

UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERÍA



TRABAJO DE GRADUACION
PARA OPTAR AL GRADO DE
INGENIERO EN CIENCIAS DE LA COMPUTACIÓN

**DESARROLLO DE SISTEMA BIBLIOTECARIO ESTANDAR
TOMANDO COMO BASE LA ESTRUCTURA DE LA CEPAL, CASO
PRACTICO PARA LA CASA DE LA CULTURA DEL MUNICIPIO DE
SOYAPANGO**

PRESENTADO POR:
RIVAS SALGADO, JAIRO BENJAMIN
VALIENTE MARTINEZ, CESAR ADOLFO PAOLO

ASESOR:
ING. RAUL MARTINEZ RIVAS

MARZO 2009

EL SALVADOR, CENTROAMÉRICA

**UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERIA**



**ING. FEDERICO MIGUEL HUGUET RIVERA
RECTOR**

**INGA. YESENIA XIOMARA MARTÍNEZ OVIEDO
SECRETARIO GENERAL**

**ING. ERNESTO GODOFREDO GIRON
DECANO FACULTAD DE INGENIERIA**

UNIVERSIDAD DON BOSCO
FACULTAD DE INGENIERÍA



**DESARROLLO DE SISTEMA BIBLIOTECARIO ESTANDAR
TOMANDO COMO BASE LA ESTRUCTURA DE LA CEPAL, CASO
PRACTICO PARA LA CASA DE LA CULTURA DEL MUNICIPIO DE
SOYAPANGO**

ING. RAUL MARTINEZ RIVAS
ASESOR

ING. MELVIN ALFONSO CARIAS RODESNO
LECTOR

Agradecimientos.

- A Dios todo poderoso por darme la fuerza e iluminar mi camino para llevar a cabo el desarrollo de este trabajo de graduación y a la vez por colmar mi vida de bendiciones.
- A mis padres por su apoyo y consejos he llegado a finalizar este trabajo de graduación.
- A mi hermano por el apoyo que siempre me brindo.
- Al Dr. Enrique García Prieto y su familia por el apoyo mostrado durante toda mi carrera
- A mis familiares por estar pendientes del desarrollo de este trabajo de graduación.
- A mis amigos y amigas por su confianza y lealtad, especialmente a Paolo Valiente
- A nuestro asesor y lector por el tiempo dedicado en el desarrollo de este trabajo de graduación.
- A todos aquellos que estuvieron al pendiente del desarrollo de este trabajo de graduación.

Jairo Benjamín Rivas Salgado

Agradecimientos.

Primero quiero agradecer a Dios todo poderoso, por haberme permitido finalizar mis estudios e iluminarme durante tantas veces dándome la sabiduría necesaria para poder superar todos los retos que se me presentaron a lo largo de la carrera.

Quisiera agradecerles a mis padres que durante tanto tiempo me brindaron su apoyo, y día a día lucharon duramente por brindarme la mejor educación posible. Que estuvieron dándome la mano a lo largo de toda mi educación y en aquellos momentos en los cuales necesite, ellos no dudaron en ayudarme en todas las maneras que los necesite.

A mis hermanos que durante toda mi carrera, me brindaron su apoyo y entendieron que hay ocasiones en las cuales se tiene que dejar todo de lado y orientarse sobre los objetivos.

Mis amigos que me apoyaron y me brindaron su apoyo emocional y junto hemos finalizado y superado todos los retos que se nos presentaron a lo largo de la carrera y retos cotidianos.

Mi agradecimiento a la Universidad Don Bosco, que me brindo la oportunidad de aprender tanto, durante todo este tiempo, a los docentes que con tanta dedicación lograron brindarme las herramientas necesarias para enfrentar los retos de la vida laboral.

ÍNDICE GENERAL

I. ANTECEDENTES.....	ii
II.I PLANTEAMIENTO DEL PROBLEMA.....	v
II.II DEFINICIÓN DEL TEMA.....	vi
II.III JUSTIFICACIÓN DEL TEMA.....	vii
III.I Objetivo General.....	x
III.II Objetivos Específicos.....	x
IV ALCANCES.....	xi
V LIMITACIONES.....	xii
VI Proyección Social.....	xvi
Struts Validator.....	62
Configuración del Framework.....	62
<u>La configuración se realiza mediante dos ficheros de propiedades en formato XML. Además para utilizar este Framework se debe declara en el struts-</u>	
<u>config.xml.....</u>	<u>62</u>
<u>Validator-rules.xml.....</u>	<u>62</u>
<u>validation.xml.....</u>	<u>64</u>
<u>Struts-config.xml.....</u>	<u>65</u>
INTRODUCCION A	69
UDBOOK.....	69
UDBOOK.....	72
Modulo.....	72
Sistema Bibliotecario UDBOOK.....	78
Modulo.....	78
ANEXOS	95
Anexo B.....	105

INTRODUCCIÓN

Las bibliotecas representan la fuente principal de información necesaria para la búsqueda de conocimiento de acuerdo a las necesidades de los diferentes usuarios, por lo que es necesario organizar dicha información de manera que sea fácil y rápido poder acceder a ella, esto representa el fundamento de la automatización de los sistemas bibliotecarios básicos en la actualidad.

La Casa de la Cultura de Soyapango, representa un espacio necesario para el sano esparcimiento de los jóvenes de dicha zona, actualmente existe una biblioteca pero no cuentan con un sistema informático capaz de administrarla por lo que se propone crear dicho sistema para mejorar el servicio que se presta a los usuarios. Hay que tomar en cuenta que el software que se desarrollará es de tipo estándar, éste se realizará bajo la estructura de Open Source. De igual manera dicha aplicación podrá ser instalada en cualquier biblioteca pública, privada, educativa y de gobierno.

Por lo expuesto anteriormente, ésta investigación constituirá un esfuerzo para el desarrollo del sistema bibliotecario de dicha institución, de manera que facilite su uso, sea amigable, robusto, y posea todas las características necesarias tanto requeridas por el usuarios como por los estándares internacionales de normalización de bibliotecas propuesto por la CEPAL¹.

Este documento, proporciona información de las razones por las que se lleva a cabo la investigación y los problemas ha combatir, se establecen los objetivos, alcances, limitantes y delimitantes que conllevará la realización del proyecto. Se incluye una investigación con referencias históricas, conceptuales y referenciales, tanto de los sistemas bibliotecarios en general como de las herramientas a utilizar para el desarrollo de nuestra propuesta. Finalmente se establece la metodología de trabajo y quienes serán los beneficiados con el producto final del proyecto

¹ Ver anexo A, Sistema de Información Bibliográfica de la CEPAL, Manual de Referencias.

I. ANTECEDENTES

Las bibliotecas son el pilar de la educación para todas las comunidades y son el mayor apoyo de conocimiento para los estudiantes de nivel superior, docentes e investigadores. Sin embargo las distancias geográficas y las restricciones en el uso de sus servicios impiden un eficiente aprovechamiento de los recursos que ponen a disposición de ellos. Además en estas épocas cambiantes, con renovaciones políticas, sociales y tecnológicas las bibliotecas deben ampliar sus responsabilidades y su horizonte laboral: redes, bases de datos, análisis estadístico, evaluación, diseminación y comercialización de la información, nuevos tipos de usuarios, nuevas disciplinas, nuevos formatos de información, imágenes, hipertextos, educación a distancia, telecomunicación, protección de la información, derechos de autor, criptografía y correo electrónico

Las bibliotecas más importantes del planeta cuentan entre sus servicios la consulta de material didáctico en línea, ya sea referencias, enlaces a otras bibliotecas, leer documentos completos o partes de ellos, entre otros, las más visitadas son: la biblioteca del vaticano, la red de bibliotecas MultiMatch (<http://www.multimatch.eu/>) financiada por la Comisión Europea, la cual pretende ser un motor de búsqueda multilingüe y multimedia para facilitar el acceso a los contenidos digitales relacionados con el patrimonio cultural europeo; Su contraparte, la CEPAL (Comisión Económica para América Latina y el Caribe) a creado un manual de referencias y estándares para la creación de bibliotecas virtuales, dichos estándares ayudan a enlazar la biblioteca que se construyen con el resto de bibliotecas afiliadas a dicha institución².

A estas fundaciones se les une un sin numero de instituciones educativas que en su lucha por mantenerse en la vanguardia de la información se han visto obligadas a asociarse entre si para crear una red de información que les permita atender las demandas de sus usuarios.

² Ver anexo A, Sistema de Información Bibliográfica de la CEPAL, Manual de Referencias.

Existen un número grande de bibliotecas virtuales en Internet, creadas en diferentes lenguajes de programación, por lo que resultaría difícil describir cada una de éstas, los lenguajes más usados para la creación de éste tipo de aplicación son:

- PER-PHP
- VISUAL BASIC ASP y ASP.NET
- JAVA

Las bibliotecas que han sido automatizadas están siendo transformadas o complementadas implementando bibliotecas digitales, bajo el concepto general de recopilar, almacenar y organizar información de manera digital, para poder realizar búsquedas, recuperaciones y procesamientos vía las redes de cómputo; todo bajo un ambiente sencillo para el usuario y tomando en cuenta factores fundamentales como la presentación y la representación de la información, los mecanismos de almacenamiento y recuperación, la interacción humano-computadora, la plataforma tecnológica y el ancho de banda de la red.

En la actualidad, la mayoría de bibliotecas, tanto públicas como privadas no poseen sistemas informáticos automatizados, capaces de realizar cualquier tipo de consultas, ya sean externas o internas, manejo de aplicaciones administrativas propias de la biblioteca como aplicación de multas, reservaciones, estadísticas de consultas de libros ó uso de otra clase de materiales, registro detallado de cada usuario (información personal y académica), y si poseen algún tipo de software para el manejo administrativo, es de un poco rustico o de interacción poco amigable para el usuario.

El manejo de los registros que se realizan en algunas bibliotecas que no cuentan con software para su administración es de forma manual, es decir, se tiene que llenar un afiche con el libro de texto que se quiere consultar, y luego se le entrega a la persona encargada del registro de libros, en casos en los que no se sabe que tipo de libro en concreto se necesita, se le consulta a dicha persona para que le recomiende uno.

II. IMPORTANCIA DE LA INVESTIGACIÓN

Con los nuevos avances tecnológicos, las bibliotecas se ven obligadas a adaptarse de la mejor manera posible a dichos cambios, es por eso que la creación e implementación de bibliotecas de consulta en línea es un punto de vital importancia para promover sus servicios de calidad a los usuarios, actualmente, en El Salvador se está implementando una nueva modalidad de bibliotecas virtuales, en las cuales se logrará tener acceso a cualquier tipo información a través de la red sin tener que hacer préstamo de libros³.

Por lo anterior expuesto, se ve la necesidad de crear un sistema automatizado para la consulta de material bibliográfico de forma estándar, que se rija por el sistema de codificación de la CEPAL , de ambiente agradable y sencillo, que cuente con los componentes necesarios para la administración ordenada y precisa de los registros y ubicación de los diferentes textos, escalable, accesible mediante la Web, y sobre todo desarrollado bajo plataforma de licencias gratis.

Con ésta investigación se pretende crear un software para la administración de las bibliotecas tanto publicas o privadas, caso practico para la administración de la biblioteca de la Casa de la Cultura del Municipio de Soyapango, a fin de solucionar algunos procesos manuales y poco fiables que actualmente se realizan en dicha institución, en otras bibliotecas puede ser para actualizar sus sistemas a una herramienta que les pueda disminuir costos teniendo en cuenta que el sistema será desarrollado con herramientas Open Source.

³ Universidad de El Salvador, primera publicación de documento virtual, noviembre 2001

II.I PLANTEAMIENTO DEL PROBLEMA

En la actualidad, realizar procesos de préstamo de libros, consulta de existencia de material, reposición de libros y otros procesos normales que se realizan en algunas bibliotecas que posean software adecuado para su administración es de vital importancia para el buen manejo y control adecuado por parte del encargado de dicha área, con lo cual se evita la pérdida de registros, falta de información para realizar estadísticas y otros problemas que surgen con el manejo de registros de forma manual.

Es por esto que se ve la necesidad de crear un sistema capaz de administrar la información de manera tal que permita a sus usuarios evitar hacer registros manuales de peticiones de consultas de libros, consulta de cantidad de prestamos realizados, fechas limites para devolución de material, o para consultas de reservación de material, entre otros, permitiendo acortar los tiempos de respuesta a dichas solicitudes, presentar un ambiente mas amigable tanto para el usuario cómo para el administrador, incluyendo aplicaciones propias para la administración adecuada de los registros e información extra para la generación de reportes.

Además, dicho software se creara bajo la plataforma de código abierto, con lo cual se exonera a la institución de comprar licencias para el desarrollo y uso de dicho sistema, minimizando los costos económicos para la producción de dicha aplicación

II.II DEFINICIÓN DEL TEMA

Desarrollo de sistema bibliotecario estándar tomando como base la estructura de codificación de la CEPAL, caso práctico para la Casa de la Cultura de Soyapango

Se describe de la siguiente manera:

Aplicación usuarios.

- Prestamos de libros, consulta de material, reservación de material, consulta de fechas de entregas y fechas de reservación de material, consulta de penalizaciones, renovación de material, devolución de libros, actualización de información personal.

Aplicación administrador.

Además de las aplicaciones de usuario normal se incluirán las siguientes:

- Imprimir listado por autor. título, tipo de material, año de publicación, casa editora
- Reflejar estadísticas de búsqueda por usuarios, estadísticas de préstamos, material en mora y material no encontrado.
- Control de préstamos interno y externos
- Asignar material a usuarios, material en reservación y material con mora
- Aplicar multas a usuarios y estipular fechas considerables de préstamo de material
- Agregar nuevo material a la base de datos de manera sencilla y rápida
- Dar de baja o alta a un usuario a partir de un código único generado por la Institución
- Agregar información personal de forma detallada de cada usuario
- Administración de las diferentes cuentas de usuarios, tanto de cuentas de administrador como de usuarios normales
- Otros elementos que puedan surgir durante la etapa de diseño del sistema

II.III JUSTIFICACIÓN DEL TEMA

La Casa de la Cultura de Soyapango es un lugar de esparcimiento cultural para los jóvenes de la zona, en esta se imparten diferentes cursos artísticos y literarios, poseen material didáctico para reforzar dichos contenidos y además otra clase de libros que pueden ser adquiridos por todos los inscritos a estos cursos, las personas no inscritas en ningún curso, pueden de igual forma utilizar la biblioteca si se inscriben por un año a ésta por un costo de setenta y cinco centavos.

Hace un par de años la biblioteca era frecuentada por muchos niños entusiastas de aprender, los cuales llegaban a toda hora del día, en la actualidad con el apogeo del Internet las visitas se han visto reducidas increíblemente⁴, a tal grado que hay días en los que no llega nadie, el fin ultimo de una biblioteca es poder ser parte de la educación de las personas, facilitándoles de la mejor manera posible la información, de esta forma las bibliotecas tienen que ir de la mano de las tecnologías y así poder llamar la atención de las personas y que estas entiendan la importancia de las mismas en el proceso de educación.

Para poder sobrevivir las bibliotecas actualmente deben ser capaces de brindar búsquedas rápidas de materiales, información digital en línea, pero sobretodo brindar la mejor información posible, la casa de la cultura tiene información necesaria para la educación, no solo de jóvenes del municipio si no de todo el país, por lo tanto lo único que necesita para poder seguir existiendo de manera efectiva, es poder incorporar las tecnologías a ésta.

Por lo anterior expuesto se ve la necesidad de que dichos materiales se puedan obtener ágilmente, ya sea para préstamo, reserva, renovación de material, revisión de estatus de usuario y modificación de información personal; dichas operaciones podrían ser realizadas de forma presencial o de forma remota a través de la

⁴ Ver entrevista a bibliotecaria en anexo C.

pagina en Internet de la Casa de la Cultura, en la cual se pretende dar acceso a los usuarios a través de un código.

La idea de crear un sistema para la administración de la biblioteca de la Casa de la Cultura surge de la necesidad de cambiar la forma en como administran actualmente dicha biblioteca⁵, ya que no poseen ninguna herramienta tecnológica para su adecuada gestión, como valor agregado se espera crear aplicaciones propias del administrador del sistema capaz de generar reportes consistentes, consultas fáciles y administración adecuada de la información.

Actualmente no hay forma de llevar las estadísticas de las visitas a la biblioteca, todo este proceso no queda en ningún acta, por lo que es imposible tener un control estadístico, se pretende que el programa ayude también a documentar todos los movimientos que se hagan en la biblioteca, económicos y de materiales de esta forma poder justificar donaciones en un futuro, y que sus materiales bibliográficos se miren incrementados cada vez mas.

El proceso de préstamo incluye una platica verbal con la bibliotecaria y esta pasa a buscar el ó los libros de temas afines que puedan ayudar a los usuarios en su búsqueda de la información, el tiempo de respuesta puede variar de acuerdo a la frecuencia con que este libro es solicitado ya que de estos depende la experiencia de la bibliotecaria, el proceso se agilizaría de forma notable a un par de segundos, si se utilizara un software para la búsqueda de los materiales. Por otra parte, usuarios de mayor experiencia buscan los libros directamente y aun así el proceso no es muy rápido ya que no poseen la misma experiencia que la bibliotecaria.

El software que se desarrollará tendrá la posibilidad de realizar consultas tanto internas (intranet) como externas (Internet) facilitando al usuario la comodidad y tiempo invertido en realizar peticiones, y por otra parte, se le hará más fácil al

⁵ Ver entrevista a bibliotecario en anexos C

administrador de la aplicación atender peticiones en menor tiempo, aumentando la rapidez de atención de cada usuario.

La biblioteca es abierta para cualquier tipo de personas, para poder tener acceso se tiene que pagar una tarjeta de afiliación que cuesta \$0.75 ctv por un año. El promedio de usuarios mensual es de unos 50 a 60 de las diferentes zonas del municipio, anualmente con un aproximado de 2000 visitas entre consultas rápidas y prestamos externos.

El número de empleados que laboran en dicha institución son dos personas, la encargada de la biblioteca y su asistente, posteriormente contratan más personas para dar los diferentes cursos que imparten, dichas personas trabajan por hora, por lo que no se toman como empleados fijos.

Actualmente dicha biblioteca cuenta con un aproximado de 2500 ejemplares que incluye libros, enciclopedias, además de revistas, ensayos, artículos, y otras publicaciones. Para la adquisición de material, se realiza de forma esporádica y de acuerdo a planes de donaciones por parte del gobierno u organizaciones no gubernamentales.

III OBJETIVOS

III.I Objetivo General

Desarrollar el Sistema para la administración de la Biblioteca de forma estándar tomando como base el sistema de codificación de la CEPAL caso práctico para la Casa de la Cultura del municipio de Soyapango, desarrollado bajo plataforma de código libre, facilitando con ello una mejor atención al usuario, mejorando búsquedas de literatura y en un futuro dicha biblioteca poder ser consultada por Internet.

III.II Objetivos Específicos

- Investigar aplicaciones de administración de bibliotecas en línea montadas en otras instituciones, permitiendo tener un panorama más claro de los posibles resultados esperados al finalizar éste proyecto.
- Comparar los diferentes lenguajes de programación orientados a código libre y definir el que mejor se adapte a las necesidades y limitantes de las diferentes bibliotecas, y con especial atención a la Casa de la Cultura de Soyapango.
- Analizar el flujo de información y diseñar la base de datos adecuada a dicho flujo de manera que cumpla con los estándares propuesto por la CEPAL sin dejar de lado la normalización adecuada de las diferentes tablas que formarán dicha Base de datos.
- Desarrollar los diferentes módulos que conformaran el sistema de administración de la biblioteca de la Casa de la Cultura de Soyapango
- Crear los manuales de manejo del software y manual del programador, con lo cual se tendrá un apoyo extra al momento de hacer futuras expansiones de dicho sistema

IV ALCANCES.

1. Sentar las bases del análisis y diseño del software a desarrollar que más se adapte a las necesidades, definiendo si se usara la herramienta de Lenguaje de Modelado Unificado (UML)⁶, programación en cascada o ciclo de vida del software
2. Crear los diferentes módulos (pantallas) que correspondan tanto a sesiones de usuario normal así como también a usuarios de tipo administrador, con sus respectivas opciones y sobre todo que no se dificulte su uso.
3. Minimizar el tiempo de realización de una consulta mediante el diseño de pantallas amigables y el manejo de información necesaria para realizar dichas consultas de información personal de los usuarios
4. Proporcionar al sistema de un apartado especial para la generación de reportes mensuales de los diferentes movimientos que se realizan en la biblioteca.
5. Facilitar la búsqueda de información, seleccionando parámetros ya establecidos o palabras claves introducidas por el usuario minimizando el tiempo de retardo en la obtención de la información requerida.
6. El sistema será multiempresa y su estructura será definida por el esquema de la CEPAL para administración de bibliotecas.
7. El sistema será capaz generar códigos de barras, para poder introducir en un futuro el modulo de lector láser.

