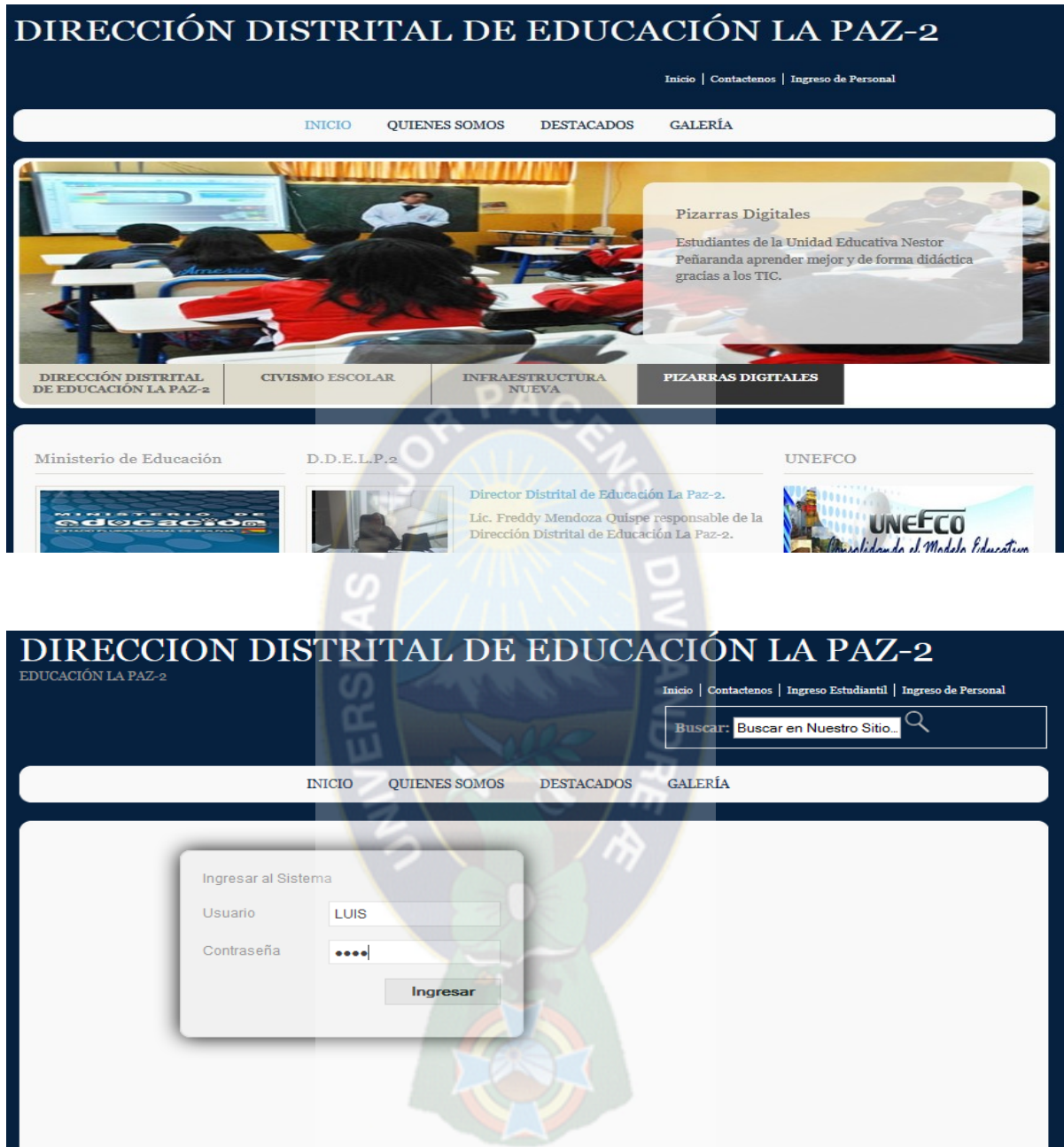
<b>Contrato para Mensaje</b>	
Nombre:	Mensaje ()
Responsabilidades:	<p>Debe devolver un mensaje de respuesta cada vez que realiza las siguientes operaciones:</p> <ul style="list-style-type: none"> <li>✓ Adicionar: (Usted ha registrado correctamente)</li> <li>✓ Eliminar: (está seguro que desea eliminar a este operador)</li> <li>✓ Modificar: se modificaron los datos correctamente</li> </ul>
Tipo:	Sistema
Referencia:	Casos de Uso : Registrar Unidad Educativa Nueva
Cruzadas:	
Notas:	
Excepciones.	Antes de enviar cualquiera de estos mensajes se debe elegir las opciones descritas anteriormente
Salida:	Desplaza los datos de la Unidad Educativa encontrada
Precondiciones:	Se requiere haber elegido una opción para que esta operación funcione.
Poscondiciones	Se encontró los datos de la Unidad Educativa buscada.

**Tabla 3.9 Contrato: Mensaje**

Fuente: Elaboración Propia

### 3.8.8 DEFINIENDO INTERFAZ DE USUARIO

A continuación se muestran las pantallas principales del sistema. En el desarrollo del sistema se tomó muy en cuenta la funcionalidad de cada pantalla, para que pueda ser amigable ante el usuario.



**Figura 3.12 Pantalla Principal del Sistema**

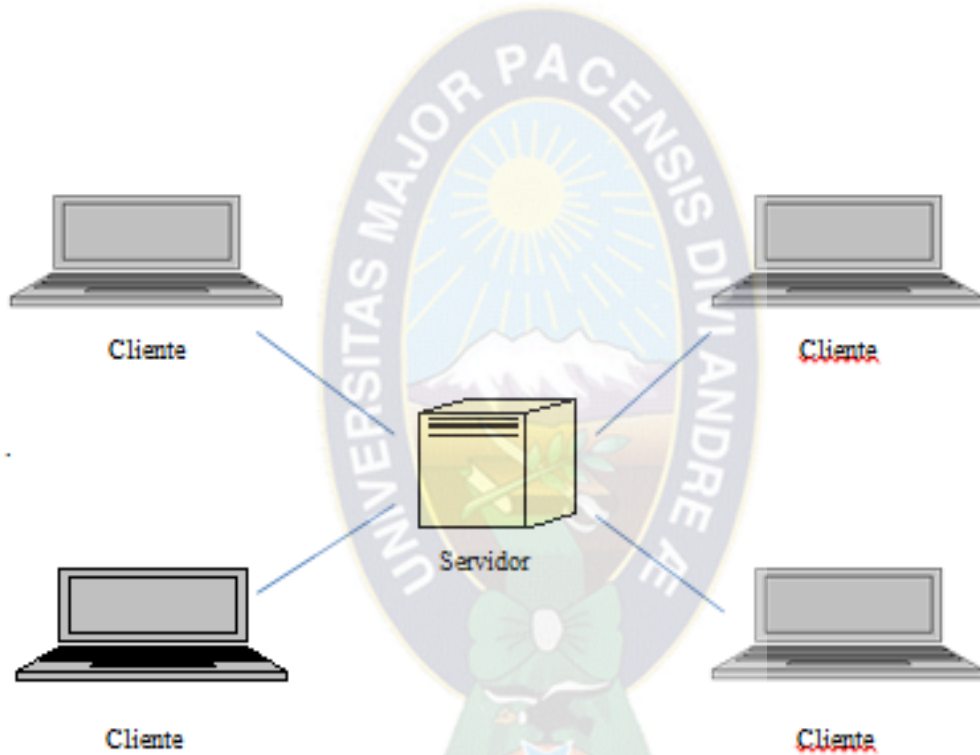
Fuente: Elaboración Propia

### 3.4 FASES DE IMPLEMENTACIÓN

En la fase de implementación, se puede ver si los análisis realizados son apropiados para su implementación.