⁶ Para mayor referencia ver anexos A

V LIMITACIONES

- El sistema a desarrollar estará limitado por los requisitos establecidos por las autoridades competentes de la Casa de la Cultura de Soyapango, por lo que presentara algunas diferencias al compararlo con otros tipos de software dedicados a la administración de bibliotecas.
- La base de datos no debe involucrar ningún gasto para la institución que implemente el software, por lo tanto la administración de esta debe ser en un sistema de software libre.
- El lenguaje de programación seleccionado debe ser un lenguaje WEB, ya que la aplicación podrá ser utilizada a distancia, por medio de Internet.
- Las reservas de libros en el sistema serán solamente para personas afiliadas a la institución.
- El software podrá ser accedido vía intranet, si la institución lo requiere, logra obtener un hosting y luego montar la página en dicho hosting, eso dependerá de los recursos económicos con que se cuentan en dicha institución.

V.I DELIMITACIÓN

Se ha delimitado el sistema de la siguiente manera: Desarrollo de sistema bibliotecario estándar para la administración básica de Bibliotecas, tomando como base el sistema de codificación de la CEPAL caso práctico para la Casa de la Cultura, bajo plataforma de código libre.

El tiempo que tomara desarrollar el sistema será de 9 meses. Primero se efectuará un análisis preliminar de la forma en como se lleva acabo el proceso de administración de la biblioteca, a partir de este análisis se realizara el diseño del sistema y posteriormente el desarrollo del software que mejorará todos los procesos que se realizan actualmente culminando con la instalación del software y capacitaciones.

El sistema será capaz de las siguientes funciones:

- Realizar los mantenimientos para:
 - Nuevos usuarios
 - Tipos de usuarios
 - Nuevo material a la biblioteca.

- Generar reportes:
 - Estados de Usuarios
 - Estado de Libros
 - Control estadístico de acceso al material.
 - Búsqueda de Material
 - Reserva
 - Control de deudores

Modulo seguridad.

El modulo de seguridad tendrá una apartado que le permitirá controlar todas las aplicaciones del sistema, de esta forma decidiendo los diferentes tipos de acceso que tendrán los usuario y pudiendo manejar nuevos tipos de usuarios si así lo consideran necesario.

Se maneja un modulo de cifrado de contraseñas, para que no puedan ser conocidas por nadie y de esta forma tener un mayor control de auditoria sobre la aplicación.

Modulo de reportes.

El modulo de reportes tendrá la capacidad de generar reportes estadísticos necesarios para poder tener un control de todas las actividades realizadas en la biblioteca, entre los que se consideraran son los siguientes:

- Libros más usados.
- Cantidad de usuarios.
- Ingresos.
- Gastos.

Modulo biblioteca

Este modulo incluirá las siguientes funciones que realizara la biblioteca:

- Búsqueda de Material
- Reserva
- Control de deudores
- Ubicación de libros
- Material en línea. (Biblioteca virtual)

Modulo administración

En este modulo se deberá permitir la administración de usuario y de materiales, por lo que contara con las siguientes funciones:

- Estados de Usuarios
- Estado de Libro
- Permisos
- Mantenimientos de materiales
- Mantenimientos de Usuarios

Modulo institucional

En este modulo el usuario podrá conocer un poco mas de la institución y poder contactar con los administradores y dejar sugerencias las cuales ayuden a mejorar la institución.

- Mantenimiento de sugerencias
- Información institucional

VI Proyección Social

Este Sistema beneficiará en primer lugar a los estudiantes y personas con la necesidad de consultar material bibliográfico que posea dicha biblioteca, ya que contarán con una herramienta que les facilitara el uso de los recursos bibliográficos por ser un sistema amigable y de fácil manejo.

Para el administrador de la biblioteca el software se convertirá en un buen recurso para el control adecuado de la información, ya que tendrá el control total de los registros de manera fácil y poder hacer modificaciones eficientes de las diferentes publicaciones que se manejan.

Este proyecto representaría un primer paso para la digitalización de la información de los diversos recursos bibliográficos con que cuentan, teniendo la posibilidad de crear una herramienta con mayor funcionalidad, es decir, sentaría las bases para que futuros profesionales desarrollen un software que permita acceder a información completa de tomos de libros y tesis completamente digital, con lo cual se crearía un nuevo concepto de Biblioteca Virtual.

El sistema al ser multiempresa, brindara una posibilidad real de ser utilizado por instituciones que no cuenten con los fondos suficientes, para comprar un sistema informático, ya que el proyecto se desarrollara bajo plataformas de software libre no conllevara ningún costo para las instituciones que lo utilicen.

Todas aquellas personas que puedan acceder de una u otra forma al Internet, se miraran favorecidas al poder tener la información de primera mano, y poder saber con que material cuenta la biblioteca si en un futuro la institución quiere montarla en la WEB

CAPITULO I
MARCO REFERENCIAL

1.1 MARCO REFERENCIAL

1.1.1 REFERENCIA HISTORICO

Origen De Las Bibliotecas

La palabra biblioteca tiene origen del griego βιβλιοθήκη *biblion* = libro y *theke* = caja, puede entenderse como una gran caja donde guardamos los libros, pero esta claro que la definición a evolucionado desde que fue sugerida en los pueblos mesopotámicos⁷.

Desde entonces existía una necesidad, por tener un control sobre toda la información a la que se tenía acceso, Los libros eran de carácter religioso en la mayoría de los casos, aunque también se generaban libros sobre administración, política y economía.

Los documentos se escribían en escritura cuneiforme⁸ en tabletas de barro y estas eran archivadas, las cuatro más grandes bibliotecas de este tipo fueron:

Biblioteca de Mari ubicada en la actual Siria.

Biblioteca de Lagash ubicada en una ciudad antigua de Sumeria.

Biblioteca de Ebla ubicada en la actual Siria.

Biblioteca del rey asirio Assurbanipal⁹ ubicada en el suroeste asiático.

En el antiguo Egipto, existieron dos clases de instituciones con similares características a las de una biblioteca, las casas de los libros y las casas de la vida.

⁷ Se llaman pueblos mesopotámicos los que vivieron en el dilatado valle ubicado entre los ríos Éufrates y Tigris, que desembocan luego en el Golfo Pérsico, en la península Arábiga (aproximadamente en el año 4000 a. C).

⁸ La **escritura cuneiforme** está aceptada comúnmente como la forma más temprana conocida de expresión escrita de la que se han encontrado restos arqueológicos

⁹ fue el último gran rey de Asiria. Reinó entre el 669 aC y el 627 aC. Famoso por ser uno de los pocos reyes de la antigüedad que sabía leer y escribir.

Las primeras funcionaban como bodegas donde se llevaba un archivo de todos los libros de la época.

Las casas de la vida eran centro de estudio para los escribas¹⁰, en donde estos podían hacer copias en papiros¹¹ de los libros.

Hasta ahora hemos hablado de bibliotecas de carácter muy religioso, que si bien ya eran consideradas como tales, estas estaban al alcance de muy pocos, la idea más general de lo que simbolizan las bibliotecas, nace en Grecia con el surgimiento de una de las más importantes de todos los tiempos, la biblioteca alejandrina. Que llegó albergar alrededor de 700,000 volúmenes, la destrucción de esta es uno de los más grandes misterios de la civilización occidental.

La biblioteca contaba con diez salones, cada uno dedicado a un área diferente del saber, contenía laboratorios y esta estaba al alcance de más personas, no sólo religiosos.

Con todo lo anterior era evidente que la información era demandada cada vez más por muchas más personas, lo que llevaría con el paso del tiempo al nacimiento de las primeras bibliotecas públicas, estas tienen su origen en roma y fueron llamadas como Octaviana y Palatina¹².

Las bibliotecas en la edad Media (SIGLO V-XV DC)

Con la caída del imperio romano en el año 476, todos los avances que habían existido a la fecha en el alcance de la información para más personas, sufre un retroceso fatal, y otra vez las bibliotecas quedan en las manos de los monasterios lejos del alcance popular, pero sin duda existieron grandes bibliotecas de monasterios, entre las que podemos mencionar las siguientes:

¹⁰ La palabra española **escriba** procede del latín: *scriba*. En hebreo *so-fér*, procede de una raíz que significa “contar”, y se traduce “secretario”, “escribano”, “copista”; y la palabra griega *gram·ma·téus* se traduce “escriba”, “instructor público”; el término alude a una persona instruida.

¹¹ **Papiro** es el nombre que recibe el soporte de escritura elaborado a partir de una planta acuática

¹² Bajo el reinado del emperador Augusto se fundan las grandes bibliotecas Palatina y Octaviana

1. Saint Gall ubicada en Suiza
2. Fulda ubicada en Alemania
3. Reichenau ubicada en Alemania

El mundo árabe también creó bibliotecas en sus mezquitas y centros de enseñanza coránicas, entre las más grandes de esta clase podemos mencionar:

La del califa¹³ Al-Mamun en Bagdad o Abd-al-Rahman III y su hijo Al Hakam II en Córdoba.

Las bibliotecas en la edad moderna

Los chinos fueron los primeros en trabajar en sistemas de imprenta en el siglo X, pero sin duda se considera el nacimiento de la imprenta hasta el siglo XV, debido a que aunque fuese inventada antes, el problema en sí, no era la imprenta, era la existencia de un material barato en el cual, se pudieran reproducir escritos y es justo hasta el siglo XV que con el apareamiento de la industria del papel esto se vuelve una realidad.

La edad moderna ve renacer las bibliotecas públicas, ya que los reinados abren las puertas de sus centros de información, para que todo aquel con conocimientos de lectura pueda beneficiarse de toda la información que estas bibliotecas contenían.

Nacen grandes bibliotecas en el mundo, entre las que podemos mencionar las siguientes:

¹³ del árabe خليفة *jalifa*, "representante".

- La Ambrosiana en Milán
- La Mazarina en Paris
- La del Museo Británico
- Las bibliotecas de Yale, Harvard y Princeton

Las bibliotecas en la edad contemporánea

Con la revolución francesa (entre 1789 y 1799) y la revolución americana (entre 1775 y 1783) nuevas tendencias en el mundo aparecerían, y la idea de que todos tenían acceso a la información, cada vez se convertiría en una realidad mucho más probable, millones de personas lucharían por ideales de libertad, igualdad y respeto. Fue así como a partir de esta serie de revoluciones, se generan nuevas opciones para las personas y las bibliotecas públicas al fin se instalan en todo el mundo.

Durante el siglo XX se generan tres corrientes en el mundo bibliotecarios que darán pie a fundamentar estas:

1. La concepción europea continental

Esta tendencia separaría dos clases de biblioteca, las bibliotecas de investigación y las bibliotecas públicas, las bibliotecas de investigación serían las bases de laboratorios de desarrollo de nuevas tecnologías. Las bibliotecas públicas serían de carácter netamente educativo, es decir estas no en todos los casos brindarían la posibilidad de utilizar los laboratorios pero si; brindarían una opción real a personas de todas clases sociales de poder desarrollar sus conocimientos, a raíz de esto, en el siglo XX se desarrollan una serie de inventos importantes para el mundo actual.

2. La concepción anglosajona

Esta corriente se diferenciaba de la anterior, ya que esta hace referencia a la idea de que todas las personas tenemos acceso a la información y a la misma clase de oportunidades, por lo que diverge de la anterior en el apartado de que en esta corriente, se habla de que si existen laboratorios para algunos, todos

deberían tener oportunidad de desarrollar en estos siempre y cuando sean capaces.

3. La concepción socialista

Si bien esta corriente esta a favor de que la información deba llegar a todos, la diferencia con esta corriente es que habla del tipo información que pueden recibir las personas, durante el tiempo del apogeo del socialismo en el mundo, las bibliotecas controladas por este régimen, utilizaron las bibliotecas como medio de propaganda ideológico de sus ideales.

Origen de las Bibliotecas en El Salvador

La Biblioteca de la UES surge a los pocos años que la Asamblea Nacional Legislativa decretara el establecimiento del colegio La Asunción y la Universidad Nacional, en 1841. Se estima que la formación inicial de la biblioteca se realizó entre los años 1848 y 1854, gracias a materiales donados y a compras de libros de medicina, filosofía, gramática, derecho, y literatura. La naciente Biblioteca apoyaba a las 4 Facultades originales (Jurisprudencia y Ciencias Sociales, Medicina, Farmacia, y Odontología), que fueran creadas con el Decreto de Fundación de la Universidad de 1841.

1.1.2 REFERENCIA CONCEPTUAL

Bibliotecas Virtuales

En los primeros tiempos de la computación [cliente-servidor](#), cada aplicación tenía su propio programa cliente y su interfaz de usuario, estos tenían que ser instalados separadamente en cada estación de trabajo. Una mejora al servidor, como parte de la aplicación, requería típicamente una mejora de los clientes instalados en cada una de las estaciones de trabajo, añadiendo un costo de soporte técnico y disminuyendo la eficiencia del personal.

En contraste, las aplicaciones Web generan dinámicamente una serie de páginas en un formato estándar, soportado por navegadores Web comunes como [HTML](#) o [XHTML](#). Se utilizan [lenguajes interpretados](#) del lado del cliente, tales como [JavaScript](#), para añadir elementos dinámicos a la interfaz de usuario. Generalmente cada página Web individual es enviada al cliente como un documento estático, pero la secuencia de páginas provee de una experiencia interactiva.

Las bases de las bibliotecas virtuales nacieron de la mano del protocolo FTP. Cuando en 1969 como parte de un proyecto del gobierno de los Estados Unidos, se desarrollo el protocolo, que permitía la transferencia de archivos en redes de computadoras.

Pero fue hasta la implementación de sistemas de información como el caso de Gopher, que se comenzaron a observar lo que seria los pilares fundamentales de lo que hoy son las bibliotecas virtuales, estos programas permitirían a las personas poder almacenar de forma amigable la información para los usuarios finales, y así comenzar archivar un mundo entero de información.

Lo que comenzó como una transmisión de archivos en código de bajo nivel, se desarrollo enormemente en la década de los ochenta y noventa surgiendo diferentes lenguajes de programación Web, en la actualidad uno de los lenguajes mas populares para la creación de paginas Web, es el lenguaje PHP. Este lenguaje representa una herramienta de alto nivel, para los programadores por la forma en que interactúa con diferentes tecnologías.

La creación y administración de una biblioteca virtual contiene las siguientes partes:

- Jefe de proyecto
- Coordinador de proyecto
- Administrador de base de datos
- Diseñador gráfico
- Programador
- Administrador del sitio

1.1.3 REFERENCIA EXPERIMENTAL

Alrededor del mundo existen muchos sistemas de biblioteca, con diferencias marcadas entre ellos, existen bibliotecas en línea que ofrecen la totalidad de la información de los libros que ellos manejan, como es el caso de wikipedia, que empezó a partir del año 2001 y ha tenido gran difusión en el mundo entero.

Las bibliotecas de consulta, son otro tipo de biblioteca Web ya que estas tienen la característica especial que al buscar un libro en especial, no encontraremos toda la información del libro en la mayoría de los casos; pero si un fragmento considerable que invita al lector a buscar el libro completo y poder dar solución a todas sus dudas.

El ejemplo de mayor difusión de este tipo es “google books” en el cual se pueden encontrar los libros de forma inmediata con colaboraciones de usuarios de todas partes del mundo.

Las bibliotecas de clientes directos son el tipo de bibliotecas mas utilizado por las instituciones educativas del mundo, este tipo de sistema presenta al usuario la opción de buscar libros, a los cuales el usuario tiene acceso directo por medio de la institución que cuenta con ellos.

Sistema bibliotecarios Universidad Don Bosco

Este sistema de biblioteca creado para todos los miembros de la institución salesiana de El Salvador y fue implementado en el 2000, fue creado con el objetivo de que los estudiantes pudieran acceder a la base de datos de los libros con los cuales cuenta, y de esta forma poder hacer reservas de materiales con la seguridad de la existencia de estos.

La aplicación Web, les permite a los clientes poder encontrar libros ya sea desde su casa o en la biblioteca misma y de esta forma tener un control sobre la existencia y la disponibilidad de los libros o materiales bibliotecarios al acceso de los estudiantes.

Existe una variedad importante de bibliotecas en línea para bibliotecarios entre las que podemos mencionar las siguientes:

The Internet Library For Librarians

www.itcompany.com

Este sitio presenta información detallada por profesión, permitiendo a las personas poder encontrar materiales detallados de sus intereses y conocimientos.

The Internet Public Library

www.ipl.org ipl.ub.lu.se

Esta biblioteca virtual contiene recursos como: guías para usar Internet, listas de periódicos on-line de todo el mundo, listas de magazines y journals electrónicos, desarrollo profesional, temas para bibliotecarios y profesionales de la información, biblioteca virtual para niños, material didáctico para niños, calendario de eventos y congresos.

ISI Institute for Scientific Information

www.isinet.com

Esta página ofrece un fragmento de la información de cada libro, y de esta forma indicarle al lector una visión mas acertada de la información que encontrara, invita a buscar el original si lo que ha visto es de del agrado del cliente.

1.2 METODOLOGIA DE LA INVESTIGACION

La metodología contará de dos partes, primero la preparación del análisis y diseño para luego realizar el desarrollo del mismo, a continuación se describen detalladamente los pasos a seguir:

PREPARACIÓN DEL SISTEMA

La preparación del sistema se realizará utilizando diversas herramientas de evaluación de sistemas:

Entrevista:

Se realizará entrevista con los usuarios directos del sistema, entendiendo usuarios directos como aquellos que harán uso de la herramienta para generar diversas consultas frecuentes, esta entrevista se realizaran al alumnado de una institución pública o privadas.

Entrevista con el administrador de biblioteca:

Con el fin de conocer las necesidades reales del administrador y soluciones a los problemas que se están presentando al actual sistema manual de administración de la biblioteca.

Entrevista con el Administrador de Base de datos:

Esta entrevista tiene el fin de encontrar las actuales herramientas informáticas con las que cuenta la institución.

Encuestas:

Las encuestas se realizarán para determinar elementos de prioridad en todos los usuarios que no fueron detectados en las entrevista, así como orientar el sistema a dar soluciones a las demandas comunes de todos los usuarios del sistema.

Evaluación del sistema

Definición de atributos del sistema:

En esta parte se podrá ya definir con que características contara el sistema y los atributos que debe presentar, en esta sección consideraremos ya las necesidades y el diagrama de la base de datos con el fin de crear una base de datos optima y normalizada.

Diagrama de flujos del sistema:

A partir de la preparación del sistema, se generará un estudio concreto del diagrama de flujos del sistema, para tomar en cuenta las necesidades y observar la forma en la que se pretende darle soluciones a cada proceso.

Se pretende luego de realizar el análisis previo del sistema, entrar en los cinco flujos fundamentales de trabajo para el proceso unificado

Requerimientos.

La preparación del sistema ayudara a concretar los requerimientos, ayudando de esta forma a consolidar todos los requerimientos de los diferentes usuarios,

después de haber realizado una investigación profunda de las necesidades individuales.

Análisis.

La evaluación del sistema, generará las herramientas necesarias para poder entrar a fondo en lo que será el análisis en el proceso unificado.

Diseño.

La parte de diseño contará con un modelado de los datos, unificación del diagrama de flujos, generando como pilar fundamental el diseño del sistema.

Creación.

La creación del sistema es de gran importancia en todo proyecto informático, porque es en ésta parte donde se pueden realizar las últimas correcciones con el fin de generar de forma dinámica modificaciones el usuario crea conveniente.

Pruebas.

Las pruebas se realizaran en tres etapas:

Etapas Prueba Programador:

La etapa de prueba del programador consiste en realizar pruebas de forma directa con el código y afinar detalles que se presenten.

Etapa Prueba Administrador.

Esta prueba se realizará en sistema Beta con el fin que el administrador observe y apunte consideraciones al sistema pre-final.

Etapa Prueba Usuarios:

Estas pruebas se realizarán de forma directa con los usuarios y estos podrán descubrir detalles últimos para el funcionamiento correcto del sistema.

CAPITULO II

METODOLOGÍA DE LA PROGRAMACIÓN

2.1 ANALISIS DE RESULTADO Y DIAGNOSTICO

Tipo de Investigación

A continuación se describe el tipo de investigación realizada y los objetivos que se han planteado para esta etapa.

Los objetivos de la investigación son:

- Definir la muestra de la población a la que se estudiara.
- Seleccionar el método y las técnicas adecuadas al objeto de estudio.
- Verificar el uso, frecuencia y tipo de información solicitada en la biblioteca de la casa de la cultura de soyapango
- Verificar la factibilidad de desarrollo del sistema bibliotecario.

La técnica que se utiliza es la de campo la cual es el instrumento de observación diseñado según el objeto de estudio.

Para lograr obtener un resultado de la investigación que favorezca a la interpretación y solución del problema se realizan las siguientes actividades:

- Verificar la utilización adecuada de las instalaciones y el sistema bibliotecario actual por parte de los usuarios.
- Reunir información para interpretar hallazgos.
- Interpretación de la información generada.

Población y Muestra

Como universo, se pretende abarcar todo tipo de usuarios, tanto estudiantes de educación básica, media y universitaria, así como a personas interesadas en la lectura.

De acuerdo a las estadísticas del Ministerio de Educación, el municipio de Soyapango reporto para el 2007 un total de 68,097 alumnos matriculados en educación media, básica y parvularia¹⁴, de dicho dato se tomara como base de muestra el número de visitas anuales que tiene dicha biblioteca, el cual es de 2000 consultas, pero como las visitas son muy esporádicas y dependen mucho de las épocas del año, se tomara un total de 500 personas como un global, el cual es el correspondiente a un trimestre de consultas.

Para conocer exactamente cual debe ser el espacio muestral para desarrollar o implementar nuestro estudio, se utilizó la siguiente fórmula:

$$n = \frac{Z^2 * N * p * q}{I^2 (N - 1) + Z^2 * p * q}$$

En donde:

- N: Tamaño de la población.
- Z: valor correspondiente a la distribución de Gauss 1.96 para probabilidad de 0.5.
- p: probabilidad de éxito esperada. En caso de desconocerse, aplicar la opción mas desfavorable (p=0.5), que hace mayor el tamaño muestral.
- q: probabilidad de fracaso 1-p.
- I: Error que se prevé cometer. Se tiene un error de 10%

¹⁴ Dato tomado deL Ministerio de Educación <http://www.mined.gob.sv/> entre el periodo de 2005 y 2007.

$$n = \frac{1.96^2 * 500 * 0.5 * 0.5}{0.1^2(500 - 1) + 1.96^2 * 0.5 * 0.5}$$

Cuyos valores serían $Z = 1.96$, $p = q = 0.5$, $N = 500$; con lo que obtendríamos un valor de $n = 53$.

Por lo que nuestra muestra es de 53 personas, a los cuales se les pasará la encuesta a fin de recabar información, y así poder obtener datos que nos brinde pautas para poder determinar la importancia de la investigación.

2.2 TÉCNICAS Y HERRAMIENTAS DE INVESTIGACIÓN

La técnica es indispensable en el proceso de la investigación científica, ya que integra la estructura por medio de la cual se organiza la investigación. La técnica pretende los siguientes objetivos:

- Ordenar las etapas de la investigación.
- Aportar instrumentos para manejar la información.
- Llevar un control de los datos.
- Orientar la obtención de conocimientos.

En cuanto a las técnicas de investigación que se utilizarán para el desarrollo del proyecto son dos formas generales: técnica documental y técnica de campo.

La técnica documental permite la recopilación de información que permita tener un panorama de los diferentes sistemas bibliográficos utilizados actualmente en el país y como benefician a sus diferentes usuarios.

La técnica de campo permite la observación en contacto directo con el objeto de estudio, es decir, estudiantes y personas particulares que hacen uso de la biblioteca de la casa de la cultura, y el acopio de testimonios que permitan confrontar la teoría con la práctica en la búsqueda de la verdad objetiva.

Técnica documental

El objetivo de la investigación documental es elaborar un marco teórico conceptual para formar un cuerpo de ideas sobre el objeto de estudio, se consolidará toda la información que sea relevante para el desarrollo del editor.

Fuentes primarias de información

Estas fuentes son los documentos que registran o corroboran el conocimiento inmediato de la investigación incluyen libros, revistas, informes técnicos y tesis.

Técnica de campo

El instrumento de observación se diseña según el objeto de estudio.

Objetivos de la observación

- Explorar. Precisar aspectos previos a la observación estructurada y sistemática.
- Reunir información para interpretar hallazgos.
- Describir hechos.

Al llevar a cabo la investigación se contemplará algunos requisitos necesarios para que el resultado sea favorable, entre estos están:

- Delimitar los objetivos de la observación.
- Especificar el procedimiento o instrumentos de observación.
- Comprobación continua.

La observación sistemática se realiza de acuerdo con un plan de observación preciso, en el que se han establecido variables y su relación, objetivos y procedimientos de observación.

Para la *observación sistemática*, el instrumento que se uso es la Entrevista.

Entrevista. La encuesta es una pesquisa o averiguación en la que se emplean cuestionarios para conocer la opinión pública. Consiste en el acopio de testimonios orales y escritos de personas vivas. En la investigación de campo, para la recopilación de información pueden utilizarse las entrevistas, los cuestionarios y el muestreo, entre otros. La entrevista es una de las técnicas más usuales en ciencias sociales. Puede definirse como la relación que se establece entre el investigador y los sujetos de estudio. Puede ser individual o grupal, libre o dirigida.

Objetivos de la entrevista:

1. Obtener información sobre el objeto de estudio.
2. Describir con objetividad situaciones o fenómenos.
3. Interpretar hallazgos.
4. Plantear soluciones.

2.2.1 Presentación de los resultados (Encuesta)

Objetivo General

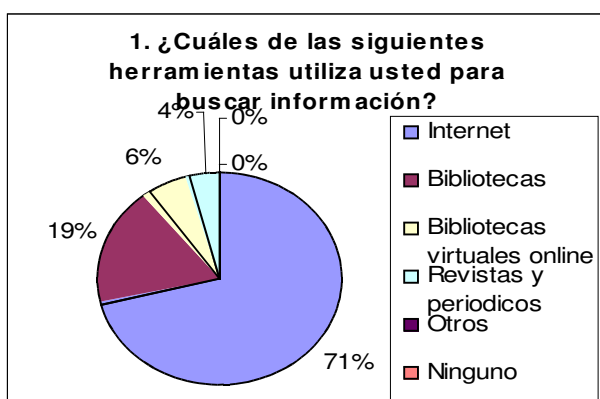
Identificar el uso de la biblioteca de la casa de la cultura de soyapango por parte de los estudiantes y personas particulares y la distribución de dichas instalaciones

Resultados de la encuesta

Pregunta 1.

¿Cuáles de las siguientes herramientas utiliza usted para buscar información?

Objetivo: Indagar sobre que tipos de medios utiliza para obtener la información que busca



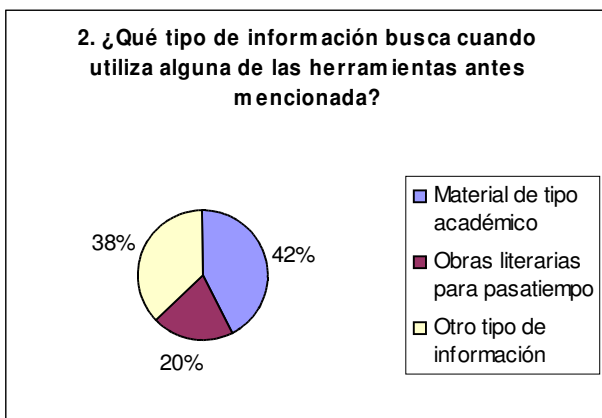
Conclusiones.

- Es notorio el gran auge de la tecnología en los centros educativos y en la población en general, por esta razón la mayoría de estudiantes prefieren buscar en Internet cualquier tipo de información que buscan
- Aunque el uso de Internet poco a poco está reemplazando a las bibliotecas se mantiene cierto porcentaje de personas que aun las visita.

Pregunta 2.

¿Qué tipo de información busca cuando utiliza alguna de las herramientas antes mencionada?

Objetivo: Identificar el tipo de información que buscan cuando hacen uso de la biblioteca de la casa de la cultura.



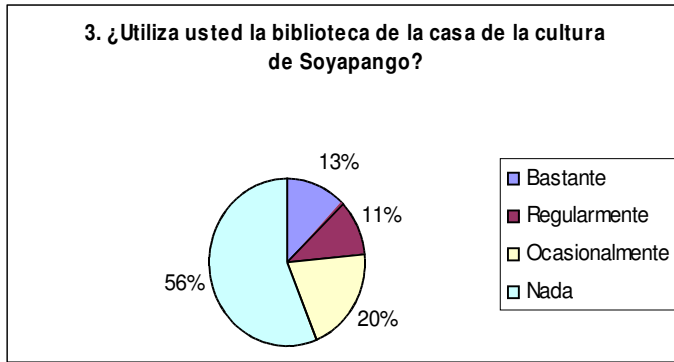
Conclusiones.

- Se identifica que la mayoría de usuarios buscan información de apoyo para tareas y material extra para diversión o pasatiempo en su mayoría, lo que permite en ocasiones el abuso del acceso a la información sobre todo pornografía en Internet

Pregunta 3.

¿Utiliza usted la biblioteca de la casa de la cultura de Soyapango?

Objetivo: Conocer que tanto utilizan la biblioteca de la casa de la cultura de soyapango.



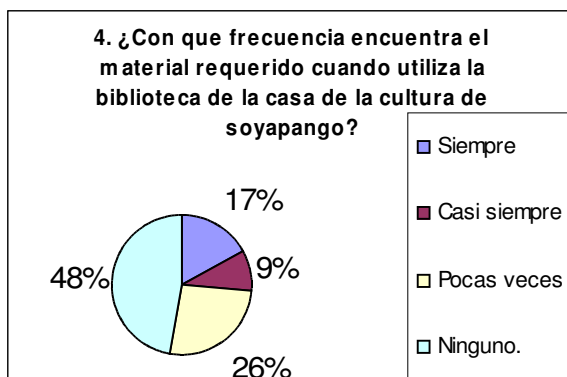
Conclusiones.

- Se observa que más de la mitad de los encuestados no visitan dicha biblioteca, ya sea por falta de material o por tener otro tipo de herramientas para suplir dicha necesidad.
- Las personas que visitan bastante la biblioteca son aquellas que asisten a cursos de música, pintura, teatro, etc.

Pregunta 4.

¿Con que frecuencia encuentra el material requerido cuando utiliza la biblioteca de la casa de la cultura de soyapango?

Objetivo: Conocer si los usuarios encuentran toda la información que buscan en la biblioteca.



Conclusiones.

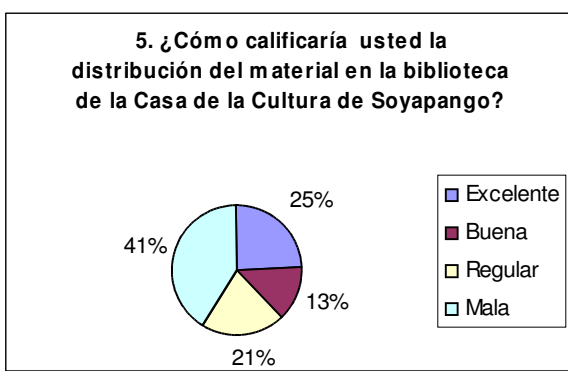
- Debido a la poca existencia de material educativo, es notorio la cantidad de personas que no encuentran dichos textos, por lo que optan por buscar en otro tipo de medio

- Las personas que si encuentran la información requerida, es del tipo de libros de textos para educación básica.

Pregunta 5.

¿Cómo calificaría usted la distribución del material en la biblioteca de la Casa de la Cultura de Soyapango?

Objetivo: Investigar la correcta distribución del espacio utilizado por la biblioteca de la casa de la cultura de Soyapango



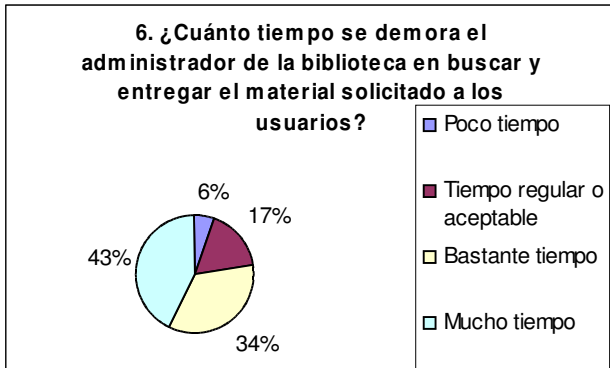
Conclusiones.

- Debido al espacio reducido el cual es destinado para la Casa de la Cultura de Soyapango se nota que los usuarios no se sienten cómodos en dichas instalación y no existe una correcta ubicación de los libros.
- Al no existir una optimización adecuada del lugar se inician los problemas de retardo en tiempos para la realización de consultas.

Pregunta 6.

¿Cuánto tiempo se demora el administrador de la biblioteca en buscar y entregar el material solicitado a los usuarios?

Objetivo: Indagar sobre el tiempo que tardan los administradores de la biblioteca para realizar la búsqueda del material solicitado y entregarlo al usuario



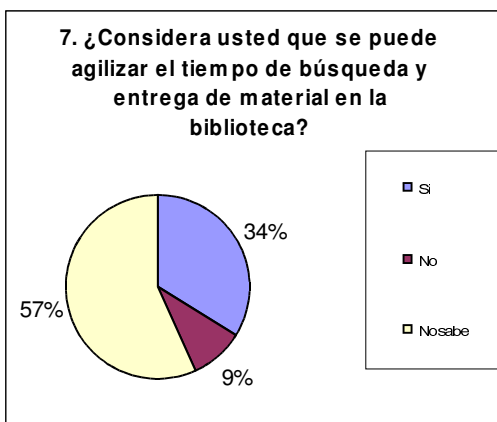
Conclusiones.

- Existe mucha inconformidad con el tiempo que se demoran los administradores de la biblioteca para realizar una consulta y entregar el material solicitado.

Pregunta 7.

¿Considera usted que se puede agilizar el tiempo de búsqueda y entrega de material en la biblioteca?

Objetivo: Comprobar si los usuarios desearían que se agilizaran dichas búsquedas o si están conformes de cómo se realiza actualmente dichos procesos



Conclusiones.

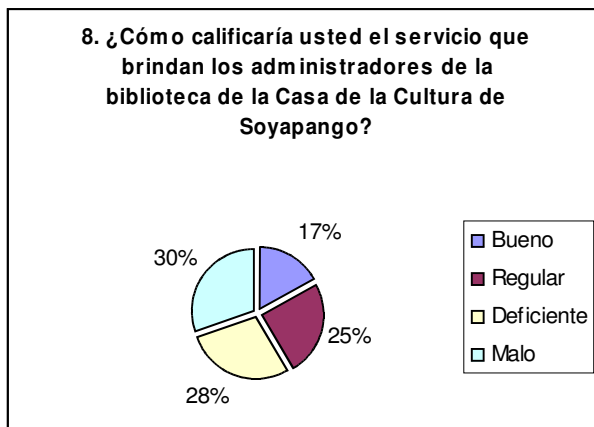
- Debido al tipo de población tomada, muy pocos saben de optimización de tiempo, espacio y ubicación por lo que no dieron su opinión al respecto.

- Este es un buen parámetro a tomar en cuenta en la construcción de un software para la administración de la biblioteca, de forma tal que pueda agilizar los tiempos de consultas.

Pregunta 8.

¿Cómo calificaría usted el servicio que brindan los administradores de la biblioteca de la Casa de la Cultura de Soyapango?

Objetivo: Conocer la calidad de servicio que se esta brindando a los usuarios por parte de los administradores de la biblioteca



Conclusiones.

- Es notorio un equilibrada aceptación del servicio que esta brindando dicha biblioteca a pesar de los limitados recursos con que cuenta actualmente, se espera mejorar la aceptación para tener un mayor auge de visitantes en un futuro

2.3 EDITOR DE DESARROLLO

El editor seleccionado para desarrollar el sistema es NetBeans, Es mejor conocido como un ambiente integrado de desarrollo de aplicaciones en JAVA, que cuenta con arquitectura modular y extensible marco de trabajo.

Plataforma Netbeans.

Se le conoce como Plataforma NetBeans al marco de trabajo (framework) para aplicaciones de cliente rico (rich client)¹⁵, con el propósito de crear software que se *escriba una vez y funcione en cualquier sistema operativo*. Se le conoce como aplicaciones de cliente rico (rich client application) a una pieza de software donde una buena porción de las características del sistema trabaja en el sistema local del usuario. Es el término de NetBeans como sinónimo de aplicaciones de escritorio.

Entre los beneficios que provee Netbeans en su plataforma están:

- NetBeans es gratis y su código es libre de reutilizar.
- NetBeans es framework maduro con gran cantidad de características para el desarrollo de aplicaciones.
- NetBeans se basa en el pensamiento *“escrito una vez, funciona donde sea”*, lo cual permite desarrollar aplicaciones independiente del sistema operativo del usuario.
- NetBeans es una tecnología basada en estándares y fuente abierta (open source).
- NetBeans cuenta con una gran comunidad de desarrolladores.

Para el desarrollo de aplicaciones NetBeans provee las siguientes características:

- Un sistema Windows que simplifica la manipulación de múltiples componentes en una misma área de trabajo.

¹⁵ Se denomina **cliente rico** al programa "cliente" de una arquitectura cliente-servidor cuando la mayor carga de cómputo está desplazada hacia la computadora que ejecuta dicho programa.

- Un sistema Action que facilita la instalación y desinstalación de menús, barra de herramientas, etc.
- Un mecanismo de actualización dinámica de los elementos de las aplicaciones.
- Una arquitectura extensible para las aplicaciones al usar técnicas de programación modular.

La arquitectura modular de netbeans convierte a las aplicaciones en un conjunto de pequeño, separados y aislados módulos, conteniendo su respectiva funcionalidad.

Estos pueden ser de tres tipos:

- Interfase de Usuarios. (End-Users Interface).
- Librería Simple.
- Modulo de Librerías.

Este sistema de módulos es un contenedor en tiempo de ejecución que asegura la integridad de estas piezas. La utilización de módulos lo realiza por medio de las llamadas dependencias de modulo. El propósito de la modularidad es hacer que la aplicación se convierta en interacción entre sistemas, en vez de partes de subsistemas.

NetBeans cuenta con una interfase GUI, como componente de Windows System, la cual proporciona al desarrollador todo un conjunto de herramientas para el desarrollo de aplicaciones de escritorio utilizando la librería Swing.

2.4 SOFTWARE WEB

Características De Java

Las características principales que ofrece Java respecto a cualquier otro lenguaje de programación, son:

Simple

Java ofrece toda la funcionalidad de un lenguaje potente, pero sin las características menos usadas y más confusas de éstos. C++ es un lenguaje que adolece de falta de seguridad, pero C y C++ son lenguajes más difundidos, por ello Java se diseñó para ser parecido a C++ y así facilitar un rápido y fácil aprendizaje.

Java elimina muchas de las características de otros lenguajes como C++, para mantener reducidas las especificaciones del lenguaje y añadir características muy útiles como el garbage collector (reciclador de memoria dinámica). No es necesario preocuparse de liberar memoria, el reciclador se encarga de ello y como es un thread de baja prioridad, cuando entra en acción, permite liberar bloques de memoria muy grandes, lo que reduce la fragmentación de la memoria.

Java reduce en un 50% los errores más comunes de programación con lenguajes como C y C++ al eliminar muchas de las características de éstos, entre las que destacan:

- Aritmética de punteros
- No existen referencias
- Registros (struct)
- Definición de tipos (typedef)
- Macros (#define)
- Necesidad de liberar memoria (free)

Distribuido

Java se ha construido con extensas capacidades de interconexión TCP/IP. Existen librerías de rutinas para acceder e interactuar con protocolos como http y ftp. Esto permite a los programadores acceder a la información a través de la red con tanta facilidad como a los ficheros locales.

Robusto

Java realiza verificaciones en busca de problemas tanto en tiempo de compilación como en tiempo de ejecución. La comprobación de tipos en Java ayuda a detectar errores, lo antes posible, en el ciclo de desarrollo. Java obliga a la declaración explícita de métodos, reduciendo así las posibilidades de error. Maneja la memoria para eliminar las preocupaciones por parte del programador de la liberación o corrupción de memoria.