### 3.4.1. PERFECCIONAMIENTO DE LA ARQUITECTURA DEL SISTEMA

La arquitectura del sistema (ver figura 3.12), es del tipo Cliente Servidor, referido este a aplicaciones que utilizan una red de computadoras, donde los procesos se dividen en componentes que pueden ejecutarse en diferentes máquinas. El término servidor se aplica a cualquier programa que ofrece un servicio aceptando peticiones recibidas a través de la red, procesándolas y regresando resultados a quien generó la petición. Un programa se convierte en cliente cuando envía una petición a un servidor y espera una respuesta.



**Figura 3.13** Arquitectura del Sistema de Información

Fuente: Elaboración Propia

### 3.4.2. DISEÑO DE LA BASE DE DATOS

El diseño y construcción de la Base de Datos Relacional, confronta el problema de que los objetos no pueden ser diseñados directamente en un conjunto de entidades relacionales, ya que los objetos tienen identidad, estados y comportamiento. La solución a este problema es usar una técnica, que manipule el desarrollo robusto del diseño orientado a objetos, que



construirá un conjunto de entidades con las semejanzas y diferencias que se hallen en los diagramas de clase y objetos, estas entidades usan a los objetos creando referencias directas, con el uso de llaves extranjeras y primarias, (ver figura 3.13).

Se debe considerar lo siguiente:

- La meta del modelo relacional de datos, es la normalización de los datos, es decir, eliminar la redundancia que hay entre los datos existentes en las tablas, y así hacer una rápida recuperación de los datos que existen dentro de la Base de Datos.

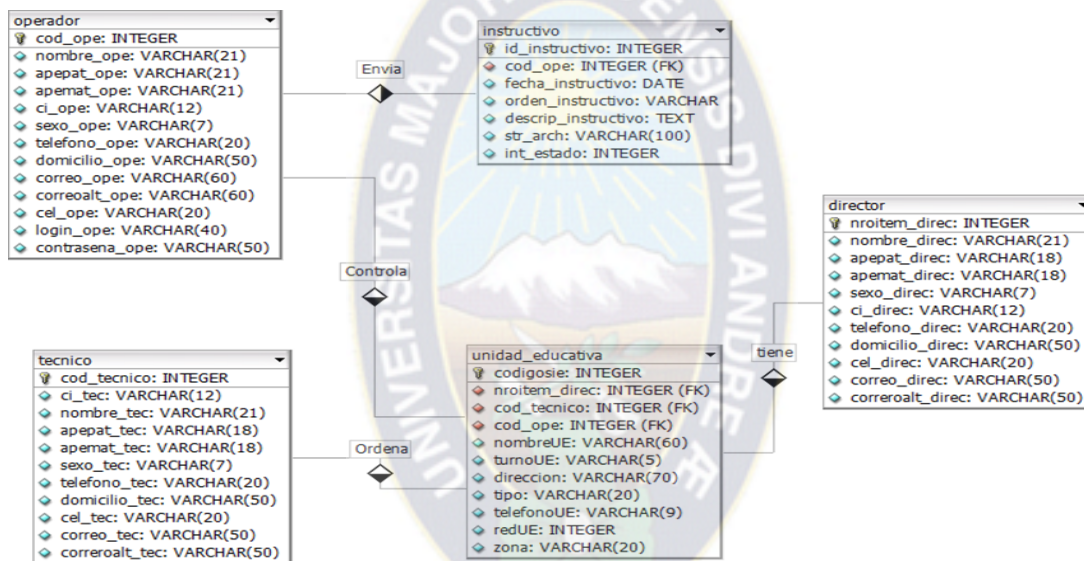


Figura 3.14 Diagrama de Entidad/Relación

Fuente: Elaboración Propia

### 3.4.3. DEFINIENDO EL ESQUEMA DE LA BASE DE DATOS

Definir el esquema de la base de datos, resulta independiente del análisis y diseño orientado a objetos, ya que pertenece a la capa de almacenamiento descrita en la arquitectura de tres capas que usa el Sistema, por tanto la implementación de la Base de Datos es resultado del modelo relacional descrito anteriormente (ver tabla 3.10). A continuación se lista la descripción de las tablas relacionales.

**Base de datos: BDdistrital**

**Tabla: OPERADOR**

<b>Campo</b>	<b>Tipo Dato</b>	<b>Descripción</b>
<u>cod_pe</u>	Int(11)	Se refiere al código de Operador
nombre_ope	Varchar(50)	Nombre del operador
apepat_ope	Varchar(30)	Apellido paterno del operador
apemat_ope	Varchar(30)	Apellido materno del operador
sexo_ope	Varchar(5)	Sexo del operador
ci_ope	Varchar(10)	Camet de Identidad del operador
teléfono_ope	Varchar(15)	Teléfono del Operador.
domicilio_ope	Varchar(15)	Dirección del operador
correo_ope	Varchar(70)	Correo electrónico del Operador
cel_ope	Varchar(10)	Teléfono móvil de referencia
correoalt_ope	Varchar(40)	Correo electrónico alterno
login_ope	Varchar(15)	Nombre de Usuario
contrasena_ope	Varchar(15)	Contraseña de Usuario

**Tabla 3.10 Operador**

Fuente: Elaboración Propia

**Base de datos: BDdistrital**

**Tabla: UNIDADEUCATIVA**

<b>N°</b>	<b>Campo</b>	<b>Tipo Dato</b>	<b>Descripción</b>
1	<u>CodigoSie</u>	Int(11)	Se refiere al código de la U.E.
2	nombre_UE	Varchar(50)	Nombre de la U.E.
3	turno_UE	Varchar(5)	Se refiere al turno de mañana tarde o noche.
4	dirección_UE	Varchar(70)	La ubicación de la U.E.
5	tipo_UE	Varchar(15)	Se refiere si es privado, fiscal o alternativo.
6	teléfono_UE	Varchar(10)	Teléfono de ubicación de la U.E.
7	red_UE	Int(5)	La ubicación de la U.E. en el macrodistrito
8	zona_UE	Varchar(20)	Ubicación según las zonas
9	cod_tecnico	Int(11)	Código del Técnico asignado a la U.E.
10	nroitem_direc	Int(11)	Ítem del director asignado

**Tabla 3.11 UNIDAD EDUCATIVA**

Fuente: Elaboración Propia

**Base de datos: BDdistrital**

**Tabla: INSTRUCTIVO\_CIRCULAR**

<b>Nº</b>	<b>Campo</b>	<b>Tipo Dato</b>	<b>Descripción</b>
1	<u>id_instructivo</u>	Int(7)	Código del instructivo.
2	descrip_instructivo	Text	Motivo del instructivo
3	Hora	Time	Hora de publicación del Instructivo.
4	int_estado	Int(11)	Es la activación o desactivación del instructivo
5	str_arch	Varchar(200)	Es el Instructivo ó Circular
6	orden_instructivo	Varchar(200)	Es el orden de publicación
7	fecha_instructivo	Date	Fecha de publicación del Instructivo

**Tabla 12Tabla 3.12 CIRCULARES E INSTRUCTIVOS**

Fuente: Elaboración Propia

**Base de datos: BDdistrital**

**Tabla: TECNICO**

<b>N°</b>	<b>Campo</b>	<b>Tipo Dato</b>	<b>Descripción</b>
1	<u>cod_tecnico</u>	Int(11)	Se refiere al codigo del técnico
2	nombre_tec	Varchar(50)	Nombre del tecnico
3	acepat_tec	Varchar(30)	Apellido paterno del técnico
4	apemat_tec	Varchar(30)	Apellido materno del técnico
5	sexo_tec	Varchar(5)	Sexo del técnico
6	teléfono_tec	Varchar(10)	Número telefónico del técnico
7	ci_tec	Varchar(10)	Carnet de Identidad del técnico
8	domicilio_tec	Varchar(100)	Ubicación de domicilio del técnico
9	cel_tec	Varchar(10)	Teléfono móvil de ubicación
10	CorreoTec	Varchar(100)	Correo electrónico del técnico
11	correault_tec	Varchar(100)	Correo alterno del técnico