También implementa los arrays auténticos, en vez de listas enlazadas de punteros, con comprobación de límites, para evitar la posibilidad de sobrescribir o corromper memoria resultado de punteros que señalan a zonas equivocadas. Estas características reducen drásticamente el tiempo de desarrollo de aplicaciones en Java.

Java proporciona:

- Comprobación de punteros
- Comprobación de límites de arrays
- Excepciones
- Verificación de byte-codes

Arquitectura neutral

Para establecer Java como parte integral de la red, el compilador Java compila su código a un fichero objeto de formato independiente de la arquitectura de la máquina en que se ejecutará. Cualquier máquina que tenga el sistema de

ejecución (run-time) puede ejecutar ese código objeto, sin importar en modo alguno la máquina en que ha sido generado.

El código fuente Java se "compila" a un código de bytes de alto nivel independiente de la máquina. Este código (byte-codes) está diseñado para ejecutarse en una máquina hipotética que es implementada por un sistema run-time, que sí es dependiente de la máquina.

En una representación en que tuviésemos que indicar todos los elementos que forman parte de la arquitectura de Java sobre una plataforma genérica, obtendríamos una figura como la siguiente:

Seguro

La seguridad en Java tiene dos facetas. En el lenguaje, características como los punteros o el casting implícito que hacen los compiladores de C y C++ se eliminan para prevenir el acceso ilegal a la memoria. Cuando se usa Java para crear un navegador, se combinan las características del lenguaje con protecciones de sentido común aplicadas al propio navegador.

Portable

Más allá de la portabilidad básica por ser de arquitectura independiente, Java implementa otros estándares de portabilidad para facilitar el desarrollo. Los enteros son siempre enteros y además, enteros de 32 bits en complemento a 2. Además, Java construye sus interfaces de usuario a través de un sistema abstracto de ventanas de forma que las ventanas puedan ser implantadas en entornos Unix, Pc o Mac.

Interpretado

El intérprete Java (sistema run-time) puede ejecutar directamente el código objeto. Enlazar (linkar) un programa, normalmente, consume menos recursos que compilarlo, por lo que los desarrolladores con Java pasarán más tiempo desarrollando y menos esperando por el ordenador. No obstante, el compilador

actual del JDK es bastante lento. Por ahora, que todavía no hay compiladores específicos de Java para las diversas plataformas, Java es más lento que otros lenguajes de programación, como C++, ya que debe ser interpretado y no ejecutado como sucede en cualquier programa tradicional.

Tecnología Struts

Es un framework que implementa el patrón de arquitectura MVC (Model-View-Controller) en Java. Un framework es la extensión de un lenguaje mediante una o más jerarquías de clases que implementan una funcionalidad y que (opcionalmente) pueden ser extendidas. El framework puede involucrar TagLibraries.

Struts se desarrollaba como parte del proyecto Jakarta de la Apache Software Foundation, pero actualmente es un proyecto independiente conocido como Apache Struts.

Struts permite reducir el tiempo de desarrollo. Su carácter de "*software libre*" y su compatibilidad con todas las plataformas en que Java Enterprise esté disponible, lo convierte en una herramienta altamente disponible.

Funcionamiento de Struts

Struts se basa en el patrón del Modelo-Vista-Controlador (MVC) el cual se utiliza ampliamente y es considerado de gran solidez. De acuerdo con este patrón, el procesamiento se separa en tres secciones diferenciadas, llamadas el modelo, las vistas y el controlador.

Cuando se programan aplicaciones Web con el patrón MVC, siempre surge la duda de usar un solo controlador o usar varios controladores, pues si se considera mejor usar un solo controlador para tener toda la lógica en un mismo lugar, se

encuentra con un grave problema, ya que el controlador se convierte en lo que se conoce como "fat controller", es decir un controlador de peticiones, Struts surge como la solución a este problema ya que implementa un solo controlador (ActionServlet) que evalúa las peticiones del usuario mediante un archivo configurable (struts-config.xml).

Componentes del modelo

Corresponden a la lógica del negocio con la cual se comunica la aplicación Web. Usualmente el modelo comprende accesos a Bases de Datos o sistemas que funcionan independientemente de la aplicación Web.

Componentes del control

Los componentes de control son los encargados de coordinar las actividades de la aplicación, que van desde la recepción de datos del usuario, las verificaciones de forma y la selección de un componente del modelo a ser llamado. Por su parte los componentes del modelo envían al control sus eventuales resultados o errores de manera de poder continuar con otros pasos de la aplicación.

Esta separación simplifica enormemente la escritura tanto de vistas como de componentes del modelo: Las páginas JSP no tienen que incluir manejo de errores, mientras que los elementos del control simplemente deciden sobre el paso siguiente.

Entre las características de Struts se pueden mencionar:

- Configuración del control centralizada.
- Interrelaciones entre acciones y página u otras acciones se especifican por tablas XML en lugar de codificarlas en los programas o páginas.
- Componentes de aplicación, que son el mecanismo para compartir información bidireccionalmente entre el usuario de la aplicación y las acciones del modelo.

- Librerías de entidades para facilitar la mayoría de las operaciones que generalmente realizan las páginas JSP.
- Struts contiene herramientas para validación de campos de plantillas bajo varios esquemas que van desde validaciones locales en la página (en javaScript) hasta las validaciones de fondo hechas a nivel de las acciones.

Struts permite que el desarrollador se concentre en el diseño de aplicaciones complejas como una serie simple de componentes del Modelo y de la vista intercomunicados por un control centralizado. Diseñando de esta manera puede obtenerse una aplicación más consistente y más fácil de mantener.

2.5 MANEJADOR DE BASE DE DATOS

Se ha seleccionado MySQL como manejador de base de datos por su gran fortaleza, robustez, por su alta fiabilidad dentro del mercado comercial, por ser gratis, con lo que se exonera a la institución de pagar por una licencia de este tipo; por lo que se detalla ciertos aspectos técnicos e históricos de dicho manejador.

MySQL es un sistema de gestión de base de datos relacional, multihilo y multiusuario con más de seis millones de instalaciones. MySQL AB —desde enero de 2008 una subsidiaria de Sun Microsystems— desarrolla MySQL como software libre en un esquema de licenciamiento dual.

Por un lado se ofrece bajo la GNU GPL para cualquier uso compatible con esta licencia, pero las empresas que quieran incorporarlo en productos privativos pueden comprar a la empresa una licencia específica que les permita este uso. Está desarrollado en su mayor parte en ANSI C.

Al contrario que proyectos como Apache, donde el software es desarrollado por una comunidad pública y el copyright del código está en poder del autor individual, MySQL es propiedad y está patrocinado por una empresa privada, que posee el copyright de la mayor parte del código.

Esto es lo que posibilita el esquema de licenciamiento anteriormente mencionado. Además de la venta de licencias privativas, la compañía ofrece soporte y servicios. Para sus operaciones contratan trabajadores alrededor del mundo que colaboran vía Internet.

Historia del proyecto

SQL (*Lenguaje de Consulta Estructurado*) fue comercializado por primera vez en 1981 por IBM, el cual fue presentado a ANSI y desde ese entonces ha sido considerado como un estándar para las bases de datos relacionales. Desde 1986, el estándar SQL ha aparecido en diferentes versiones como por ejemplo: SQL:92, SQL:99, SQL:2003. MySQL es una idea originaria de la empresa opensource MySQL AB establecida inicialmente en Suecia en 1995 y cuyos fundadores son David Axmark, Allan Larsson, y Michael "Monty" Widenius. El objetivo que persigue esta empresa consiste en que **MySQL** cumpla el estándar SQL, pero sin sacrificar velocidad, fiabilidad o usabilidad.

Michael Widenius en la década de los 90 trató de usar *mSQL* para conectar las tablas usando rutinas de bajo nivel ISAM, sin embargo, mSQL no era rápido y flexible para sus necesidades. Esto lo conllevó a crear una API SQL denominada **MySQL** para bases de datos muy similar a la de *mSQL* pero más portable.

La procedencia del nombre de MySQL no es clara. Desde hace más de 10 años, las herramientas han mantenido el prefijo My. También, se cree que tiene relación con el nombre de la hija del cofundador Monty Widenius quien se llama *My*.

Lenguajes de programación

Existen varias APIs que permiten, a aplicaciones escritas en diversos lenguajes de programación, acceder a las bases de datos MySQL, incluyendo C, C++, C#, Pascal, Delphi (via dbExpress), Eiffel, Smalltalk, Java (con una implementación nativa del driver de Java), Lisp, Perl, PHP, Python, Ruby, Gambas, REALbasic (Mac), FreeBASIC, y Tcl; cada uno de estos utiliza una API específica. También existe un interfaz ODBC, llamado MyODBC que permite a cualquier lenguaje de programación que soporte ODBC comunicarse con las bases de datos MySQL. También se puede acceder desde el sistema SAP, lenguaje ABAP.

Aplicaciones

MySQL es muy utilizado en aplicaciones Web como, Drupal o phpBB, en plataformas (Linux/Windows-Apache-MySQL-PHP/Perl/Python), y por herramientas de seguimiento de errores como Bugzilla. Su popularidad como aplicación Web está muy ligada a PHP, que a menudo aparece en combinación con MySQL. MySQL es una base de datos muy rápida en la lectura cuando utiliza el motor no transaccional MyISAM, pero puede provocar problemas de integridad en entornos de alta concurrencia en la modificación. En aplicaciones Web hay baja concurrencia en la modificación de datos y en cambio el entorno es intensivo en lectura de datos, lo que hace a MySQL ideal para este tipo de aplicaciones.

Plataformas

- MySQL funciona sobre múltiples plataformas, incluyendo:
- AIX
- BSD
- FreeBSD
- HP-UX
- GNU/Linux
- Mac OS X

- NetBSD
- Novell Netware
- OpenBSD
- OS/2 Warp
- QNX
- SGI IRIX
- Solaris
- SunOS
- SCO OpenServer
- SCO UnixWare
- Tru64
- eBD
- Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows Vista y otras versiones de Windows.
- OpenVMS

Características adicionales

- Usa GNU Automake, Autoconf, y Libtool para portabilidad
- Uso de multihilos mediante hilos del kernel.
- Usa tablas en disco b-tree para búsquedas rápidas con compresión de índice
- Tablas hash en memoria temporales
- El código MySQL se prueba con Purify (un detector de memoria perdida comercial) así como con Valgrind, una herramienta GPL
- Completo soporte para operadores y funciones en cláusulas select y where.
- Completo soporte para cláusulas group by y order by, soporte de funciones de agrupación

- Seguridad: ofrece un sistema de contraseñas y privilegios seguro mediante verificación basada en el host y el tráfico de contraseñas está cifrado al conectarse a un servidor.
- Soporta gran cantidad de datos. MySQL Server tiene bases de datos de hasta 50 millones de registros.
- Se permiten hasta 64 índices por tabla (32 antes de MySQL 4.1.2). Cada índice puede consistir desde 1 hasta 16 columnas o partes de columnas. El máximo ancho de límite son 1000 bytes (500 antes de MySQL 4.1.2).
- Los clientes se conectan al servidor MySQL usando sockets TCP/IP en cualquier plataforma. En sistemas Windows se pueden conectar usando named pipes y en sistemas Unix usando ficheros socket Unix.
- En MySQL 5.0, los clientes y servidores Windows se pueden conectar usando memoria compartida.
- MySQL contiene su propio paquete de pruebas de rendimiento proporcionado con el código fuente de la distribución de MySQL.

Mejoras futuras

El mapa de ruta de MySQL 5.1 indica soporte para:

- Particionado de la base de datos
- Backup en línea para todos los motores de almacenamiento
- Replicación segura
- Restricciones a nivel de columna
- Planificación de eventos
- Funciones XML

Características distintivas

Las siguientes características son implementadas únicamente por MySQL:

- Múltiples motores de almacenamiento (MyISAM, Merge, InnoDB, BDB, Memory/heap, MySQL Cluster, Federated, Archive, CSV, Blackhole y Example en 5.x), permitiendo al usuario escoger la que sea más adecuada para cada tabla de la base de datos.

- Agrupación de transacciones, reuniendo múltiples transacciones de varias conexiones para incrementar el número de transacciones por segundo.

Tipos de compilación del servidor

Hay tres tipos de compilación del servidor MySQL:

- Estándar: Los binarios estándar de **MySQL** son los recomendados para la mayoría de los usuarios, e incluyen el motor de almacenamiento InnoDB.
- Max (No se trata de MaxDB, que es una cooperación con SAP): Los binarios incluyen características adicionales que no han sido lo bastante probadas o que normalmente no son necesarias.
- MySQL-Debug: Son binarios que han sido compilados con información de depuración extra. No debe ser usada en sistemas en producción porque el código de depuración puede reducir el rendimiento.

Estado actual

La serie en desarrollo de MySQL Server actualmente, es la 5.1 al cual se añaden nuevas características en relación a la serie 5.0. La serie de producción actual de MySQL es 5.0, cuya penúltima versión estable es la 5.0.26 lanzada en octubre de 2006. Actualmente, se puede descargar la serie 5.0.27. La serie de producción anterior fue la 4.1, cuya versión estable es 4.1.7 lanzada en octubre de 2004. A estas versiones de producción sólo se arreglan problemas, es decir, ya no se añaden nuevas características. Y a las versiones anteriores solamente se les corrigen bugs críticos.

La versión que se usará en el sistema a crear es la 5.0.27 con licencia gratis.

CAPITULO III

SITUACION ACTUAL

3.1 FUNCIONAMIENTO DEL SISTEMA ACTUAL

En la Biblioteca de la Casa de la Cultura de Soyapango actualmente no tienen un sistema capaz de automatizar todos los procedimientos necesarios para una correcta administración de los recursos con que cuenta, debido a lo anterior se observa la necesidad de dotar a dicho centro de un software capaz de realizar todas las tareas necesarias de forma rápida y fácil de usar.

Debido a que dicha Casa de la Cultura no cuenta con los recursos económicos necesarios para equiparse adecuadamente, se desarrollara un sistema que sea portable, que no demande tantos recursos de hardware ni de software para su puesta en producción, que independientemente del sistema operativo que se use se pueda acceder al sistema, en general que no demande ningún tipo de gasto para dicho centro.

3.2 DESCRIPCIÓN DEL ENTORNO

La Casa de la Cultura de Soyapango se encuentra ubicada al costado oeste de la Iglesia San José, frente a la alcaldía municipal, cuenta con un área aproximada de 26 mts de largo por 10 de ancho, dicho recinto se encuentra dividido en dos partes, una dedicada a la biblioteca y la otra en donde se imparten los diferentes cursos de verano o charlas culturales.

El área dedicada a la biblioteca cuenta con dos estantes en donde se encuentran organizados los diferentes materiales, ya sean libros, revistas, obras literarias enciclopedias y literatura variada, permitiendo de esta manera hacer más fácil la búsqueda por parte del encargado de la biblioteca.

Actualmente, cuentan con una maquina de escribir un poco deteriorada, un teléfono con línea fija y un contómetro, lo antes mencionado es todo el equipo con que cuentan para llevar acabo la administración de la biblioteca.

3.3 PROCEDIMIENTOS

El procedimiento que se lleva a cavo para poder prestar material bibliográfico en la biblioteca de la Casa de la Cultura es el siguiente.

- Se debe pagar una cuota anual de afiliación a dicha biblioteca, la cual es de \$0.7 ctv anual, se puede renovar continuamente cuando dicho plazo se expire.
- Se busca en el catalogo de material (el cual es una caja con fichas que contienen la información de ubicación, titulo del material, autor y año de publicación).
- Se copia la información del material en un afiche y se le entrega al encargado de la biblioteca para su búsqueda.
- Dicho encargado revisa la ubicación solicitada y si hay en existencia el material solicitado lo saca para tomar sus datos, de lo contrario le informa al usuario que no poseen dicho material.
- Cuando encuentra en existencia el libro solicitado, anota todos sus datos y le pide al usuario su respectiva credencial de afiliación para poder cargarlo en un listado de entrega de material
- En dicho listado se anota tanto la información completa del usuario como del libro que se le prestara, al mismo tiempo, el encargado le otorga el tiempo de préstamo del material.
- Por ultimo, entrega el libro al usuario indicándole la fecha de devolución y en caso de retraso le informa de la multa por día que se le aplicara.

Lo anterior se puede describir como un proceso normal para préstamo de libro, dicho proceso puede variar de acuerdo a las necesidades de cada usuario y a las diferentes dificultades que pueda tener el encargado de la biblioteca.

El proceso de entrega de material se resume en dos pasos el cual se describe a continuación.

- El usuario se hace presente a la biblioteca y le comunica al encargado que entregara material, dicho encargado lo recibe junto con la credencial de afiliación.
- Luego, dicho encargado verifica en el listado de material entregado si concuerdan la información del libro con la credencial de afiliación, si todo es correcto, anota la fecha de entrega y en caso de multa le comunica al usuario el valor, de lo contrario entrega la credencial al usuario.

Al igual que para el proceso de préstamo, la entrega de material puede variar, dependiendo de las necesidades del usuario y a las dificultades que pueda tener el encargado de la biblioteca.

Para la actualizar los datos de un usuario, el bibliotecario le pide si credencial y luego procede a crear otra credencial con los nuevos datos, antes de realizar la nueva credencial se verifica si el usuario esta activo o no para poder proceder con el cambio.

Los reportes estadísticos de préstamos mensuales, trimestrales, anuales, usuarios morosos, libros mas solicitados y cualquier otro reporte se realiza de forma manual, es decir se verifica en el listado para poder proceder a sacar la información que se necesite.

Los procedimientos anteriores descritos son los que se llevan acabo actualmente en la biblioteca de la Casa de la Cultura de Soyapango.

3.4 USUARIOS INVOLUCRADOS

Los encargados de la administración de la Casa de la Cultura de Soyapango son dos personas, la administradora general de la casa de la cultura y al mismo tiempo de la biblioteca y la asistente, en ocasiones contratan personal extra para los diferentes cursos que imparten, dichas personas son empleadas de forma temporal.

Generalmente es la asistente quien se encarga del manejo de la biblioteca, siendo ella quien atiende a los diferentes usuarios tanto de biblioteca como los interesados en los cursos, la administradora por el contrario, se dedica a los tramites de donaciones, renovación de material, control de los recursos y a generar los diferentes reportes.

3.5 VENTAJAS Y DESVENTAJAS

Ventajas

Por ser una biblioteca relativamente pequeña (cuenta con 2500 ejemplares) en muchas ocasiones es fácil encontrar el material solicitado, por otra parte, el número de visitas no es muy grande por lo que no es tan difícil atender a los usuarios.

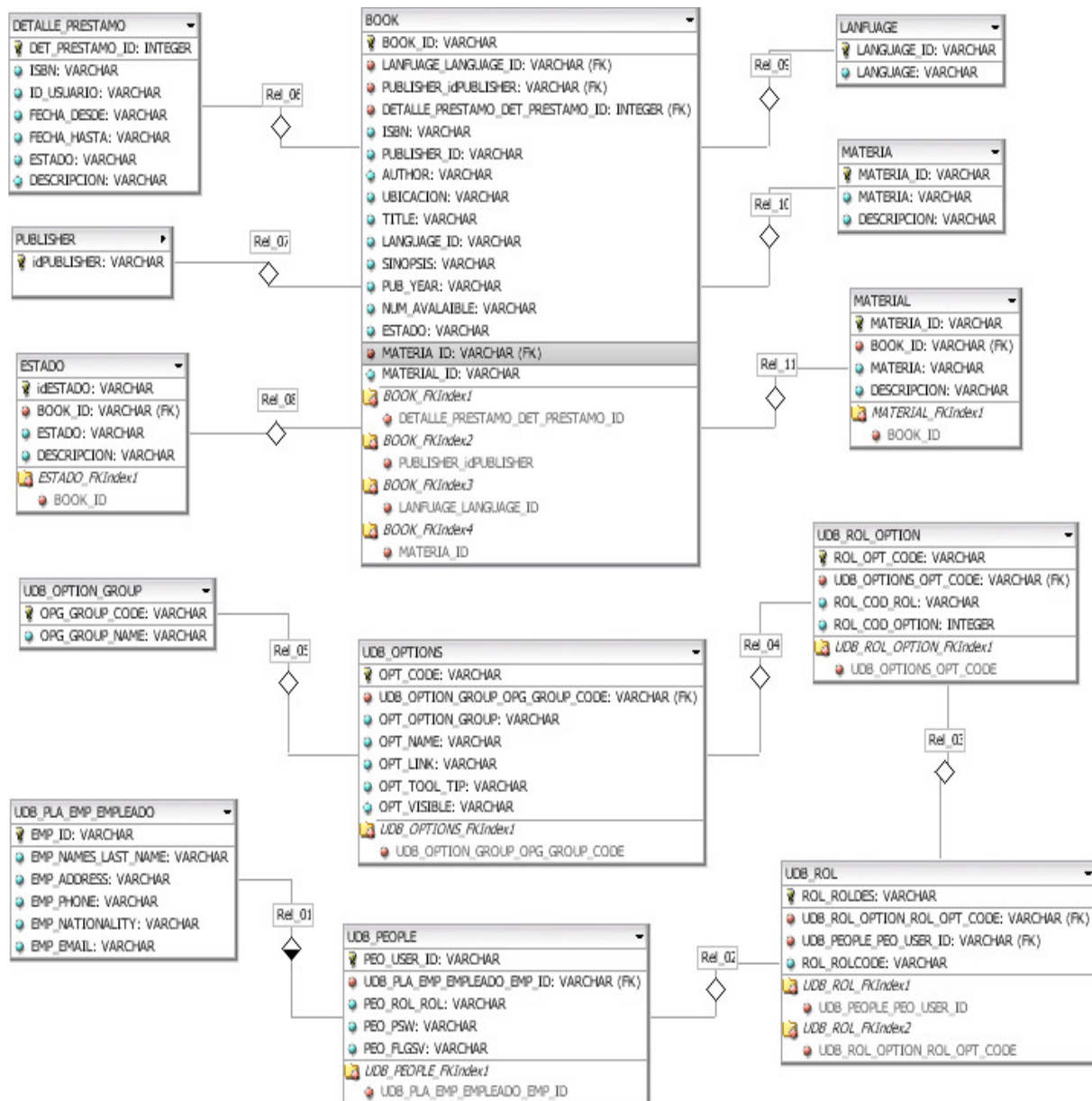
Desventajas

Debido a la falta de automatización, es muy tedioso sacar los diferentes reportes para su posterior análisis, además, no se puede saber de antemano si de un material hay en existencia hasta que se busca físicamente y se comprueba, por otro lado, en muchas ocasiones se pierden los afiches del catalogo lo que lleva al administrador a revisar periódicamente que la información se encuentre debidamente ordenada y completa.

CAPITULO IV

DISEÑO DE BASE DE DATOS

4.1 DIAGRAMA ENTIDAD – RELACION



4.2 DICCIONARIO DE DATOS

Entidad	UDB_ OPTIONS		
Descripción	Tabla donde almacena los datos relacionados a cada uno de las opciones del menu principal		
Nombre Campo	Tipo Dato	Constraints	Descripción
OPTION_CODE	NUMBER(10,2)	PK_APP_OPTIONS	Llave primaria que almacena el código de la opción del menú
OPTIONS_GROUP_CODE	NUMBER(10,2)	FK_APP_OPTION_GROUP	Llave foránea que enlaza el número del grupo de opciones del menú a la que pertenece dicha opción.
OPTIONS_NAME	VARCHAR2(50)		Campo que almacena el nombre de la opción.
OPTION_LINK	VARCHAR2(200)		Campo que almacena el link del action a la que hace referencia la opción una vez seleccionado.
OPTION_TOOLTIP	VARCHAR2(50)		Campo que almacena el nombre del tipo de opción.
OPTION_VISIBLE	NUMBER		Campo de control que establece la accesibilidad de las opciones.

Entidad	UDB__OPTIONS_GROUP		
Descripción	Tabla que almacena información sobre los grupos generales en donde se enlazaran varias opciones en ellas.		
Nombre Campo	Tipo Dato	Constraints	Descripción
OPTION_GROUPCODE	NUMBER(10,2)	PK_APP_OPTION_GROUP	Llave primaria que almacena el nombre del grupo de opciones.
OPTION_GROUPNAME	VARCHAR2(50)		Campo que almacena el nombre del grupo de opciones.

Entidad	UDB_ROL_ROLES		
Descripción	Tabla que almacena los distintos roles de los usuarios dentro del Jobsite.		
Nombre Campo	Tipo Dato	Constraints	Descripción
ROL_CODE	VARCHAR2(50)	PK_ROLES	Llave primaria que almacena el codigo del rol de los distintos usuarios.
ROL_DESC	VARCHAR2(50)		Campo que almacena la descripción del rol del usuario.

Entidad	UDB_OPTION_ROL		
Descripción	Tabla puente para enlazar la tabla de roles con la tabla de opciones para determinar que opciones le pertenece a cada rol.		
Nombre Campo	Tipo Dato	Constraints	Descripción
CODE	NUMBER(10,2)	PK_AOAOG	Llave primaria que almacena el codigo perteneciente a la opcion por roles.
ROL_CODE	VARCHAR2(50)	FK_OFR_OPTION	Llave foranea que almacena el rol del usuario.
OPTION_CODE	NUMBER(10,2)	FK_OPTION_ROL	Llave foranea que almacena la opcion a las que tiene derecho el usuario por su rol.

BIBLIOTECA

Entidad	DETALLE_PRESTAMO		
Descripción	Tabla donde se almacenan los detalles de los prestamos		
Nombre Campo	Tipo Dato	Constraints	Descripción
DET_PRESTAMO_ID	VARCHAR	PK_PRESTAMO_ID	Llave primaria de los detalles de los prestamos
ISBN	VARCHAR		Numero único de identidad de los libros
ID_USUARIO	VARCHAR		Campo ID_USUARIO
FECHA_DESDE	VARCHAR		Fecha de origen del prestamo
FECHA_HASTA	VARCHAR		Fecha en la que se tenia que hacer la devolucion
DESCRIPCION	VARCHAR		Campo que almacena comentarios sobre el prestamo

ntidad	BOOK		
Descripción	Tabla donde almacena los datos relacionados a cada uno de los libros		
Nombre Campo	Tipo Dato	Constraints	Descripción
BOOK_ID	VARCHAR	PK_BOOK_ID	Llave primaria que almacena el codigo de los libros
ISBN	VARCHAR		Numero unico de identidad de los libros
PUBLISHER	VARCHAR	Llave foranea a la tabla publisher	ID de editorial
AUTHROR	VARCHAR		Campo que almacena los autores de los libros
UBICACIÓN	VARCHAR		Campo que almacena la ubicación de los libros

TITLE	VARCHAR		Campo que almacena los títulos de los libros.
LANGUAGE_ID	VARCHAR	Llave foránea a la tabla lenguaje	Campo que almacena el id de lenguaje de los libros
SINOPSIS	VARCHAR		Campo que almacena una breve descripción de los libros
PUB_YEAR	VARCHAR		Campo que almacena el año de publicación de los libros
NUM_AVALAIBLE	VARCHAR		Campo que almacena el numero de ejemplares disponibles
ESTADO	VARCHAR	Llave foránea a la tabla estado	Campo que almacena el id de los estados de los libros
MATERIAL_ID	VARCHAR	Llave foránea a la tabla material	Campo que almacena el id de tipo de materiales de los libros
MATERIA_ID	VARCHAR	Llave forane a la tabla Materia	Campo que almacena la materia relacionada al libro

Entidad	LANGUAGE		
Descripción	Tabla que almacena el lenguaje		
Nombre Campo	Tipo Dato	Constraints	Descripción
Language	VARCHAR		Campo que almacena el lenguaje
Language_id	VARCHAR	Llave primaria para la tabla lenguajes	Campo que almacena el id de lenguajes

Entidad	MATERIA		
Descripción	Tabla que almacena las materias relacionadas a los libros		
Nombre Campo	Tipo Dato	Constraints	Descripción
MATERIA	VARCHAR		Campo que almacena la materia
MATERIA_ID	VARCHAR	Llave primaria para la tabla MATERIA	Campo que almacena el id de materia
DESCRIPCION	VARCHAR		Campo que en el cual se detalla la materia y areas.

Entidad	MATERIAI		
Descripción	Tabla que almacena los tipos de materiales		
Nombre Campo	Tipo Dato	Constraints	Descripción
MATERIAI	VARCHAR		Campo que almacena el tipo de material
MATERIAI_ID	VARCHAR	Llave primaria para la tabla MATERIAI	Campo que almacena el id de material
DESCRIPCION	VARCHAR		Campo que en el cual se detalla los tipos de materiales

Entidad	ESTADO		
Descripción	Tabla que almacena los tipos de ESTADOS		
Nombre Campo	Tipo Dato	Constraints	Descripción
ESTADO	VARCHAR		Campo que almacena el tipo de ESTADO
ESTADO_ID	VARCHAR	Llave primaria para la tabla MATERIAL	Campo que almacena el id de ESTADO
DESCRIPCION	VARCHAR		Campo que en el cual se detalla los tipos de materiales

Entidad	PUBLISHER		
Descripción	Tabla que almacena Las editoras		
Nombre Campo	Tipo Dato	Constraints	Descripción
Publisher	VARCHAR		Campo que almacena el editorial
Publisher_ID	VARCHAR	Llave primaria para la tabla PUBLISHER	Campo que almacena el id de EDITORIAL
Address	VARCHAR		Campo que en el que se detalle la direccion de contacto de la editorial
PHONE	VARCHAR		Campo en el que se detalla en telefono de contacto de la editorial.

CAPITULO V

DESARROLLO DEL SISTEMA

5.1 SEGURIDAD DE ESTRUCTURAS

Los creadores de Jakarta han adoptado extensamente para crear aplicaciones web en el mundo de J2EE la implementación de Struts ya que hace fácil crear una aplicación web porque emplea tecnologías estándar de J2EE como los servlets de Java, JavaBeans, las páginas de JavaServer, y las etiquetas de encargo. La validación de la entrada del usuario y la internacionalización de las vistas sobre el uso son algunas de las razones importantes y más conocidas para elegir struts como la base para su aplicación web.

Sin embargo, los Struts no fingen ofrecer una solución completa para todas las necesidades de la aplicación web; ésa nunca ha sido la meta del framework . el programador puede sustituir fácilmente las clases que desee modificar o que son necesarias para agregar los pedazos código que falta y así adaptarlo a las necesidades de la aplicación.

Configuración de seguridad de Struts

Las aplicaciones de nivel empresarial necesitan regulaciones rigurosas de la seguridad con roles diversos; cada rol también requiere su propio sistema de listas de control de acceso. Estos roles llegan a ser más importantes en los usos en Internet, que son accesibles a una audiencia más ancha. En la mayoría de los casos, la seguridad de uso debe controlar el acceso a cada cualidad que aparezca visible en la pantalla.

En struts-config.xml usted puede especificar los roles y atributos, una lista comandos delimitados de nombres roles de seguridad que no prohíban el acceso al objeto de ActionMapping en la consulta. Pero ése es todo allí está disponible el resto de la seguridad en struts, y no es seguramente suficiente la implementación la delegación de la seguridad que la aplicación necesita.

La solución genérica para la seguridad que se pueda utilizar por la mayoría de las aplicaciones web del nivel empresarial se divide en dos pequeñas, unidades manejables, las cuales son:

- **Page-level security**, que implica control de acceso en un nivel de la página
- **Attribute-level security**, que implica control de acceso en un nivel de atributo

Se puede solucionar estos dos problemas usando diversas tecnologías en el espacio de J2EE. La page-level security es la seguridad mas importante en la cual se permite al usuario acceder o negar a una página particular. Por una parte, la seguridad del attribute-level se hace cumplir en un nivel muy bajo, con los controles de seguridad realizados en cada atributo que se exhibe en la pantalla

En una organización típica, muchos usuarios desempeñan papeles múltiples, y una buena arquitectura de seguridad reconoce esto. Así, se puede asumir que se permite a los usuarios desempeñar papeles múltiples dentro de la aplicación. Con este requisito, y partido ya en unidades manejables, se puede realizar la solución de la seguridad.

El patrón de interceptación del filtro.

Los filtros del servlet introducidos con la versión 2.3 de la especificación de Java Servlet se prestan hacia una solución para el problema de seguridad del page-level. Se escribirá un filtro de seguridad que realice los cheques necesarios que aseguren de que se permite al usuario que tiene acceso a una página ver esa página. Porque el control de seguridad se debe realizar para cada usuario, se debe encapsular esta lógica en una clase de Java del ayudante y la almacenará en el user sesión. Por funcionamiento, se debe depositará el user del perfil de seguridad en esta clase así que no se tienen que hacer una llamada de base de datos cada vez que una petición de una aplicación. En esta aplicación, la información del rol de la seguridad se almacena en una base de datos, pero se podría reescribir fácilmente código para almacenarlo en alguna parte, como adentro un archivo de XML. La clase de seguridad es una puesta en práctica del interfaz de SecurityHelper, que se ilustra en el siguiente código.

```
public interface SecurityHelper {  
    //method to check page level access
```



```

    public boolean isAccessAllowed(String uri);

    //method to check if the attribute can be viewed by the user
    public boolean isViewableField(String page, String attribute);

    //method to check if the attribute can be edited by the user
    public boolean isEditableField(String page, String attribute);

    //method to refresh the cached user roles if there is some
change
    public void refreshCache();
}

```

La clase de SecurityFilter utiliza la clase de SecurityHelper, ilustrada en el listado siguiente para realizar los cheques del page-level. El método isAccessAllowed (uri) se utiliza para considerar si un usuario tiene permiso para ver o para modificar la página especificada por uri. La página real se exhibe que puede variar dependiendo del resultado del proceso. Sin embargo, se asume que cada URI es individual, cuando está a pasado a isAccessAllowed (), da lugar a una relación para fijar las páginas; si no lo hace, después se puede modularizar el uso para alcanzar este efecto. Para realizar los controles de seguridad, se elimina del Java Servlet specification's el método doFilter(), como se muestra a continuación.

```

....
public class SecurityFilter implements Filter {
    public void doFilter(ServletRequest req, ServletResponse res,
                        FilterChain chain) throws IOException,
ServletException {
        ...
        //Get the user's security profile of the user from the
session
        //If the session is invalid, redirect to some login/invalid
page.
                                SecurityHelper    userSecObj    =
session.getAttribute(SEcurity_PROFILE);
        ...
        //Compare the current URI with the access associated with
the access
        //to that URI for the role using the userSecObj.
        accessAllowed = userSecObj.isAccessAllowed(requestUri);
        if(accessAllowed) {
            //If the user is allowed access to the URI, let the flow
proceed as normal
            chain.doFilter(request, response);

```

```

        }
        else{
            //if the user is not allowed access, you can redirect him
            to an error page.
            response.sendRedirect("AccessDenied.jsp");
        }
    }
    ...
}

```

Este código asume que el user, el perfil (o el rol) está ya en la sesión cuando la petición viene al filtro. Puesto que no se está asumiendo ninguna sola forma de autenticación, se puede utilizar este modelo conjuntamente con las herramientas como SiteMinder, que puede hacer la autenticación la aplicación. Se obtiene el URI del objeto de la petición. Se observe que se puede hacer esta comparación con una lista de roles de usuario, si el uso asigna que por mandato los roles de cada usuario múltiplo del juego; la lógica para ésa sería encajada en la puesta en práctica para la clase de SecurityHelper.

Ahora que se ha tomado la seguridad del page-level, procedemos a la segunda parte del problema, que es asegurarse de que el usuario no ve ni modifica ninguna cualidad que él o ella no debe tener acceso a la aplicación.

Etiquetas de tags y struts de JSP

Se desarrollan etiquetas de tags de JSP para alcanzar controles de seguridad en el terreno, utilizará dichas etiquetas de seguridad para cada tipo de componente que planea exhibir, por ejemplo una caja combinada o un campo del texto. se utilizará la clase de SecurityHelper contorneada en el bloque de código 1 para los cheques de los derechos de acceso. También las etiquetas desarrolladas en struts para construir etiquetas de tags de la seguridad. Los struts tienen puestas en práctica para la mayoría de las etiquetas del HTML y apoyan la mayoría de las cualidades que se requieran para estas etiquetas. El isEditableField de los métodos de la clase de SecurityHelper () y hace del isVisibleField () la lógica para asegurarse de que el campo exhibido tenga el permiso requerido para los roles del usuario, y hacen visible componentes o lisiados, como lo muestra el siguiente código.

```
....  
    public class SecurityComboTag extends  
org.apache.struts.taglib.html.SelectTag {  
        //this attribute stores the attribute that's being shown on  
the screen.  
        private String attribute;  
        //this attribute stores the page that's being shown  
        private String page;  
        ...  
        public int doStartTag() throws JspException{  
            //check the access using the SecurityHelper class  
            //Helper instance is obtained from the session  
            ...  
            isVisible = securityHelper.isVisibleField(page,  
attribute);  
            if(!isVisible) {  
                return SKIP_BODY;  
            }  
            isModifiable = securityHelper.isVisibleField(page,  
attribute);  
            if(!isModifiable) {  
                setDisabled(true);  
            }  
            ...  
        }  
    }  
}
```

La clase tiene dos cualidades: uno para almacenar el nombre de la cualidad que es referenciado actualmente usando la etiqueta selecta, y uno para la página en la cual la etiqueta está presente. Se verifica la página y los atributos al SecurityHelper class's los métodos para considerar si se permite al usuario ver y/o modificar los atributos particulares, y cambiar la exhibición. Toda la lógica de la seguridad se encaja en la puesta en práctica de la clase de SecurityHelper, así manteniendo el código del control de seguridad un lugar central.

5.2 TIPOS DE USUARIO

Para el sistema de administración de la biblioteca de la Casa de la Cultura de Soyapango se han creado tres tipos de usuarios, los cuales se describen a continuación con sus respectivas características:

- **Administrador IT:** Usuario capaz de realizar cualquier tipo de modificación en el sistema, desde crear nuevos usuarios hasta modificar perfiles de vistas de pantallas, se puede decir que es el usuario que gobierna el sistema en su totalidad.
- **Administrador:** Este usuario tiene como capacidad la administración del sistema, creando otros usuarios con características de administrado y usuarios estándar, administra el flujo de préstamo de material, reservas, multas, reportes e historial de préstamo, dicho usuario tiene capacidades menores que el **administrador IT**, pero posee mayor privilegios que el estándar.
- **Estándar:** Posee menores privilegios que los dos anteriores y es el que se le designa a las personas que visitan la biblioteca para préstamo de material, solo pueden buscar material, reservar material, historial de usuario y revisión de multas.

5.3 VALIDACIÓN

Struts Validator

El **Framework Struts Validator** es distribuido junto con el Framework de Struts. Se utiliza para implementar las validaciones de los datos introducidos en el cliente.

Las validaciones pueden realizarse tanto en *el lado del cliente* (mediante JavaScript, se recomiendan por tiempo de respuesta y para evitar tráfico innecesario en la red y evitar procesos innecesarios en el servidor) y en *el lado del servidor* (se aconseja por seguridad, aunque se realicen dos veces).

Configuración del Framework

La configuración se realiza mediante dos ficheros de propiedades en formato XML.

Además para utilizar este Framework se debe declarar en el struts-config.xml.

Validator-rules.xml.

Indica los tipos de validaciones que se realizan, así como las clases que implementan estas validaciones.

Para validaciones típicas nos será suficiente utilizar el fichero de configuración por defecto y las clases que acompañan a la distribución. Pero si la aplicación necesitara validaciones más complejas se pueden añadir a este fichero e implementarlas en las clases creadas.

A continuación, un ejemplo de regla de validación (required) que indica la clase que implementa la validación en el servidor y el código JavaScript que se ejecutará en el cliente:

```
...  
<validator name="required"
```

```

classname="org.apache.struts.validator.FieldChecks"
method="validateRequired"
methodParams="java.lang.Object,
    org.apache.commons.validator.ValidatorAction,
    org.apache.commons.validator.Field,
    org.apache.struts.action.ActionMessages,
    org.apache.commons.validator.Validator,
    javax.servlet.http.HttpServletRequest"
msg="errors.required">
<javascript>
  <![CDATA[
    function validateRequired(form) {
      var bValid = true;
      var focusField = null;
      var i = 0;
      var fields = new Array();
      oRequired = new required();
      for (x in oRequired) {
        if ((form[oRequired[x][0]].type == 'text' ||
          form[oRequired[x][0]].type == 'textarea' ||
          form[oRequired[x][0]].type == 'select-one' ||
          form[oRequired[x][0]].type == 'radio' ||
          form[oRequired[x][0]].type == 'password') &&
          form[oRequired[x][0]].value == ) {
          if (i == 0)
            focusField = form[oRequired[x][0]];
          fields[i++] = oRequired[x][1];
          bValid = false;
        }
      }
      if (fields.length 0) {
        focusField.focus();
        alert(fields.join('\n'));
      }
      return bValid;
    }
  ]]>
</javascript>
</validator>

```

validation.xml

Indica los campos que se validaran de cada formulario utilizando en este Framework, así como las validaciones que se aplicara a cada uno y los mensajes a mostrar en caso de no superar las validaciones.