**Tabla 3.13 TÉCNICO**

Fuente: Elaboración Propia

**Base de datos: BDdistrital**

**Tabla: Director**

<b>N°</b>	<b>Campo</b>	<b>Tipo Dato</b>	<b>Descripción</b>
1	<u>nroitem_direc</u>	Int(11)	Se refiere al item del director
2	nombre_direc	Varchar(50)	Nombre del director
3	apepat_direc	Varchar(30)	Apellido paterno del Director
4	apemat_direc	Varchar(30)	Apellido materno del Director
5	sexo_direc	Varchar(5)	Sexo del Director
6	ci_direc	Varchar(10)	Carnet de Identidad del Director
7	domicilio_direc	Varchar(70)	Dirección del Director
8	cel_direc	Varchar(10)	Teléfono móvil de ubicación
9	correo_direc	Varchar(100)	Correo Electrónico del Director
10	correoalt_direc	Varchar(100)	Correo alterno del Director
11	teléfono_direc	Varchar(20)	Número telefónico del Director

**Tabla 3.14 DIRECTOR**

Fuente: Elaboración Propia

### 3.4.4. IMPLEMENTACIÓN DE LOS MÉTODOS

La implementación de métodos, nace de la concepción de los diagramas de colaboración, la transformación es genera en el diagrama de clases, los cuales señalan “en teoría” la arquitectura global y las condiciones más importantes que detallan en gran parte la construcción de un software.

Se detalla el método Listado de Unidades Educativas (ver Tabla 3.15)

```
<?php
require_once('Common.php');
require_once('php/lang/LangVars-en.php');
require_once('php/AjaxTableEditor.php');
class Example1 extends Common
{
    var $Editor;

    function displayHtml()
    {
        ?>
        <br />
        <div align="left" style="position: relative;"><div
id="ajaxLoader1"></div></div>
        <br />
        <div id="historyButtonsLayer" align="left">
        </div>
        <div id="historyContainer">
            <div id="information">
            </div>
            <div id="titleLayer" style="padding: 2px; font-weight: bold;
font-size: 18px; text-align: center;">
            </div>
            <div id="tableLayer" align="center">
            </div>
            <div id="recordLayer" align="center">
            </div>
            <div id="searchButtonsLayer" align="center">
```

```

        </div>
    </div>

    <script type="text/javascript">
        trackHistory = false;
        var ajaxUrl = '<?php echo $this->getAjaxUrl(); ?>';
        toAjaxTableEditor('update_html',"");
    </script>
    <?php
    }

    function initiateEditor()
    {
        $tableColumns['codigosie'] = array('display_text' => 'Codigo', 'perms' =>
'TVQSXO');
        $tableColumns['nombre_UE'] = array('display_text' => 'Unidades
Educativas', 'perms' => 'EVTAXQSHO');
        $tableColumns['direccion_UE'] = array('display_text' => 'Direccion U.E.',
'perms' => 'EVTAXQSHO');
        $tableColumns['turno_UE'] = array('display_text' => 'Turno', 'perms' =>
'EVTAXQSHO');
        $tableColumns['tipo_UE'] = array('display_text' => 'Tipo U.E.', 'perms' =>
'EVTAXQSHO');
        $tableColumns['telefono_UE'] = array('display_text' => 'Telefono U.E.',
'perms' => 'EVTAXQSHO');
        $tableColumns['red_UE'] = array('display_text' => 'Red U.E.', 'perms' =>
'EVTAXQSHO');
        $tableColumns['zona_UE'] = array('display_text' => 'Zona U.E.', 'perms' =>
'EVTAXQSHO');
        $tableColumns['cod_tecnico'] = array('display_text' => 'Tecnico Asignado', 'perms'
=> 'EVTAXQ', 'join' => array('table' => 'tecnico', 'column' => 'cod_tecnico', 'display_mask'
=> "concat(tecnico.apepat_tec,' ',tecnico.nombre_tec)", 'type' => 'left'));
        $tableColumns['nroitem_direc'] = array('display_text' => 'Director Establecimiento',
'perms' => 'EVTAXQ', 'join' => array('table' => 'director', 'column' => 'nroitem_direc',
'display_mask' => "concat(director.apepat_direc,' ',director.nombre_direc)", 'type' =>
'left'));

        $tableName = 'unidad_educativa';
        $primaryCol = 'codigosie';
        $errorFun = array(&$this,'logError');
        $permissions = 'EAVIDQSXHO';

        $this->Editor = new
AjaxTableEditor($tableName,$primaryCol,$errorFun,$permissions,$tableColumns);
        $this->Editor->setConfig('tableInfo','cellpadding="1" width="960"
class="mateTable"');
        $this->Editor->setConfig('orderByColumn','codigosie');
    }
}

```



```

$this->Editor->setConfig('addRowTitle','Nueva Unidad');
$this->Editor->setConfig('editRowTitle','Editar Unidad');
// $this->Editor->setConfig('iconTitle','Editar Unidad');
$this->Editor->setConfig('viewQuery',false);
}

function Example1()
{
    if(isset($_POST['json']))
    {
        session_start();
        // Initiating lang vars here is only necessary for the logError, and
mysqlConnect functions in Common.php.
        // If you are not using Common.php or you are using your own
functions you can remove the following line of code.
        $this->langVars = new LangVars();
        $this->mysqlConnect();
        if(ini_get('magic_quotes_gpc'))
        {
            $_POST['json'] = stripslashes($_POST['json']);
        }
        if(function_exists('json_decode'))
        {
            $data = json_decode($_POST['json']);
        }
        else
        {
            require_once('php/JSON.php');
            $js = new Services_JSON();
            $data = $js->decode($_POST['json']);
        }
        if(empty($data->info) && strlen(trim($data->info)) == 0)
        {
            $data->info = "";
        }
        $this->initiateEditor();
        $this->Editor->main($data->action,$data->info);
        if(function_exists('json_encode'))
        {
            echo json_encode($this->Editor->retArr);
        }
        else
        {
            echo $js->encode($this->Editor->retArr);
        }
    }
}

```



## 4. PRUEBAS Y CALIDAD DE SOFTWARE

### 4.1 INTRODUCCIÓN

Se realiza la descripción de las pruebas y resultados obtenidos en la implantación de software desarrollado de Caja Blanca y Caja Negra, para determinar el rendimiento del sistema, y la aplicación de métricas para cuantificar la calidad de software que fue desarrollado.

### 4.2 PRUEBAS DEL SISTEMA

Las pruebas a realizar tienen como características principales, identificar los posibles errores que puedan existir, para así reducir en gran cantidad el riesgo de impacto en el Sistema, además que nos permitirá ofrecer un software de calidad que cumpla con los requerimientos y las expectativas del usuario final.

#### 4.2.1 PRUEBA DE CAJA BLANCA

Se considera a la prueba de Caja Blanca como uno de los tipos de pruebas más importantes que se aplica al software, logrando como resultado que disminuya en un gran porcentaje el número de errores existentes en los sistemas y por ende una mayor calidad y confiabilidad.

La prueba de Caja Blanca o prueba de Cristal está orientada al código del sistema.