La estructura de este fichero es la siguiente:

```
<!DOCTYPE form-validation PUBLIC
    "-//Apache Software Foundation//DTD Commons Validator Rules
    Configuration 1.1.3//EN"
    "http://jakarta.apache.org/commons/dtds/validator\_1\_1\_3.dtd
```

Los elementos declarados en **global** son comunes a todos los formularios. "xxx" es el nombre del/de los formulario/s a validar.

Un ejemplo de una validación típica de un formulario para identificarse en una aplicación puede ser:

```
<form name="identificarForm">
  <field
    property="stIdentificador"
    depends="required">
    <arg key="identificarForm.stIdentificador.displayname"/>
  </field>
  <field
    property="stContraseña"
    depends="required,mask">
    <arg key="identificarForm.stContraseña.displayname"/>
    <var>
      <var-name>mask</var-name>
      <var-value>^[0-9a-zA-Z]*$</var-value>
    </var>
  </field>
  <field
    property="datosNotificaciones.codPostal"
    depends="required, mask">
    <var>
      <var-name>mask</var-name>
      <var-value>${codPostal}</var-value>
    </var>
    <msg name="mask" key="errors.codPostal"/>
    <arg0 key="datosNotificaciones.codPostal"/>
  </field>
</form>
```

Struts-config.xml

Finalmente, debemos indicar al fichero de configuración de Struts, **struts-config.xml**, que utilizaremos este plug-in además de indicar las rutas de los ficheros de configuración que se utilizarán.

En este código se indica que se usará el fichero validator-rules.xml que viene por defecto en la distribución de Struts :

```
<plug-in className="org.apache.struts.validator.ValidatorPlugIn">
  <set-property
    property="pathnames"
```

```
        value="/org/apache/struts/validator/validator-rules.xml,  
            /WEB-INF/validation.xml"/>  
</plug-in>
```

Mensajes de Error

Si la validación devuelve un error, devolverá un código de error y una serie de parámetros para construir el mensaje de error que se presentará.

Por defecto, estos mensajes se encuentran en el fichero `.properties` que almacena el resto de códigos de los mensajes de la aplicación, por defecto *MessageResources.properties*.

En el código anterior, en caso de no superar la validación se devolverá *errors.required* como código de error. Por tanto, el fichero de mensajes *MessageResources.properties* debe contener una entrada para este código.

```
errors.required={0} es un dato obligatorio.
```

La utilización de parámetros (se declaran en el fichero **validation.xml**) en los mensajes, `{0}`, permite reutilizar los mensajes. En el `.properties` de mensajes también se debe declarar una entrada para los parámetros que se usaran:

```
# -- loginAdministracion --  
identificarForm.stIdentificador.displayname=Identificador  
identificarForm.stContraseña.displayname=Contraseña
```

Utilización del Framework

Finalmente, para realizar las validaciones en el **lado del servidor**, los formularios deben extender de *org.apache.struts.validator.ValidatorForm*. Para realizar las validaciones JavaScript en el **lado del cliente** se debe indicar en el `jsp` que vamos a utilizar este Framework.

CAPITULO VI

DOCUMENTACIÓN

MANUAL DEL USUARIO UDBOOK

26/02/2009

UDBOOK

RIVAS SALGADO, JAIRO BENJAMIN

VALIENTE MARTINEZ, CESAR ADOLFO PAOLO

INTRODUCCION A UDBOOK

UDBOOK es una herramienta para la administración de la Biblioteca de la Casa de la Cultura de Soyapango, la cual permitirá un mejor desempeño de todos los procesos que se realizan en dicho centro

Descripción del manual

Este documento presenta los pasos necesarios para utilizar UDBOOK e inducir al usuario en los procesos a seguir para obtener un mayor provecho de esta útil herramienta.

Se divide dicho manual en dos etapas fundamentales, una es la instalación de los elementos necesarios para poder tener acceso a dicha aplicación y la otra parte explica el uso del software de manera rápida y concisa.

La parte de instalación de componentes abarca desde los requisitos mínimos del equipo, hasta la versión del jre necesario para correr dicha aplicación.

La segunda parte engloba todo los procesos de uso cotidiano del sistema de administración bibliotecario UDBOOK, pudiendo agregar material, realizar préstamo, reservar, realizar reportes y otros procesos que se llevan a cabo en la

Casa de la Cultura de Soyapango.

UDBOOK

Modulo

1

La Instalación

La instalación del sistema bibliotecario UDBOOK para la Casa de la Cultura de Soyapango con lleva un serie de pasos que incluyen recursos mínimos de Hardware y Software. (Aunque para ello no es necesario instalar una aplicación)

Requisitos necesarios para la uso de UDBOOK

a. Hardware

En la tabla # 1 se detallan los recursos recomendados y mínimos para el uso

Recomendado		Mínimo	
Procesador	Pentium 300	Procesador	Pentium 200
RAM	32Mb	RAM	16Mb
Video	8Mb	Video	4Mb
CD-ROM	16x	CD-ROM	16x
Color	Verdadero 24bits	Color	Verdadero 16bits
Disco Duro	4Gb	Disco Duro	2GB

Tabla # 1: Requisitos de Hardware

b. Software

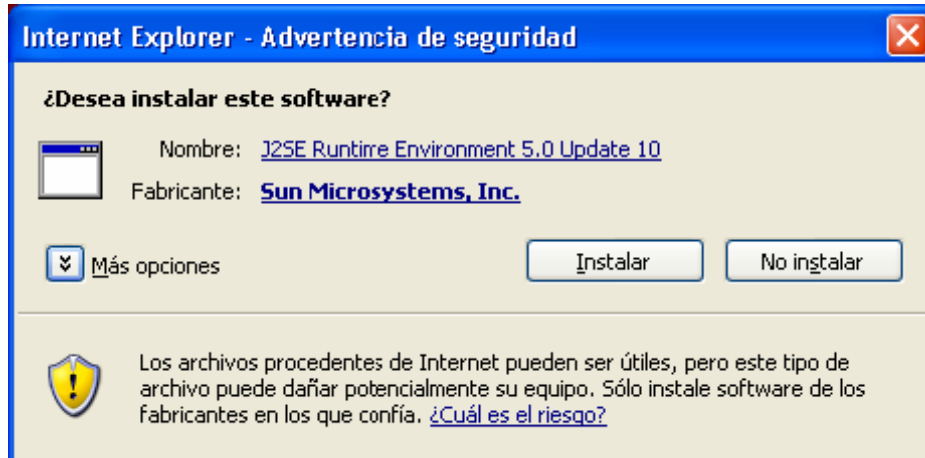
- Se requiere un Sistema Operativo el cual soporte la maquina virtual de Java (jre v1.6.0)
- La resolución de la pantalla debe establecerse en 1024 x 768 píxeles, para una visualización optima.
- Se requiere tener previamente instalado el Java Runtime Environment v 1.6.0 (jre v1.6.0)

Instalando Java Runtime Environment

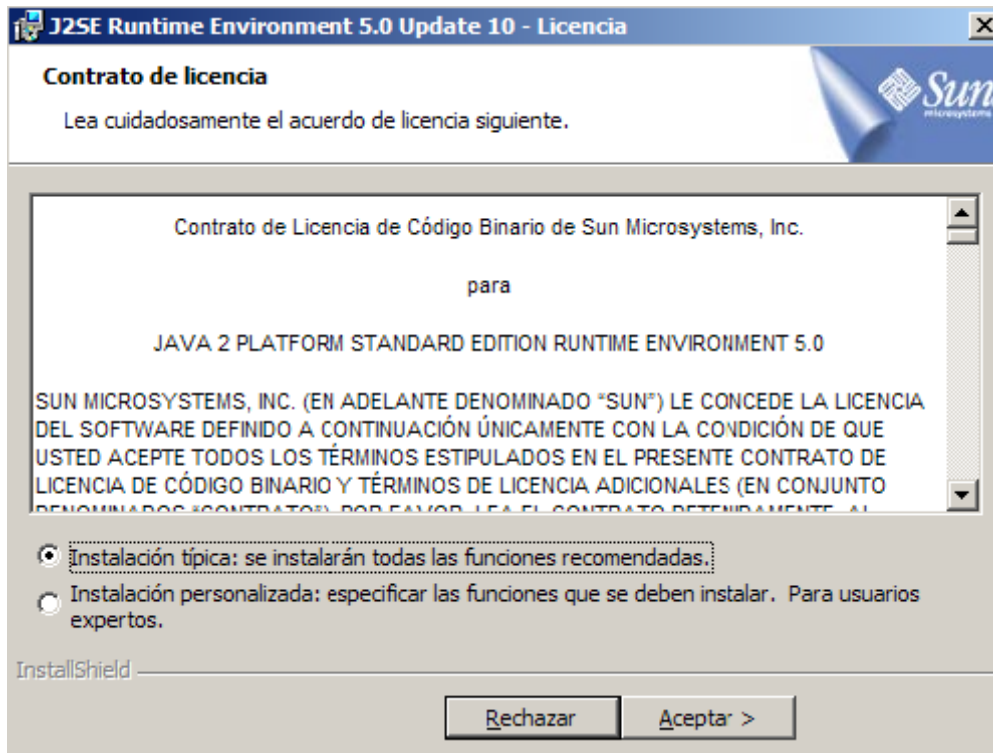
Paso a Paso

Información Adicional

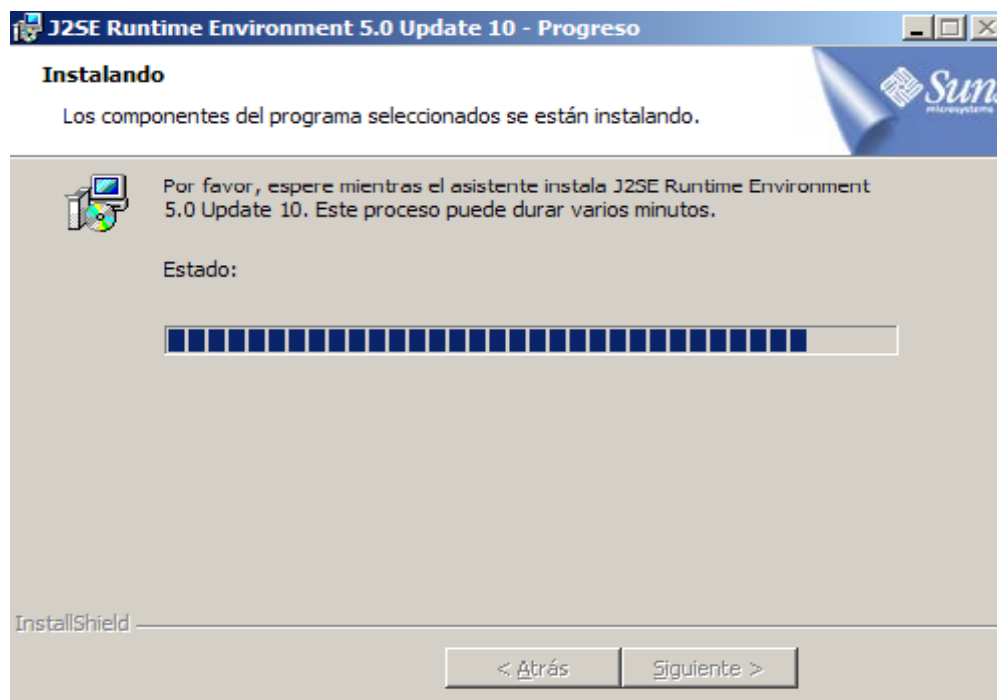
1. Ubíquese en el directorio donde se encuentra el instalador de Java Runtime Environment y dar doble click
2. Se nos preguntará si deseamos instalar la versión mínima: 1.5.0_10, se da click en el icono **Instalar**



3. Tras pulsar en "**Instalar**", aparecen varias pantallas de instalación y los siguientes cuadros de dialogo:



4. Se selecciona la **Instalación típica** y luego se da click en Aceptar y aparece la siguiente pantalla.



En dicha pantalla se muestra el avance de la instalación, por lo que el

usuario no tiene que realizar ninguna acción

5. Por último se da click en el icono de **finalización** para terminar la instalación, como se muestra en la siguiente pantalla.



Sistema Bibliotecario UDBOOK

Modulo

2

Lo Básico de UDBOOK

La herramienta está compuesta por una serie de ventanas fácilmente accedidas, las cuales permiten al usuario la navegación y uso de sus opciones.

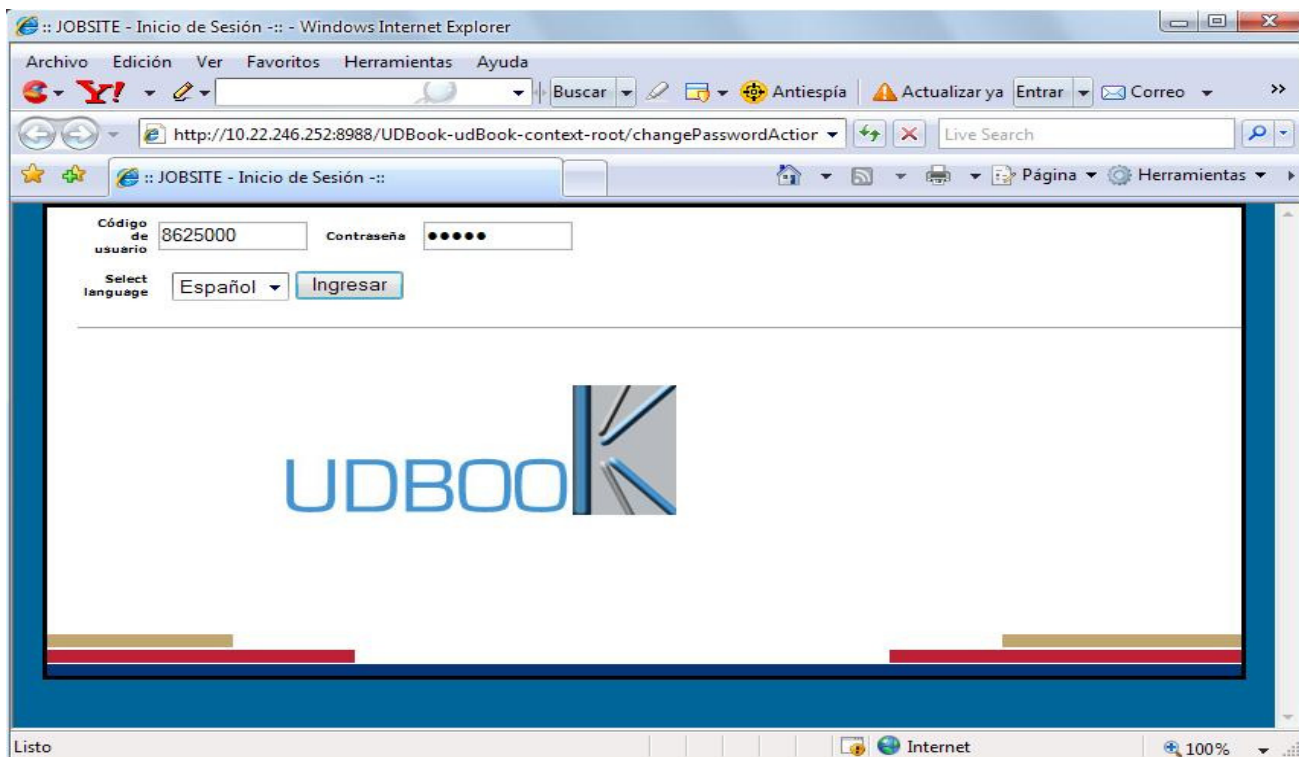
USO DE UDBOOK

1. Primeramente se debe acceder a la siguiente dirección desde Internet Explorer <http://10.22.246.252:8988/UDBook-udBook-context-root/login.do;jsessionid=0a16f6fc231c3569e79dd3ff4b55967dbfa933925828?accion=change> ó dar coble click al acceso directo que estará ubicado en

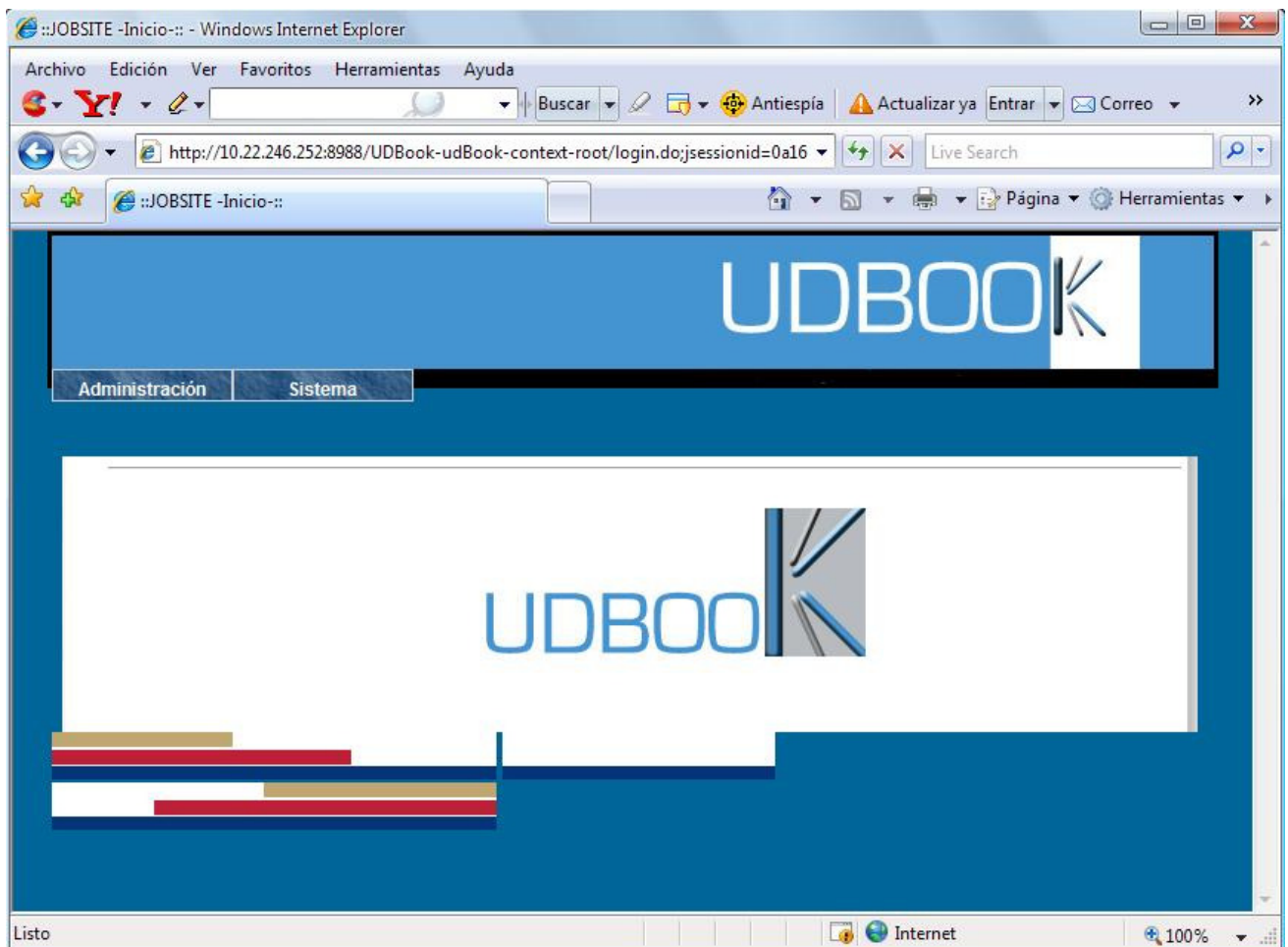


el escritorio de la computadora, el cual es el siguiente

2. Se presenta la siguiente pantalla donde se solicita el usuario y contraseña, ya sea para la cuenta de administrador o para usuario normal del sistema, para la cuenta de administrador se maneja los siguientes datos: **usuario: 8625000**, **contraseña: admin**, se digitan dichos datos en sus respectivas casillas y se da click en ingresar.