Busqueda\_UE()

1. Validar\_cod(codigo)
2. if Validar\_cod  $\diamond$  parámetros no válidos then
3. conectar\_BD bd\_distrital, “SELECT \* FROM unidad\_educativa WHERE codigosie='codigo' ”
4. if(Existe\_codigo  $\diamond$  Null)then
5. Mostrar Datos de la Unidad Educativa

6. Mostrar: Código de la Unidad Educativa

Mostrar: nombre\_UE

Mostrar: codigosie

Mostrar: turno:UE

Mostrar: tipo\_UE

Mostrar: turno\_UE

Else

7. mensaje2, ir a paso 11

end if

Else

8. Mensaje1, ir a paso 11

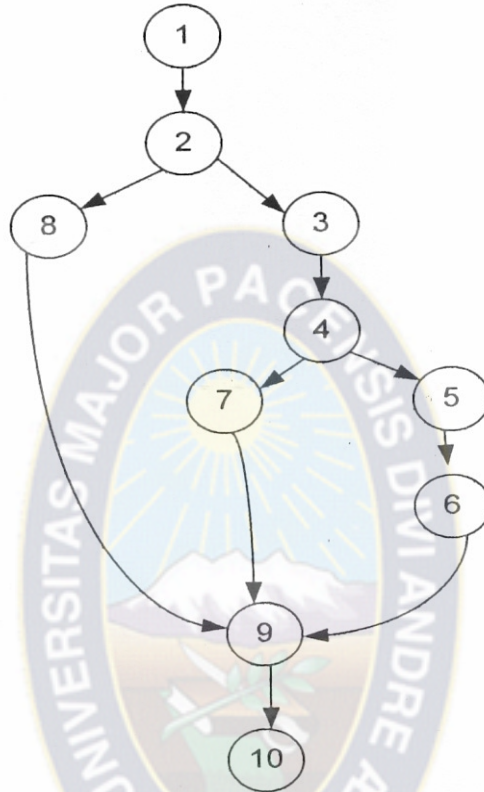
end if

9. Salir

10. End



A continuación la descripción del algoritmo mediante grafos (Figura 4.1)



**Figura 4.1 Grafo de Flujo-Búsqueda de Unidad Educativa**

Fuente: Elaboración Propia

Determinamos la complejidad Ciclomática del grafo de flujo  $V(G)$ , donde  $V(G)$  es el número de caminos básicos independientes que compone el grafo y por tanto el número de pruebas que se debe realizar y ejecutar para garantizar que se cubren todas las sentencias del programa.

Entonces  $V(G)$  viene dado por:

$$V(G) = \text{Nro. de Aristas} - \text{Nro. de Nodos} + 2$$

$$V(G) = 11 - 10 + 2$$

$$V(G) = 3$$

Los caminos básicos linealmente independientes para la Búsqueda de Solicitud son 3:

**Camino 1** : 1 – 2 – 8 – 9 – 10

**Camino 2** : 1 – 2 – 3 – 4 – 7 – 9 – 10

**Camino 3** : 1 – 2 – 3 – 4 – 5 – 6 – 9 – 10

#### **Caso de Prueba Camino 1**

Validar\_cod (codigo)

Validar\_cod == parámetros no válidos

Mensaje 1

Resultado esperado: Código no válido

#### **Caso de Prueba Camino 2**

Validar\_cod(codigo)

Validar\_cod <> parámetros no válidos

Existe\_codigo()

Existe\_codigo() == Null

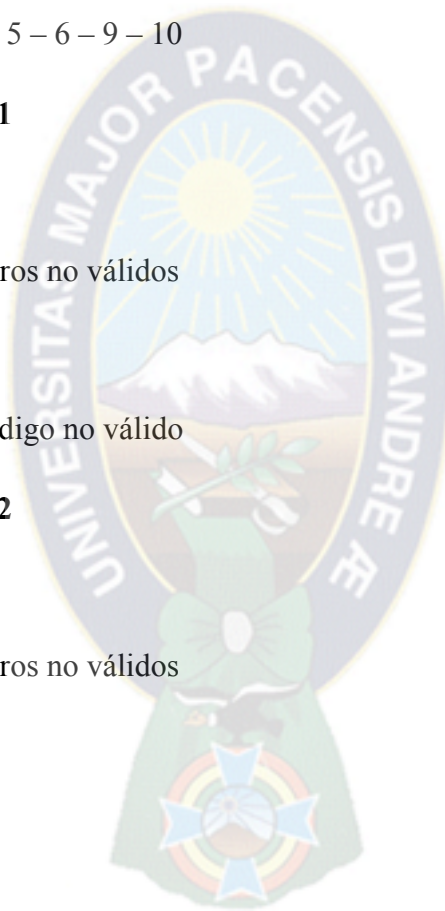
Mensaje 2

Resultado esperado: No existe la Unidad Educativa

#### **Caso de Prueba Camino 3**

Validar\_cod(código)

Validar\_cod <> parámetros no válidos



Existe\_codigo()

Existe\_codigo() <> Null

Mostrar\_información

Resultado esperado: Unidad Educativa encontrada

En la figura 4.2 se detalla la prueba de Caja Blanca

The figure shows a comparison of search results in a web application. The left side shows a successful search for 'HUGO DAVILA B', displaying a table with one record. The right side shows an unsuccessful search for 'XXXXXX', displaying a message that no records were found.

Código	Unidades Educativas	Dirección U.E.	Turno	Tipo U.E.	Teléfono U.E.	Red U.E.	Zona U.E.	Técnico Asignado	Director Establecim
80730433	HUGO DAVILA B	Av. Carrasco Final	Tarde	Fiscal	2245406	707	Miraflores	Amerson Jose	ANDRADE GERMAN

Figura 4.2 Prueba de Caja Blanca

Fuente: Elaboración Propia

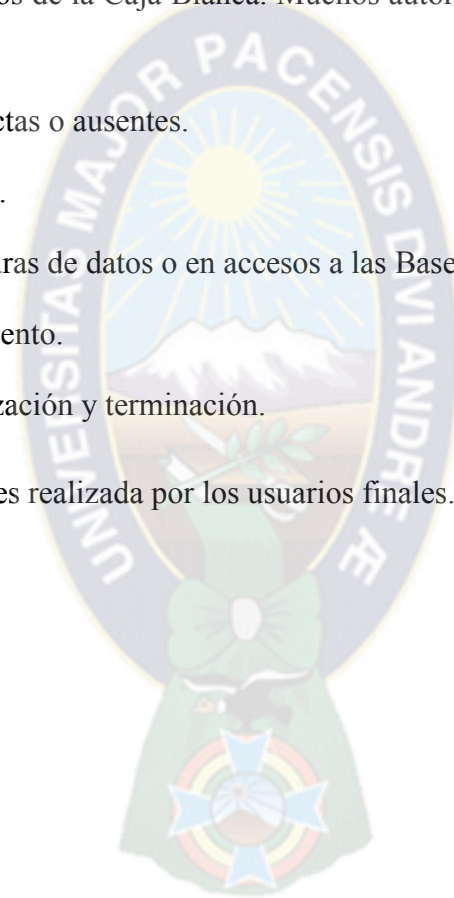
#### 4.2.2 PRUEBA DE CAJA NEGRA

La prueba de Caja Negra se centra en los requisitos funcionales del sistema, es decir, que examina algunos aspectos del modelo fundamental del sistema sin tener en cuenta la estructura lógica interna.

La prueba de Caja Negra no es una alternativa a las técnicas de prueba de la Caja Blanca, sino un enfoque complementario que intenta descubrir diferentes tipos de errores a los encontrados en los métodos de la Caja Blanca. Muchos autores consideran que estas pruebas permiten encontrar:

1. Funciones incorrectas o ausentes.
2. Errores de interfaz.
3. Errores en estructuras de datos o en accesos a las Bases de Datos externas.
4. Errores de rendimiento.
5. Errores de inicialización y terminación.