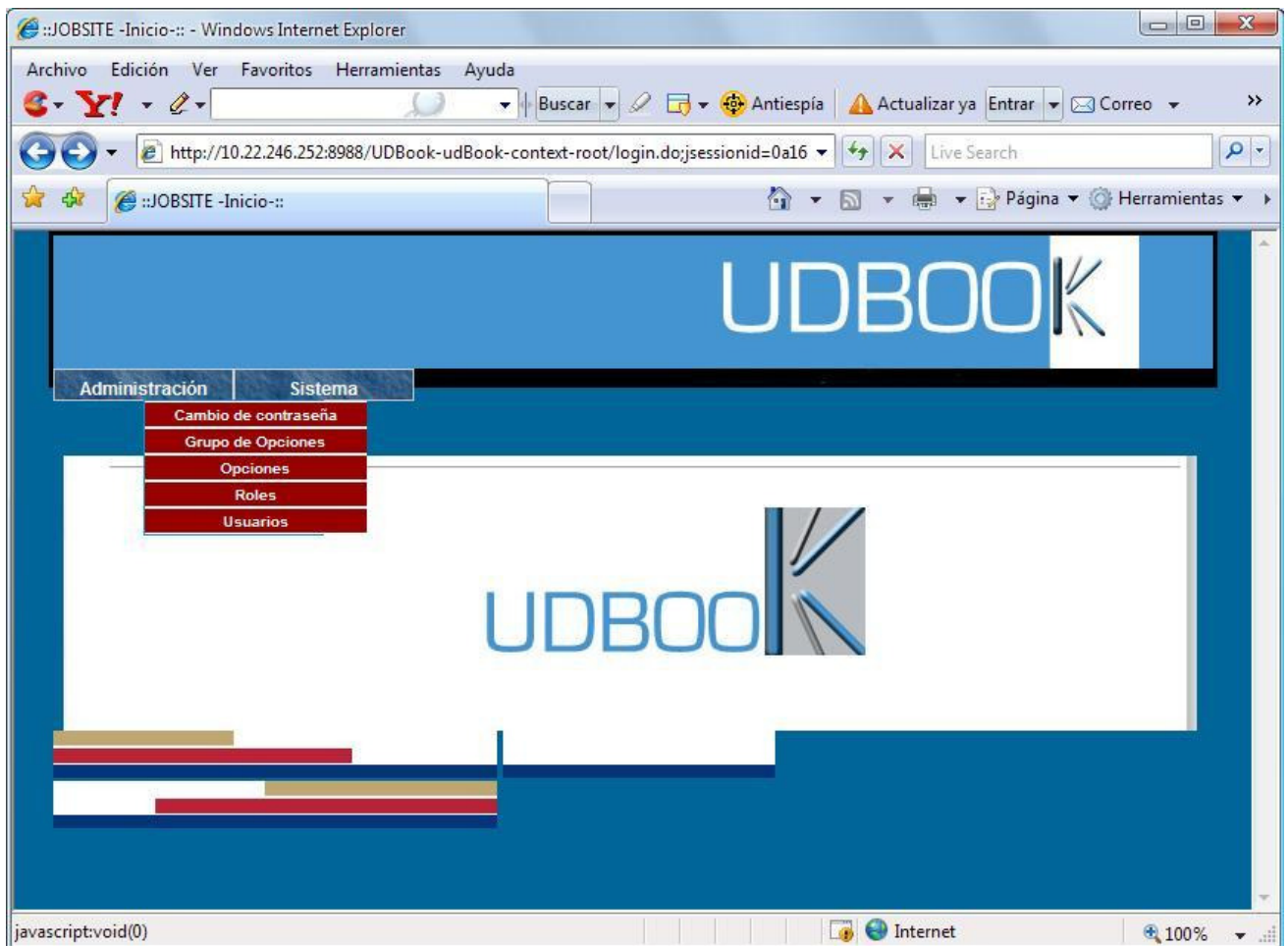


3. Luego de haber introducido el usuario y contraseña correcta se presenta la siguiente pantalla de bienvenida, con los diferentes menús, a los que se tiene acceso, como se muestra en la pantalla siguiente:

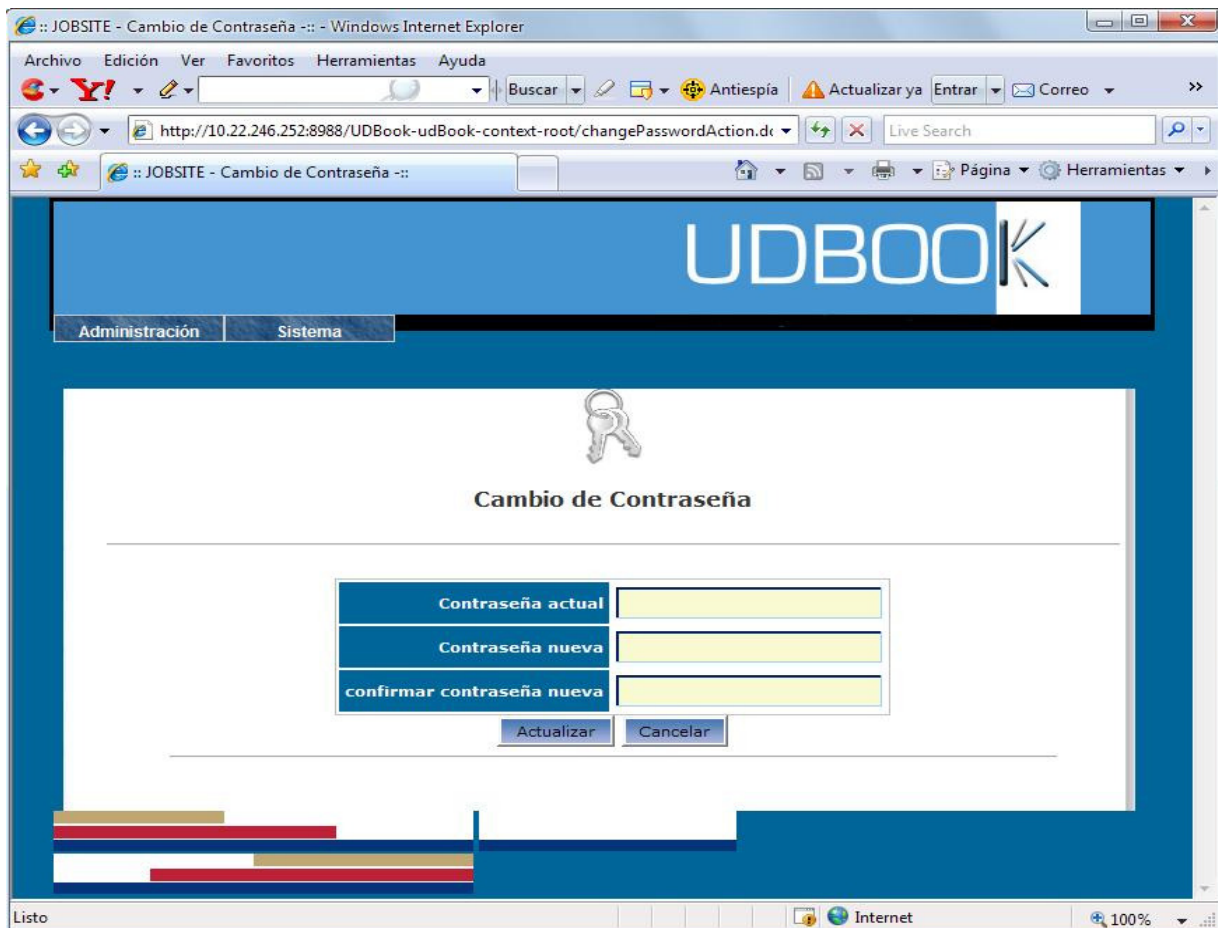


En este punto se describirán de manera separada los menús administración y sistemas, con sus respectivos sub menús y las operaciones que se pueden realizar.

- **Menú Administración:**



1. **Cambio de contraseña:** Esta opción como su nombre lo describe, se utiliza para realizar cambio de contraseña, tanto para administradores, como para usuarios normales, la pantalla que se muestra al seleccionar dicha operación es la siguiente:

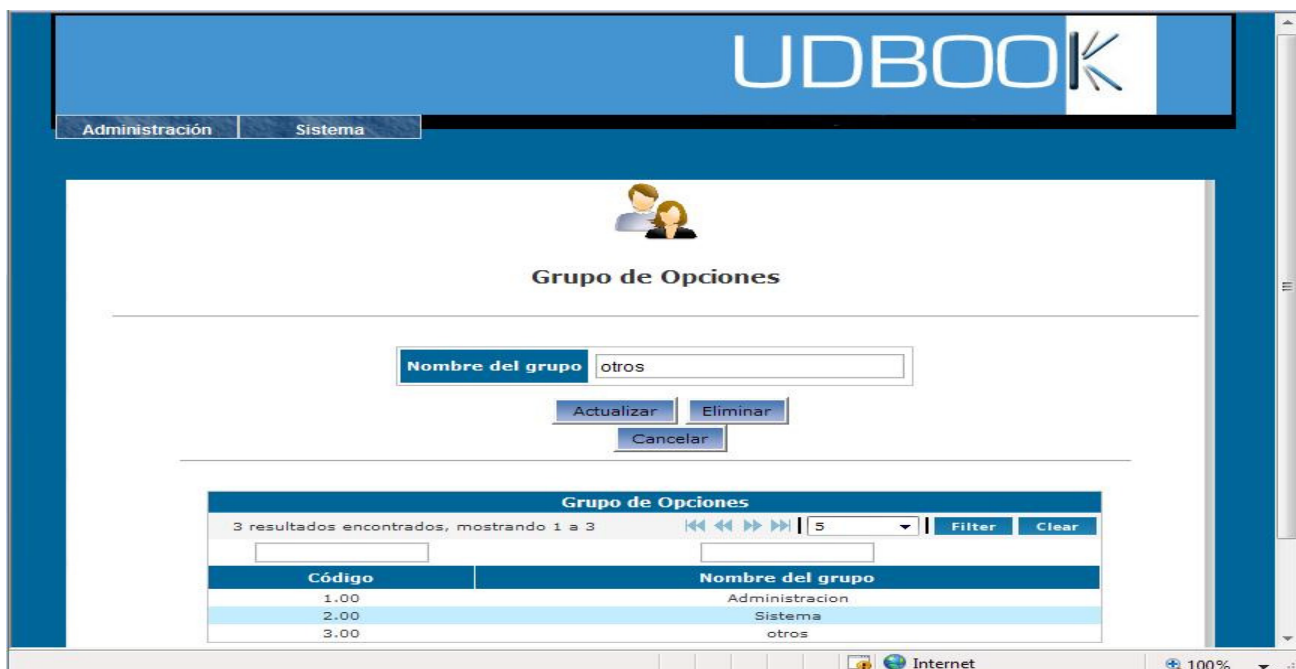


En donde se debe digitar la contraseña actual y luego la contraseña nueva que se le dará al usuario y hay que confirmar dicha contraseña para corroborar que este bien escrita, por último se da click en el icono de actualizar y se procesara dicho cambio.

2. **Grupo de opciones:** Este menú se utiliza para agregar ó eliminar grupos de usuarios, la siguiente imagen muestra los elementos con que cuenta y las operaciones que se pueden realizar.



Si se desea agregar un grupo de usuarios al sistema, se digita el nombre del grupo y se da click en agregar, para eliminar un grupo se muestra la siguiente imagen.



En donde se selecciona el grupo a eliminar y se da click en eliminar, luego se recibirá un mensaje de que se realizo la operación con éxito.

3. **Opciones:** Se utiliza para agregar en el menú más opciones o sub menús, esto permite la escalabilidad del sistema, ya que a medida aumenten necesidades, el software se puede ir acoplando perfectamente, los parámetros requeridos para esta operación son los siguientes; **Nombre de opción:** el cual es el que aparecerá en el menú, **Nombre de grupo:** Al cual pertenece dicha opción, **Nombre de roles:** Al cual estará asignado dicha opción puede ser administrador IT o usuario estándar, **Enlace:** pagina a la que estará vinculado dicha opción, **Tooltip:** Es el nombre del encabezado de la pagina y **Visible:** Decide si se quiere mostrar o no la opción que se creara

The screenshot shows a web application interface for managing application options. At the top, there is a header with a user icon and the title 'Opciones de aplicación'. Below this is a form with several input fields: 'Nombre opción', 'Nombre de grupo', 'Nombre de roles', 'Enlace', 'Tooltip', and 'Visible' (a dropdown menu set to 'True'). There are 'Agregar' (Add) and 'Cancelar' (Cancel) buttons below the form. Below the form is a table titled 'Opciones de aplicación' showing a list of options. The table has columns for 'Código', 'Nombre opción', 'Nombre de grupo', and 'Enlace'. The table shows 14 results, with the first 5 displayed. The table is filtered by '5' items.

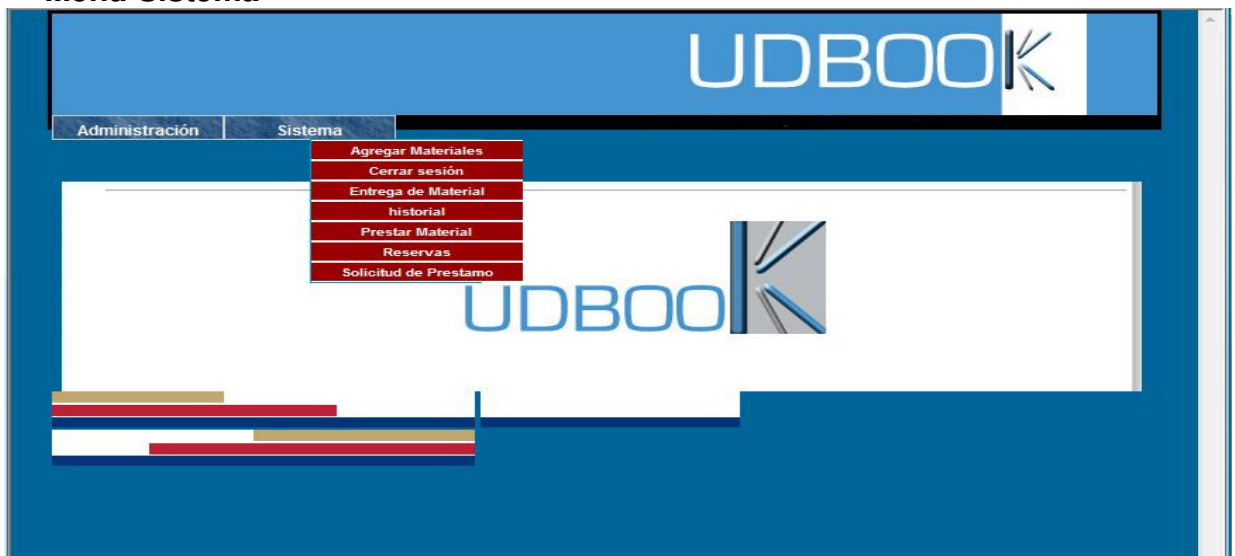
Código	Nombre opción	Nombre de grupo	Enlace
1.00	Grupo De Opciones	Administracion	optionGroupAction.do?accion=Find
2.00	Roles	Administracion	rolesAction.do?accion=Find
3.00	Prestamo	Sistema	prestamoAction.do?accion=Find
4.00	Opciones	Administracion	optionAppAction.do?accion=Find

Si se desea eliminar una opción ya creada, solo se da click en el nombre de la opción y se activara el icono de eliminar, se da click en dicho icono y se eliminara la opción.

4. **Usuarios:** Interfaz para la administración de usuarios, donde se puede crear, buscar y eliminar usuarios, para agregar usuarios se debe introducir **nombre de usuario**, **códigos**, **nombre de rol** al que pertenecerá **contraseña**, **confirmar contraseña** para verificar la integridad de la misma, para eliminar un usuario, se busca o se selecciona el usuario e inmediatamente se activa el icono de eliminar, se da click en dicho icono y se procesara la operación.

Código 86	Código M4	nombre de usuario	Nombre de rol
8623021	032865	PAOLO VALIENTE	Administrador IT
8624006	032864	CHRISTIAN EMILIO SANCHEZ MEJIA	Administrador IT
8625000	032862	ADMIN	Administrador IT

- **Menú Sistema**



1. **Agregar materiales:** En esta opción se administra los materiales bibliográficos, se puede agregar, borrar o actualizar material, para agregar material nuevo se tiene que llenar todos los campos que se presentan a continuación en donde todos los campos son obligatorios, para actualizar un material se selecciona dicho material y se procede a realizar el cambio en algún campo y se da click en el icono de actualizar, para borrar un material, se selecciona dicho material y se da click en el icono de borrar.

Agregar Materiales

Author	<input type="text"/>	Titulo	<input type="text"/>
Lenguaje	<input type="text"/>	Sinopsis	<input type="text"/>
Publicacion	<input type="text"/>	ISBN	<input type="text"/>
Copias	<input type="text"/>	Ubicacion	<input type="text"/>
Editorial	<input type="text"/>	Materia	<input type="text"/>
Tipo de Material	<input type="text"/>		

Opciones de aplicación

1 resultados encontrados, mostrando 1 a 1

Author	Titulo	Lenguaje	Materia	Tipo de Material	Publicacion
Prueba1_autor	Prueba1_titulo	español	Sociales	CD	2009

Listo Internet 100%

2. **Cerrar sesión:** Esta opción se utiliza para salir del sistema a la página de registro de usuario, se utiliza para cambiar de tipo de usuario.

3. **Entrega de material:** Esta opción se utiliza para la entrega de material a los usuarios, para realizar una entrega se busca ya sea por tipo de material, Id de usuario ó ISBN, luego se verifica en el estado si esta en préstamo o disponible, como lo muestra la siguiente pantalla.

UDBOOK

Administración Sistema

Entrega Materiales

2 resultados encontrados, mostrando 1 a 2

Filter Clear

ISBN	ID Usuario	Fecha Iniccion	Fecha Fin	Estado	Observacion
2	8625000			Reserva	
2	8625000	2009-02-01	2009-02-01	Prestamo	

Si el material está disponible, se muestra la siguiente imagen, en la cual se debe de completar todos los campos, eligiendo también la fecha de inicio del préstamo y la de finalización del préstamo, luego se da click en el icono de agregar.

Author	Prueba1_autor	Titulo	Prueba1_titulo
Usuario	8625000	Estado	
Fecha Inicion	02/26/2009	Fecha Fin	02/27/2009
Observacion	prestamo personal		

Agregar **Cancelar**

Si el material esta en préstamo no se podrá realizar otra operación, solo reservar, el cual está en otra opción en el menú de sistema.

4. **Historial:** Esta opción se utiliza para verificar el récord de un usuario, el número de material prestado y las multas impuestas, la pantalla que describe a dicha opción es la siguiente.

Fecha Inicio	04/02/09	Fecha Fin	25/02/09
Estado	Entregado	Usuario	paolo

Actualizar **Cancelar**

5. **Prestar material:** En esta pantalla se muestran en una tabla el listado de los materiales que se han solicitado para préstamo, para proceder con dicho préstamo se selecciona el material a presta y automáticamente se carga toda la información, luego se da click en el botón de actualizar y se procesara la operación, como se muestra a continuación.



PRESTAMOS

ISBN	12345	Reservas	Prueba1_titulo
Editorial	prueba	Tipo Material	CD
Lenguaje	español	Año	2009
Fecha Prestamo	25/02/09	Fecha Devolución	28/02/09
Id Copia	2		

ActualizarCancelar

Opciones de aplicación

1 resultados encontrados, mostrando 1 a 1

5

Filter

Clear

ISBN	User	Editorial	Titulo	Autor	Tipo Material	Año	Ubicacion
12345	8625000	prueba	Prueba1_titulo	Prueba1_autor	CD	2009	Prueba1_ubicación

CONCLUSION

El sistema para la administración de la biblioteca de la Casa de la Cultura de Soyapango constituyo un esfuerzo por ambas partes, tanto del lado de cliente, quienes lo integran las encargadas de dicho centro, así como del lado de la parte de los desarrolladores, obteniendo como resultado un producto que cumple con las necesidades requeridas.

UDBOOK es capaz de realizar las tareas administrativas de la casa de la cultura, realizando todo el flujo de préstamo, reservación, búsqueda de ubicación de libros, consulta de multas, creación de usuarios y generación de reportes lo que permite agilizar los procesos que se llevan en forma manual actualmente.

El sistema posee una interfaz amigable al usuario lo que permite una mejor adaptación a las diferentes pantallas, ya que cuenta con menús contextuales que llevan de la mano al usuario para realizar las tareas que desea.