La prueba de Caja Negra es realizada por los usuarios finales.





## INICIO DEL SISTEMA



## Lista de Unidades Educativas

**Lista de Unidades Educativas**

Adición Unidad Educativa CIRCULARES E INSTRUMENTOS

Código Site	Nombre U.E.	Dirección U.E.	Turno	Tipo	Teléfono U.E.	Red U.E.	Zona U.E.	Técnico Asignado	Item Director	Acciones
80730645	12 DE ABRIL	Calle 7 Obreros	Noche	Alternativo	s/t	505	Sur	355	10100	[icon]
80730095	16 DE JULIO	s/d	Mañana	Fiscal	2453070	701	Centro	366	14904	[icon]
80730173	1RO. DE MAYO	Batallon Elimani #1 656	Tarde	Fiscal	s/t	702	Centro	366	12100	[icon]
80730584	20 DE OCTUBRE II	Calle Comercio #1043	Noche	Alternativo	s/t	100	Centro	366	14915	[icon]
80730642	21 DE SEPTIEMBRE	Calle Yanacocha #983	Noche	Alternativo	s/t	100	Centro	366	14923	[icon]
80730386	5 DE JUNIO A	Calle Federico Jofre #69	Mañana	Fiscal	s/t	503	Sur	355	66148	[icon]
80730307	6 DE JUNIO B	Calle Federico Jofre #47	Tarde	Fiscal	s/t	503	Sur	355	41312	[icon]
80730729	ARIPANE	Finca Calle 19 Robinsons Divno Maestro	Mañana	Alternativo	s/t	505	Sur	355	8800	[icon]
80730380	ACHUMANI	Calle 40 Av. Las Madres s/c	Mañana	Fiscal	s/t	502	Sur	355	20774	[icon]
80730696	ADAPTACION INFANTIL	Nector Ormachea Obregón Calle 5	Mañana	Alternativo	s/t	505	Sur	355	8760	[icon]
80730401	ADELA ZAMUDIO	Calle Campesino #51	Noche	Fiscal	2340251	705	Miraflores	377	62804	[icon]

## Añadir Unidad Educativa

LOGIN

### Añadir Unidad Educativa

Codigosite:

Cod\_tecnico: 377

Nroitem\_direct: 12100

Nombre\_UE:

Turno\_UE: Mañana

Direccion\_UE:

Tipo\_UE: Fiscal

Teléfono\_UE:

Red\_UE:

Zona\_UE: Centro

## Añadir Director

### Antes Añadir Director:

Nroitem\_direct:

Nombre\_direct:

Apepat\_direct:

Apemat\_direct:

Sexo\_direct: Varón

Cl\_direct:

Teléfono\_direct:

Domicilio\_direct:

Cel\_direct:

Correo\_direct:

## Editar Unidad Educativa

LOGIN

### Editar Unidad

Unidades Educativas: ESCUELA DE DEPOR

Dirección U.E.: s/d

Turno: Mañana

Tipo U.E.: Alternativo

Teléfono U.E.: s/t

Red U.E.: 708

Zona U.E.: Miraflores

Técnico Asignado: Jimenez Jose

Director Establecimiento: CUSICANQUI JUAN CARLOS

### Editar Director

Nroitem\_direct: 10100

Nombre\_direct: ROXANA

Apepat\_direct: ROMERO

Apemat\_direct: ENRIQUEZ

Sexo\_direct: Mujer

Cl\_direct: s/c

Teléfono\_direct: s/t

Domicilio\_direct: s/d

Cel\_direct: s/c

Correo\_direct: s/c

Correoalt\_direct:

Tabla 4.1 Prueba de Caja Negra

Fuente: Elaboración Propia

### 4.3 CALIDAD DE SOFTWARE

La norma ISO 9126 permite especificar y evaluar la calidad del software desde diferentes criterios asociados con adquisición, requerimientos, desarrollo, uso, evaluación, soporte, mantenimiento, aseguramiento de la calidad y auditoria de software.

Hoy en día la calidad de software es una parte importante del desarrollo de software, el objetivo de la calidad es mejorar la experiencia del usuario final al usar el producto. El estándar que establece los parámetros de calidad de software es ISO 9126, las métricas de calidad están dadas por:

#### **Funcionalidad**

Se utilizan métricas orientadas a medir la funcionalidad.

Para calcular los puntos función (PF) se utiliza la ecuación:

$$PF = \text{Cuenta total} = x[x + \min(y) * \bar{f}_i]$$

Dónde:

*PF* :Medida de funcionalidad

CT: La suma de los datos

X: Nivel de confiabilidad del sistema

Y: Nivel de significación de error

Fi: Valores de complejidad

Parámetros de medición	Factor de Ponderación				
	Número	Simple	Medio	Complejo	
Número de entradas de usuario	8	4	5	7	40
Número de salida de usuario	12	5	6	8	72
Número de peticiones de usuario	6	3	4	6	24
Número de Archivos	50	10	13	18	650
Número de archivos Número de interfaces externas	30	5	7	10	210
TOTAL					996

**Tabla 4.2. Calculo Punto Función**

Fuente: Elaboración Propia

Los factores de ajuste tienen un rango del 0 al 5, siendo 0 sin importancia, 1 incidental, 2 moderado, 3 medio, 4 significativo y 5 esencial.

Reemplazando los valores en la ecuación PF se obtiene lo siguiente:

$$PF = 196 * 0.65 + 1.01 * 6]$$

$$PF = 205.16 \text{ punto función real obtenido}$$

$$PF_{esperado} = 196 * 0.65 + 1.01 * 0]$$

$$PF_{esperado} = 344.6$$

Calculando el porcentaje de funcionalidad se tiene:  $\%PF = \frac{PF}{PF_{esperado}} = \frac{205.16}{344.6} = 0.89$

Se concluye que el sistema tiene una funcionalidad del 89%.

### **Mantenibilidad**

Se usa el estándar IEEE 982, ya que sugiere un índice de madurez del software que proporciona una estabilidad del producto.

Mediante la siguiente ecuación se calcula el índice de madurez de software.

$$IMS = \frac{MT - (Fa + Fc + Fd)}{MT}$$

MT: Número de módulos de la versión

Fc: Número de módulos de la versión actual que se han cambiado

Fa: Número de módulos de la versión actual que se han dañado

Fd: Número de módulos de la versión que se han borrado con la versión actual.

Entonces tenemos:

$$IMS = \frac{5 - 0 + 1}{5}$$

$$IMS = 0.8$$

Se observa que el sistema tiene un 80% de mantenibilidad.

### Usabilidad

Hay distintas formas de medir la usabilidad de un sistema, siendo una de las más importantes la que usamos brindándonos una vista general en la percepción del usuario ante el sistema.

Las ponderaciones dadas en la evaluación se reflejan en la siguiente tabla

<i>ESCALA</i>	<i>VALOR</i>
5	Muy bueno
4	Bueno
3	Regular
2	Malo
1	Muy Malo

**Tabla 4.4. Ponderación de evaluación**

Fuente: Elaboración Propia

Mediante las pruebas y estudios realizados con los usuarios finales, se determinaron los siguientes datos:

N°	Pregunta	Evaluación
1	¿Se ha satisfecho todos los requerimientos establecidos?	4
2	¿Le resulta fácil de recordar las ordenes y aprender las operaciones básicas?	5
3	¿Considera Ud. Que es una herramienta útil?	5
4	¿Presenta la suficiente ayuda durante el tiempo que accede al sistema?	3
5	¿Cómo considera el formato de las salidas que genera el sistema?	4
6	¿El sistema tiene seguridad necesaria?	5
7	¿Los informes que presenta son representativos?	5
8	¿Los informes ayudan a la toma de decisiones?	3
9	¿Cómo le parece el tiempo de ejecución de tareas?	4
	TOTAL	38

**Tabla 4.5 Cuestionario en el aspecto de usabilidad**

Fuente: Elaboración propia

$$FU = \frac{\sum x_i}{9} * 100/5$$

Siendo  $\sum x_i$  el total de la tabla se tiene:

$$FU = \frac{38}{9} * 100/5$$

$$FU = 84.44\%$$

Se observa que el sistema tiene un grado de usabilidad del 84.44%

### **Portabilidad**

A nivel Sistema operativo el ordenador del cliente funciona bajo cualquier sistema operativo, teniendo como opción principal Windows 7/8.

El software funciona y es portable bajo los principales navegadores siendo los principales y en los que se probó Internet Explorer, Mozilla Firefox y Chrome.

Por lo que se concluye que el sistema del lado del cliente es portable independientemente del sistema operativo pero dependientemente del navegador y las condiciones en las que se ejecuta.

## **CAPÍTULO 5. CONCLUSIONES Y RECOMENDACIONES**

### **5.1 CONCLUSIONES**

Se llegó a alcanzar el objetivo principal, cuyo producto principal es el Sistema de Información y Control en la Dirección Distrital de Educación La Paz-2, para mejorar la administración de la información de las unidades educativas pertenecientes al distrito 2.

Se alcanzó abstraer toda la información necesaria para la adecuada construcción de la Base de Datos y un correcto funcionamiento, sustituyendo en un gran porcentaje los procesos manuales a procedimientos de almacenamiento, consultas de información, además que se resguarda toda la información de las Unidades Educativas, evitando un nuevo registro de la misma unidad educativa, ya que daría como resultado una confusión de registros que se puedan realizar al mismo.

El acceso a la información en consultas que brinda el sistema sobre la información de las unidades educativas es segura y fiable, hasta el momento gracias al módulo de administración se logra la carga y actualización de datos de las unidades educativas.

El rendimiento del sistema de Información y Control de la Dirección Distrital fue aceptable, debido a que paso satisfactoriamente las pruebas de Caja Blanca (lleva un código aceptable y lógico) y las pruebas de Caja Negra (constituye una interfaz amigable con el usuario), al igual que el tratamiento eficaz de la información de las Unidades Educativas, certificado por los usuarios finales.

Y por último es considerado un sistema de calidad ya que cumple con las normas de calidad establecidas en la ISO/9126.

### **5.2 RECOMENDACIONES**

Con la finalización del presente proyecto se realizan las siguientes recomendaciones.

Para el sistema:

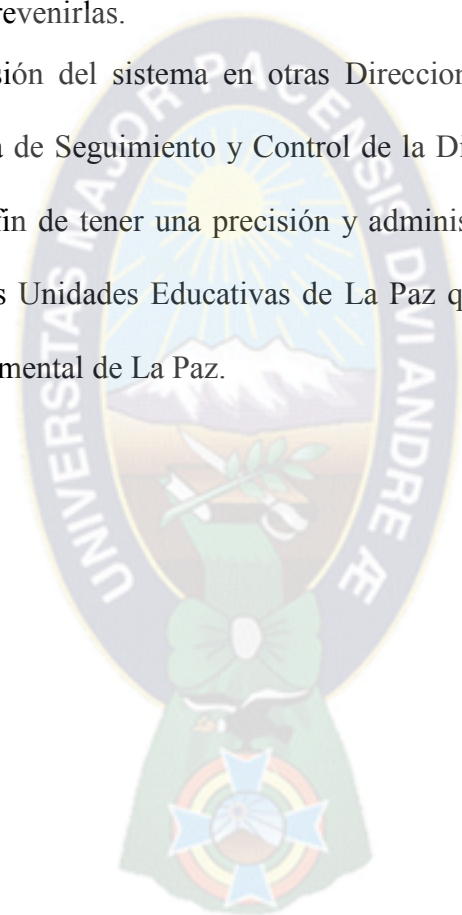
- Realizar mantenimiento preventivo del sistema cada 3 meses.



- Efectuar copias de seguridad una vez a la semana para evitar pérdidas cuantiosas de información.
- Cambiar los passwords del sistema y servidores cada determinado tiempo establecido por la institución.

Para la institución:

- Actualizar el servidor y el gestor de base de datos constantemente para corregir posibles fallas y prevenirlas.
- Realizar la extensión del sistema en otras Direcciones Distritales, tomando como modelo el Sistema de Seguimiento y Control de la Dirección Distrital de Educación La Paz-2, con el fin de tener una precisión y administración más profunda sobre la información de las Unidades Educativas de La Paz que son atendidos dentro de la Dirección Departamental de La Paz.



## BIBLIOGRAFÍA

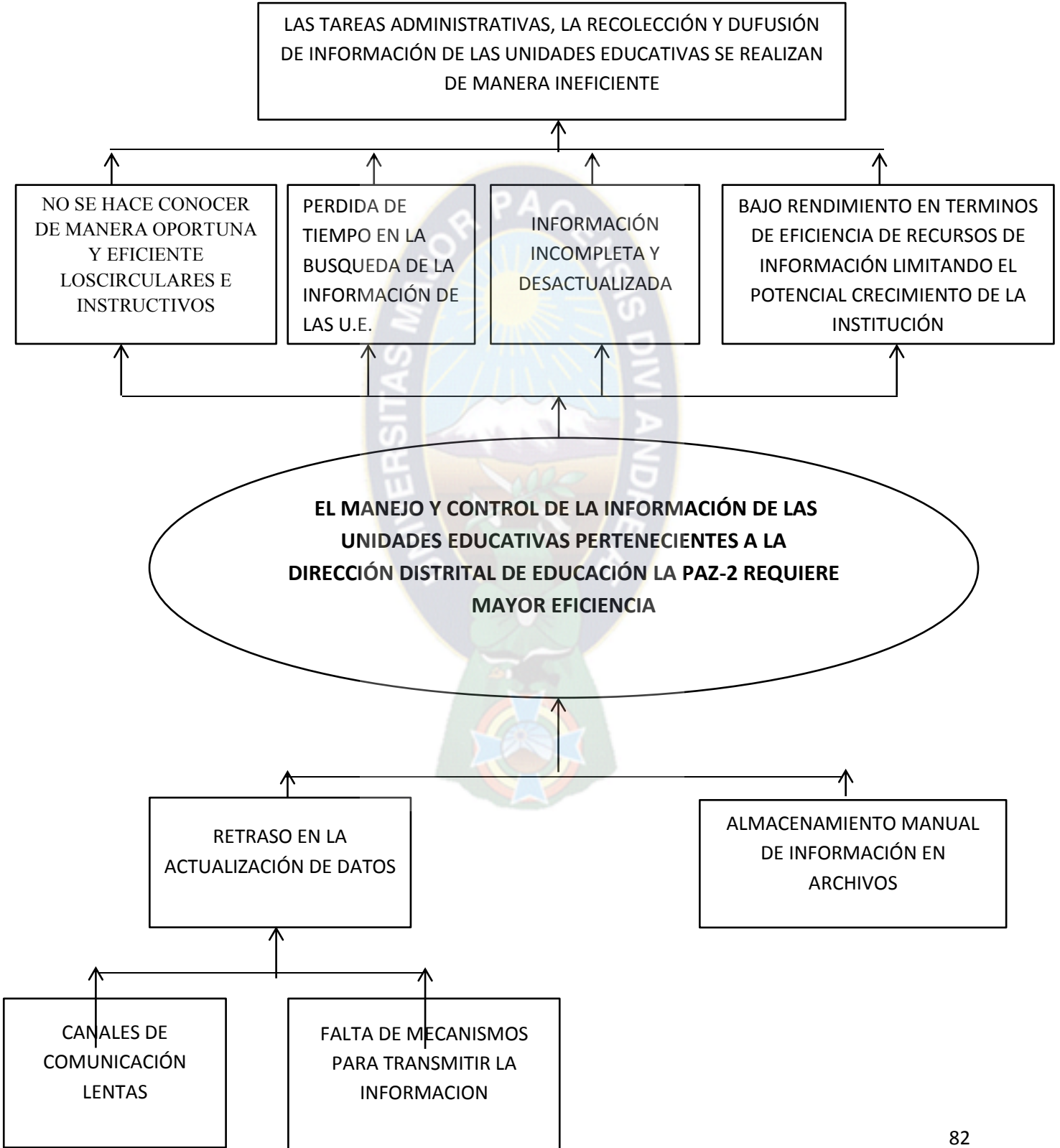
- [COBO, 2005] Angel Cobo. “Php y Mysql”. España: ediciones Díaz de Santos 2005, 4ta edición.
- [CAMPDERRICH, 2003] Campderrich, B. “Ingeniería de Software”. España: editorial UOC, 2004, 1ra edición.
- [GOMEZ, 2011] Alvaro Gomez Vieites. “Enciclopedia de la Seguridad Informática”. México: ediciones Alfaomega , 2011, 2da edición.
- [KENDALL & K. 1997] Kendall & Kendall. “Análisis y Diseño de Sistemas”. Mexico: ediciones Prentice Hall Hispanoamericana S.A., 1997, 3ra edición.
- [LARMA,1999] Larman Carig “UML y patrones”, México 1999, 1ra edición.
- [O’BRIEN, J. 2001] O’Brien, James “Sistemas de Información Gerencial”. 4ta edición.
- [PRESSMAN, 2003] Roger S. Pressman 2003, Ingeniería del Software un Enfoque Practico, 5ta edición.
- [SANCHEZ. 2010] Oscar Sanchez. “*Aplicaciones informáticas de base de datos relacionales*”. España: ediciones Parainfo S.A. 2010, 1ra edición.
- [SCHUMULLER, J. 2000] Joseph Schumuller. “*Aprendiendo uml en 24 horas*”. México. Linusa, 2000, 2da edición.

## REFERENCIAS WEB

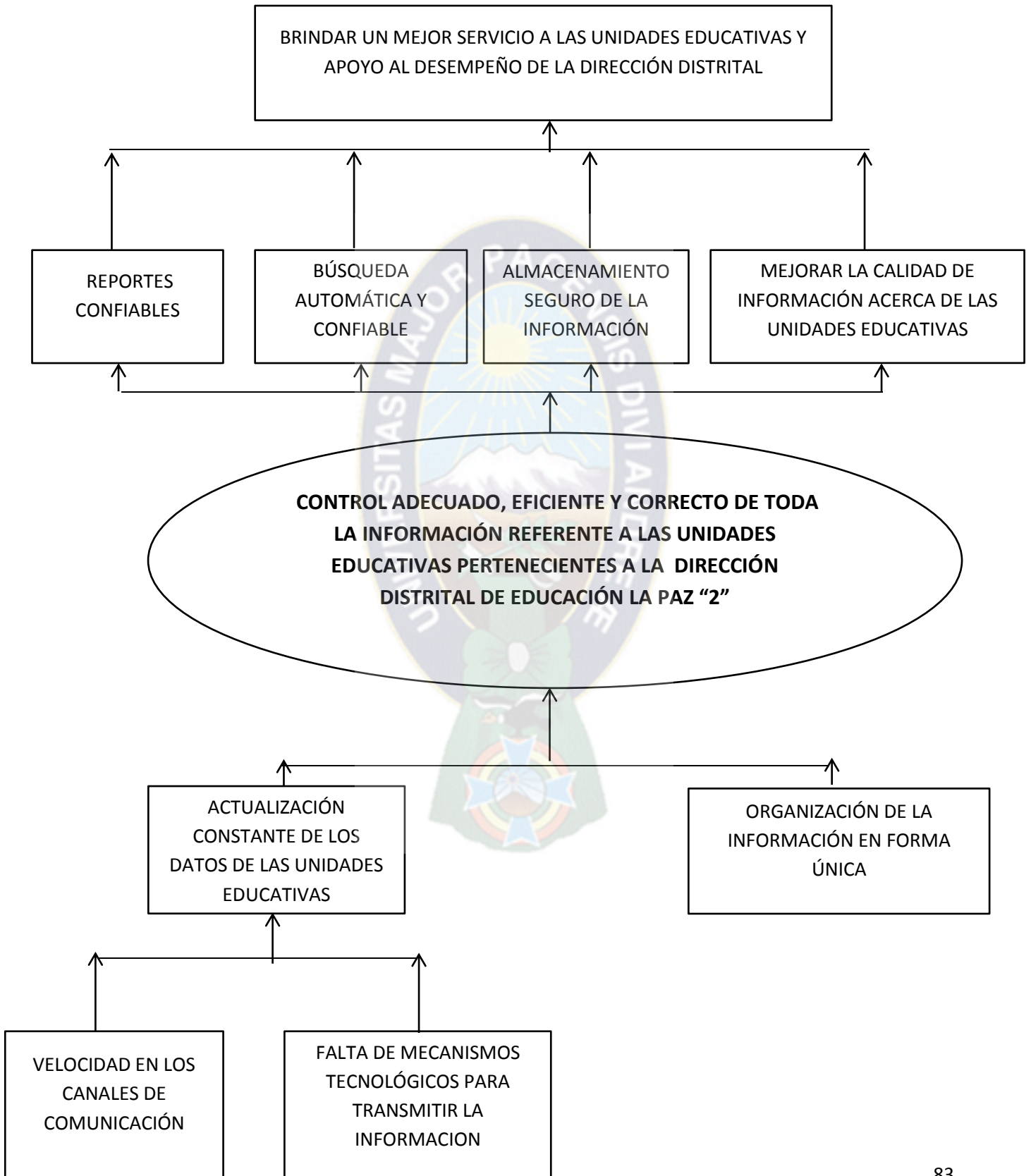
- [Iconix.2010] “Modelamiento de datos con iconix”, Obtenido el 12 se Septiembre del 2014 desde:  
<http://informatica-v-iconix.blogspot.com/2011/08/normal-0-21-false-false-false-es-x-none.html>
- [Grau. 2008] “Desarrollo orientado a objetos con uml”, Obtenido el 15 se Septiembre del 2013 desde:  
<http://www.cmcrossroads.com/bradapp/docs/patternsintro.html>
- [MARCO LÓGICO. 2012] “Como hacer proyectos marco lógico”, Obtenido el 8 se Abril del 2013 desde :  
<http://www.slideshare.net/luispretell/como-hacer-proyectos-marco-logico-11592983>

# ANEXOS

## ÁRBOL DE PROBLEMAS



## ÁRBOL DE OBJETIVOS



## MATRIZ DE MARCO LÓGICO

RESUMEN NARRATIVO	INDICADORES VERIFICABLES	MEDIOS DE VERIFICACIÓN	SUPUESTOS
<p><b>FIN</b></p> <p>Mejorar el servicio en la Dirección Distrital de Educación La Paz-2, para reducir el trabajo que se realiza manualmente.</p> <p>Desarrollando un sistema de información acorde a las necesidades de la dirección distrital.</p>	<p><b>MEDIDAS DEL LOGRO DEL FIN</b></p> <ul style="list-style-type: none"> <li>• Reducir en un 80% el trabajo manual.</li> <li>• Reducir en un 90% el retraso de información</li> </ul>	<p>Entrevista con los responsables de la Dirección Distrital de Educación La Paz-2.</p>	<ul style="list-style-type: none"> <li>• Aprobación directa de la dirección administrativa de la institución</li> <li>• La dirección y el personal aceptan los beneficios que ofrece el Sistema de Información.</li> <li>• Adecuada planificación para el desarrollo y su puesta en ejecución del proyecto.</li> </ul>
<p><b>PROPÓSITO DEL PROYECTO</b></p> <p>Automatizar los servicios para contar con información confiable y eficiente, mediante un sistema de seguimiento con un control eficiente, correcto y mayor rapidez de toda la información referente a la Dirección Distrital de Educación La Paz-2 bajo medidas de seguridad, confiabilidad y rapidez.</p>	<ul style="list-style-type: none"> <li>• Se desarrolla un Sistema de Información de Control, el cual comenzará a funcionar en el mes de Noviembre del 2014, mejorando la disponibilidad y accesibilidad a la información de las unidades educativas del distrito 2.</li> <li>• Se realiza un mejor seguimiento de las unidades educativas con el empleo del sistema para el registro y control de los mismos a partir del 2º semestre del 2014.</li> </ul>	<ul style="list-style-type: none"> <li>• Información registrada por parte de la dirección administrativa de la Dirección Distrital de Educación LP.-2.</li> <li>• Software y documentación del sistema entregada a la dirección Distrital.</li> <li>• Obtener avales correspondientes de conformidad con el desarrollo del proyecto.</li> </ul>	<ul style="list-style-type: none"> <li>• El sistema estará implementado en noviembre del 2014.</li> <li>• Facilidad en la obtención de información sobre los procesos manuales que se sistematizan.</li> <li>• Facilidad en el uso de las tecnologías necesarias para el desarrollo y ejecución del sistema.</li> <li>• Facilidad de las herramientas a utilizarse para el desarrollo.</li> </ul>

<p><b>COMPONENTES RESULTADOS</b> <b>O</b></p> <p>Desarrollo e Implementación de un Sistema de Información de Seguimiento y Control para la Dirección Distrital de La Paz-2. Con los siguientes Módulos:</p> <ul style="list-style-type: none"> <li>➤ Seguimiento y control de las Unidades Educativas.</li> <li>➤ Asignación de consultas.</li> <li>➤ Instructivos: mensuales y semestrales que se requieran.</li> <li>➤ Consulta y tratamientos.</li> </ul>	<ul style="list-style-type: none"> <li>➤ Se realiza la recopilación de la información de las fichas de registro manuales.</li> <li>➤ Generar instructivos y reportes automatizados que proporcionen información actualizada de los requerimientos de la Dirección Distrital.</li> <li>➤ Las solicitudes de información por unidad educativa reducen el tiempo de espera, en la</li> </ul>	<ul style="list-style-type: none"> <li>➤ Sistema de información para el seguimiento y control de las unidades educativas de la Dirección Distrital de Educación desarrollado y funcionando en equipos de computación al interior de la institución.</li> <li>➤ Aprobación del proyecto de grado de la carrera de Informática.</li> <li>➤ Informe de pruebas realizado por el sistema</li> </ul>	<ul style="list-style-type: none"> <li>➤ La dirección administrativa de la institución se encuentra en total acuerdo con la implantación y ejecución del sistema.</li> <li>➤ Disponibilidad de información y tiempo por parte del personal de la institución.</li> <li>➤ Personal capacitado en el manejo del sistema.</li> </ul>
--	---	---	---

búsqueda de datos.

## ACTIVIDADES

1. Realizar el levantamiento de la información mediante entrevistas y cuestionarios a todos y cada uno de los encargados de los procesos de la Dirección Distrital de Educación. Organizar la información recopilada de una manera eficiente y clara.

Realizar los respectivos diagramas de procesos de la situación actual.

2.

2.1. Emplear la fase de elaboración para iniciar el proyecto y planificar el desarrollo del producto.

2.2. Realizar el proceso de análisis en las actividades de registro de las unidades educativas de la Dirección Distrital-2.

2.3. Incorporar los requerimientos,

## COSTOS DEL SSITEMA DE INFORMACIÓN

Costos

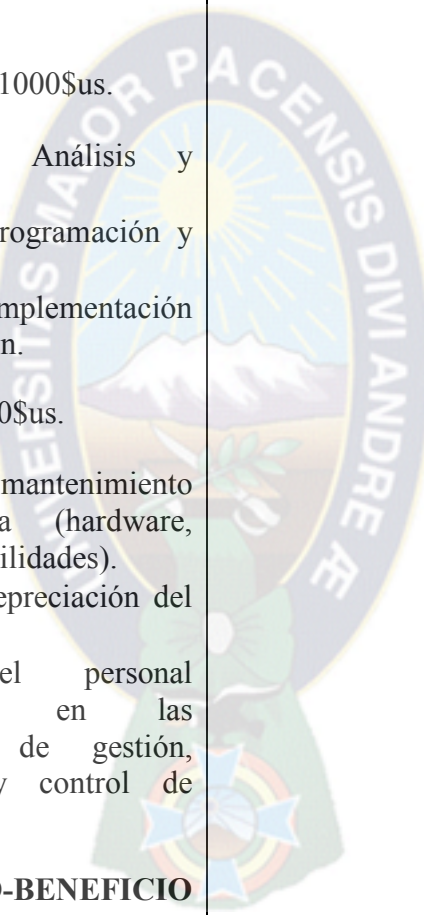
- Costos de acondicionamiento del lugar destinado al equipo.
- Costos de los gestores y del personal encargado del acondicionamiento.

Costos de puesta a punto 250\$us.

- Costos de software

- Disponibilidad de la información completa de todas las unidades educativas de la Dirección Distrital.
- Disponibilidad de la información por parte de todos los encargados de los procesos afectados.
- Aceptación de algún recurso necesario, por parte de la dirección para llevar a cabo el proyecto.
- Conocimiento pleno y exacto de los requerimientos para llevar a cabo el proyecto.
- Enfoque adecuado de los procesos reales de registro de información para una representación eficiente.



<p>operaciones y actividades del sistema mediante un respectivo proceso de diseño.</p> <p>3.</p> <p>3.1. Modelo de arquitectura del sistema.</p> <p>3.2. Mapeo de los diseños para su codificación.</p> <p>3.3. Pruebas de instalación y configuración del sistema.</p> <p>3.4. Entrega de la documentación completa del sistema.</p>	<p>(Licencias de uso para la base datos Mysql).</p> <ul style="list-style-type: none"> <li>• Costos de las actividades de búsqueda y contratación del personal.</li> </ul> <p>Costos del software 1000\$us.</p> <ul style="list-style-type: none"> <li>• Costos de Análisis y Diseño.</li> <li>• Costos de programación y codificación.</li> <li>• Costos de Implementación y capacitación.</li> </ul> <p>Costos continuos 250\$us.</p> <ul style="list-style-type: none"> <li>• Costos de mantenimiento del sistema (hardware, software y utilidades).</li> <li>• Costos de depreciación del hardware.</li> <li>• Costos del personal involucrado en las actividades de gestión, operación y control de sistema.</li> </ul> <p><b>TOTAL COSTO-BENEFICIO 1500 \$us.</b></p>		<ul style="list-style-type: none"> <li>• Una implementación adecuada y segura del sistema.</li> <li>• Capacitación del personal para un adecuado uso del sistema.</li> <li>• Documentación clara y completa del sistema que satisface a la dirección de la institución y a todo el personal.</li> </ul>
---	---	---	---