Se incluye en este documento la guía necesaria para el uso del sistema, para que tanto los administradores y los usuarios estándar pueden adaptarse de manera rápida al ambiente y a las diferentes pantallas con que cuenta el software

De forma general el sistema ayudara en gran medida a la informatización de dicho centro, por lo que marca un punto de partida para que se pueda seguir expandiendo hasta convertirse en una biblioteca que posea todas las herramientas necesarias para la búsqueda de información, tanto en libros y revistas, como en internet.

GLOSARIO.

A

Analista.

Es aquel individuo que ejerce las tareas de análisis de los sistemas informáticos, con el fin de automatizarlos.

B

Biblioteca virtual.

Biblioteca que contiene textos digitales de los diferentes tomos de libros, revistas tesis y otra documentación de consulta

C

Código Abierto.

Del inglés open source, es el término con el que se conoce al software distribuido y desarrollado libremente.

E

Editor UML.

Programa de software que permite utilizar lenguaje UML para realizar su funcionalidad.

P

Programador.

Individuo que escribe programas utilizando un lenguaje de computación, también conocido por desarrolladores de software.

S

Sistema.

Combinación de hardware y software necesarios para cumplir un objetivo.

Software.

Los programas de computadoras, procedimientos, además la documentación y asociados que forman parte de un sistema.

Software Comercial.

Es el software, libre o no, que es comercializado, es decir, que las compañías que lo producen, cobran dinero por el producto, su distribución o soporte.

U**UML.**

UML (Lenguaje Unificado de Modelado) es un lenguaje gráfico para visualizar, especificar, construir y documentar los componentes de un sistema software. UML permite tanto la especificación conceptual de un sistema como la especificación de elementos concretos, como pueden ser las clases o un diseño de base de datos.

BIBLIOGRAFIA

CHOWDHURY, G.G., CHOWDHURY, Sudatta, *Introduction to digital libraries*, Facet publishing, London, 2003.

SCHMULLER, Joseph. APRENDIENDO UML EN 24 HORAS. Prentice Hall
SISTEMA DE INFORMACION BIBLIOGRAFICA DE LA CEPAL: MANUAL DE
REFERENCIA. Carmen Vera Arendt, Santiago de Chile, septiembre de 2003

SEQUEIRA, DEYANIRA, Bibliotecas y archivos virtuales, Buenos Aires, Nuevo
Parhadigma, 2001

Sitios Web

www.wikipedia.com Enciclopedia Libre.

www.monografias.com Sitio Web con tutoriales y monografías.

www.inei.gob.pe Sitio Web del Instituto Nacional de Estadística de informática
(Republica del Perú).

<http://www.cervantesvirtual.com/> Sitio Web de la biblioteca virtual Miguel de
Cervantes

<http://www.bibliodgsca.unam.mx> Sitio Web de la Universidad Nacional Autónoma
de México

<http://www.euoparc-es.org/> Sitio Web de EUOPARC-España, recopilación de
enlaces a diferentes bibliotecas virtuales de España

<http://bd-dl.ucol.mx/bdl/digital.htm> UNESCO, *La Biblioteca Digital*, Universidad de
Colima, 2000.

ANEXOS

ANEXOS

A.	Manual de Codificación de la CEPAL.....	II
B.	Introducción a UML.....	XI
C.	Entrevista a Bibliotecaria.....	XXIV

Anexo A.

Sistema de Información Bibliográfica de la CEPAL, Manual de Referencias

El Sistema de Información Bibliográfica de la CEPAL se constituyó en la década de los años ochenta, por iniciativa de las unidades de información que lideraron esta actividad en la CEPAL. El Centro Latinoamericano de Documentación Económica y Social (CEPAL/CLADES), el Sistema de Documentación sobre Población en América Latina (CELADE/DOCPAL), y la Biblioteca de la CEPAL, diseñaron conjuntamente, en 1981, un sistema de gestión de información bibliográfica basado en normas internacionales, que fue ampliamente difundido en la región latinoamericana y del Caribe entre instituciones relacionadas con la temática del desarrollo.

Tanto el CLADES, creado en 1971, como el DOCPAL, creado en 1976, y la Biblioteca de la CEPAL, comenzaron individualmente el trabajo de organización de la información para su almacenamiento y recuperación.

Los esfuerzos conjuntos de estas unidades dieron como resultado la creación de varias bases de datos relacionadas con el desarrollo, y un conjunto de manuales, normas de procedimiento y programas computacionales para el almacenamiento y recuperación de información que constituyen actualmente el Formato CEPAL para procesamiento de información bibliográfica.

A través de la aplicación de normas comunes para el procesamiento y recuperación de la información bibliográfica se logró estandarizar los registros de las bases de datos para posibilitar el intercambio de información entre redes de información existentes en América Latina y el Caribe. Ello permite un mejor servicio a los países miembros de la Comisión, y a los organismos e instituciones dedicados al estudio e investigación económica y social en la región latinoamericana y del Caribe.

El sistema utiliza las Reglas de Catalogación Angloamericanas para la descripción bibliográfica de los documentos que ingresan en las bases de datos y en la indización temática de la documentación procesada se utiliza el Macrothesaurus para el Procesamiento de la Información Relativa al Desarrollo Económico y Social, 5a edición. Si bien el idioma básico de trabajo es el español, se ha previsto mecanismos apropiados para facilitar el ingreso de datos y el acceso a las bases de datos en otros idiomas, y se ha desarrollado programas de conversión a los idiomas de trabajo de la Comisión (inglés, francés y portugués) que permiten generar productos de información en formatos impreso y electrónico en estos idiomas.

En los años que siguieron a la creación del sistema, los productos tradicionales fueron provistos regularmente por la CEPAL en formato impreso, y luego en diskettes y en CD-ROM. Con la introducción de Internet, todas las herramientas definidas tanto para el procesamiento de información como para la publicación y presentación de información en Internet, y la preparación de diferentes productos de información, fueron redefinidas utilizando las tecnologías de información apropiadas.

Las bases de datos bibliográficas desarrolladas por la CEPAL, que se han puesto a disposición de la comunidad internacional a través del catálogo en línea de la Biblioteca de la CEPAL en el sitio de la Comisión en Internet (<http://www.cepal.cl>), en versiones en español e inglés, son el resultado de la experiencia y el trabajo conjunto desarrollado por destacados profesionales de la información en la CEPAL en las últimas décadas.

En el diseño del Formato CEPAL, se ha considerado los siguientes aspectos:

- Adopción de las recomendaciones del Manual de Referencia de UNISIST. El propósito fundamental de esta herramienta fue el de servir como un

formato normalizado de comunicación para el intercambio de información bibliográfica entre bases de datos o redes de información al introducir la automatización en sus sistemas de información

- Consideración de las necesidades específicas, en cuanto a servicios y tipo de usuario, de las unidades de información que integraron inicialmente el sistema, y utilización de la experiencia acumulada en cada una de ellas;
- Conocimiento de la realidad latinoamericana y de su producción bibliográfica;
- Adopción de normas internacionales de descripción bibliográfica, y de herramientas normalizadas de indización.

Los aspectos antes mencionados son una guía de procedimiento para el correcto uso del Formato CEPAL sólo en sus aspectos operativos y conceptuales; las políticas de catalogación y las pautas de indización y/o de elaboración de resúmenes que se observará al registrar la información bibliográfica deben ser, sin embargo, aquéllas acordadas por cada unidad (biblioteca) de información específicamente para tales propósitos.

Conceptos básicos utilizados

Se presenta a continuación algunos conceptos básicos que se ha considerado esencial definir para la correcta comprensión y uso del Manual.

Hoja de trabajo

Las hojas de trabajo son el medio necesario para el registro e ingreso de información bibliográfica al computador. Ella contiene las diversas áreas, campo a campo, destinadas a recoger la información sobre una unidad bibliográfica.

Documento

Documento es cualquier material bibliográfico, susceptible de ser descrito en un registro bibliográfico. Dentro de este contexto documento puede ser un libro, una colección de libros, una película, video, etc.

Registro bibliográfico

Un registro bibliográfico se define como el conjunto de información pertinente a un solo documento y almacenado en una estructura lógica, única y completa. Un registro bibliográfico puede incluir la descripción bibliográfica del documento, un resumen o "abstract", la indización, y cualquier otra información que se considere de interés para el usuario. La organización del registro bibliográfico en el Formato CEPAL está basada en el Manual de Referencia del UNISIST.

Descripción bibliográfica

La descripción bibliográfica de un documento es el conjunto de información que identifica unívocamente a dicho documento. Esta información se refiere a los elementos básicos tales como autor, título y datos de edición, entre otros. Con fines de normalización el sistema utiliza las Reglas de Catalogación Angloamericanas, adecuadas a las exigencias de los programas utilizados.

Información complementaria sobre un documento

La información complementaria es todo elemento que, sin formar parte de la descripción bibliográfica o temática, se considere de interés para el usuario, permita una mejor identificación del documento, o sirva para propósitos de control interno. Esta información puede estar constituida por notas, datos sobre ubicación del documento, información sobre sus características físicas, fechas de ingreso en la base de datos, etc.

Contenido temático del documento

Además de la descripción bibliográfica, un documento es susceptible de ser descrito en relación a su contenido temático. El análisis de contenido de un documento se puede efectuar a través de la elaboración de un resumen y/o de su indización (asignación de descriptores, palabras claves o términos que reflejan el contenido temático del documento).

Elementos de datos

Elemento de datos es cada una de las partes que conforman un dato. Habitualmente se ingresa en un mismo campo y puede recuperarse en forma separada.

Campo

Es el espacio en que se consigna cada uno de los datos que conforman un registro bibliográfico. Cada campo está identificado por su respectivo número de campo o rótulo. La tabla de definición de campos proporciona una lista de todos los campos definidos en el sistema.

Ocurrencias múltiples

Un campo puede presentar ocurrencias múltiples cuando en su descripción presenta más de un elemento del mismo tipo (dos autores personales, dos o más descriptores temáticos, u otro). Estos elementos pueden repetirse tantas veces como sea necesario. Al consignar más de una ocurrencia en un mismo campo, es preciso registrarlas indicando expresamente que se trata de instancias diferentes dentro de este campo.

Tipo de literatura

Los tipos de literatura se registran en forma codificada, y pueden ser:

S publicación seriada

M monografía

V documento no convencional

Las categorías siguientes complementan la identificación de tipo de literatura:

C conferencia

P proyecto

La definición de cada una de estas categorías se presenta en forma detallada en la descripción del campo 04, Tipo de Literatura.

Nivel bibliográfico

El nivel bibliográfico indica el nivel de tratamiento que se decide dar a un documento.

Los niveles de análisis bibliográfico son:

a analítico

m monográfico

c colección

s publicación seriada

Al analizar un documento, y al ingresarlo en el sistema, sólo uno de estos niveles bibliográficos se aplica como tal. Ello, sin perjuicio de que, para su correcta identificación, se requiera agregar información relativa a otros niveles bibliográficos, para complementar información del documento para el cual el registro fue creado. Por ejemplo, un registro que describe un capítulo de un libro que a su vez es un volumen dentro de una colección, contendrá: una parte analítica por el capítulo, una parte monográfica por el volumen, y una parte colectiva por la colección. Sin embargo, se dice que este registro es de nivel analítico, porque ese nivel representa al documento para el cual el registro ha sido creado, y toda la información contenida en el registro, más allá de la que es inherente a los otros niveles citados, se va a referir exclusivamente al nivel analítico. En este caso se dice que los niveles monográficos y colección están actuando sólo como fuente para complementar información del nivel analítico.

Nivel de registro

El nivel de registro está determinando el o los niveles bibliográficos que se requiere citar en la correcta descripción bibliográfica de un documento. De acuerdo con la definición de nivel bibliográfico, en la determinación de nivel de registro se combinan los niveles bibliográficos actuando siempre uno de ellos como nivel bibliográfico en sí, y los demás como fuente.

Los niveles de registro identificados en el sistema son los siguientes:

as analítico / publicación seriada

am analítico / monográfico

amc analítico / monográfico / colección

ams analítico / monográfico / publicación seriada

amcs analítico / monográfico / colección / publicación seriada

m monográfico

mc monográfico / colección

ms monográfico / publicación seriada

mcs monográfico / colección / publicación seriada

c colección

s publicación seriada

cs colección / publicación seriada

Al margen de su funcionamiento como niveles bibliográficos en sí, los niveles monográfico, colección y publicación seriada pueden presentarse también como fuente cuando se requiere citarlos para complementar información de una entrada de nivel analítico, monográfico o colección. En tales casos, la información requerida de los niveles que actúan como fuente no es toda la información pertinente a ese nivel, sino sólo aquella que es relevante y necesaria para la correcta identificación del documento para el cual el registro fue creado.

Los elementos que es necesario registrar en los niveles monográficos, colección y publicación seriada, cuando éstos aparecen citados como fuente, son los siguientes:

Nivel monográfico: autor, título, páginas, y datos de edición

Nivel colección: autor, título, número total de volúmenes, y datos de edición

Nivel publicación seriada: título de la publicación seriada, volumen, número y fecha de publicación

En un nivel de registro en que se requiera información tanto del nivel monográfico como del nivel colección como fuentes, los datos de edición van a referirse siempre al nivel monográfico.

Tipos de registro definidos en el sistema

El tipo de registro está determinado por la combinación de tipo de literatura que representa un documento y nivel de registro que se aplicará a ese documento. Esta definición es de primordial importancia por cuanto ella determina los campos que en el registro deben contener información.

La información que se requiere consignar para la correcta descripción de un documento puede ingresar con carácter de **mandatoria**, cuando se trata de elementos como el tipo de literatura, el nivel de registro y el título --que determinan

la creación de un registro--, o con carácter de **esencial** para la correcta descripción del documento (sólo si se dispone de la información), en el caso de los autores y los datos de edición, entre otros. Toda información complementaria al documento es considerada **deseable** de ser ingresada, y deberá consignarse de acuerdo con las políticas de ingreso que determine cada unidad de información.

Tabla de definición de campos del Formato CEPAL

La tabla de definición de campos presenta todos los campos utilizados por el formato del Sistema de Información Bibliográfica de la CEPAL, su numeración y características relevantes. En la primera columna se indica el número correlativo que identifica a cada campo y en la segunda el nombre de cada uno de ellos. Al asignar la numeración de los campos se ha estimado conveniente reservar algunas posiciones para posibles expansiones del formato bibliográfico. Al definir expansiones del formato para un uso particular, no necesariamente bibliográfico, se recomienda utilizar la numeración desde el campo 101 en adelante.

FORMATO CEPAL: TABLA DE DEFINICION DE CAMPOS

Número	Nombre	Observaciones
01	Nombre del archivo	Ocurrencias múltiples
03	Ubicación	Ocurrencias múltiples
04	Tipo de literatura	
06	Nivel de registro	
10	Autor personal – nivel analítico	Ocurrencias múltiples
11	Autor institucional – nivel analítico	Ocurrencias múltiples
12	Título – nivel analítico	Ocurrencias múltiples
14	Páginas – nivel analítico	
16	Autor personal – nivel monográfico	Ocurrencias múltiples
17	Autor institucional – nivel monográfico	Ocurrencias múltiples
18	Título – nivel monográfico	Ocurrencias múltiples
20	Páginas – nivel monográfico	
23	Autor personal – nivel colección	Ocurrencias múltiples
24	Autor institucional – nivel colección	Ocurrencias múltiples
25	Título – nivel colección	Ocurrencias múltiples
27	Número de volúmenes	
29	Editor institucional de publicación seriada	Ocurrencias múltiples
30	Título de publicación seriada	Ocurrencias múltiples
31	Volumen de publicación seriada	
32	Número de publicación seriada	
33	Periodicidad de publicación seriada	
34	Existencias de publicación seriada	
35	ISSN	
38	Editorial	
39	Ciudad de la editorial	
41	Edición	
42	Información descriptiva	
43	Fecha de publicación	
44	Fecha normalizada (ISO)	
45	Símbolo (Naciones Unidas)	Ocurrencias múltiples
47	ISBN	Ocurrencias múltiples
48	Número de venta (Naciones Unidas)	Ocurrencias múltiples
52	Conferencia: institución	Ocurrencias múltiples
53	Conferencia: nombre	Ocurrencias múltiples
58	Proyecto: institución	Ocurrencias múltiples
59	Proyecto: nombre	Ocurrencias múltiples
63	Formato	Ocurrencias múltiples
64	Idioma del texto	Ocurrencias múltiples
68	Notas	
72	Resumen	
73	Referencias bibliográficas	
76	Descriptores: contenido temático	Ocurrencias múltiples
77	Descriptores: contenido estadístico	Ocurrencias múltiples
80	Categoría temática primaria	Ocurrencias múltiples
81	Categoría temática secundaria	Ocurrencias múltiples
83	Países primarios	Ocurrencias múltiples
84	Países secundarios	Ocurrencias múltiples
95	Indicador de publicación	Ocurrencias múltiples
98	Indicador de procedencia	

Para ver información más detallada ingresar al catálogo en línea de la Biblioteca de la CEPAL, se ingresa en este campo información que permite acceder a las versiones electrónicas de publicaciones de la CEPAL disponibles en texto completo en el sitio de la CEPAL en Internet <http://www.eclac.cl/biblioteca/default.htm>, Formato CEPAL <http://www.cepal.cl/biblioteca/bases.htm>, UNBIS Thesaurus <http://unhq-appspub-01.un.org/LIB/DHLUNBISThesaurus.nsf>.

Anexo B

Introducción a UML

UML [UML] es un lenguaje para especificar, construir, visualizar y documentar los componentes de un sistema de software orientado a objetos (OO).

UML se está convirtiendo en un lenguaje estándar con el que sea posible modelar todos los componentes del proceso de desarrollo de aplicaciones. Sin embargo, existe un elemento importante del modelo: no pretende definir un modelo estándar de desarrollo, sino únicamente un lenguaje de modelado. Otros métodos de modelaje como OMT (Object Modeling Technique) o Booch sí definen procesos concretos. En UML los procesos de desarrollo son diferentes según los distintos tipos de trabajo; no puede ser el mismo proceso para crear una aplicación en tiempo real, que el proceso de desarrollo de una aplicación orientada a gestión.

Modelado de objetos

El UML es una técnica de modelado de objetos y como tal supone una abstracción de un sistema para llegar a construirlo en términos concretos. El modelado no es más que la construcción de un modelo a partir de una especificación. El modelo omite detalles que no resultan esenciales para la comprensión del original y por lo tanto facilita dicha comprensión.

Los modelos se utilizan en muchas actividades de la vida humana: antes de construir una casa el arquitecto utiliza un plano, los músicos representan la música en forma de notas musicales, los artistas pintan sobre el lienzo con carboncillos antes de empezar a utilizar los óleos, etc. Unos y otros abstraen una realidad compleja sobre unos bocetos, modelos al fin y al cabo. La OMT (object-modeling technique), por ejemplo, intenta abstraer la realidad utilizando tres clases de modelos OO: el modelo de objetos, que describe la estructura estática; el modelo dinámico, con el que describe las relaciones temporales entre objetos; y el modelo funcional que describe las relaciones funcionales entre valores. Mediante estas tres fases de construcción de modelos, se consigue una abstracción de la realidad que tiene en sí misma información sobre las principales características de ésta.

Componentes para el Desarrollo de Proyectos

Un componente es una información que es utilizada o producida mediante un proceso de desarrollo de software. Pueden ser componente un modelo, una descripción o un software. Los componentes de UML se especifican en forma de diagramas, éstos, junto con la documentación sobre el sistema constituyen los componentes principales que el modelador puede observar.

Se necesita más de un punto de vista para llegar a representar un sistema. UML utiliza los diagramas gráficos para obtener estos distintos puntos de vista de un sistema:

- Diagramas de Implementación.
- Diagramas de Interacción.
- Diagramas de Casos de uso.
- Diagramas de Clases.

Diagramas de Implementación

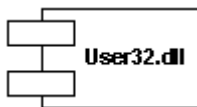
Se derivan de los diagramas de proceso y módulos de la metodología de Booch, aunque presentan algunas modificaciones. Los diagramas de implementación muestran los aspectos físicos del sistema. Incluyen la estructura del código fuente y la implementación, en tiempo de implementación. Existen dos tipos:

- **Diagramas de componentes**

Se utilizan para modelar la vista estática de un sistema. Muestra la organización y las dependencias entre un conjunto de componentes. No es necesario que un diagrama incluya todos los componentes del sistema, normalmente se realizan por partes. Cada diagrama describe un apartado del sistema.

En el se sitúan librerías, tablas archivos, ejecutables y documentos que formen parte del sistema.

Uno de los usos principales es que puede servir para ver que componentes pueden compartirse entre sistemas o entre diferentes partes de un sistema.



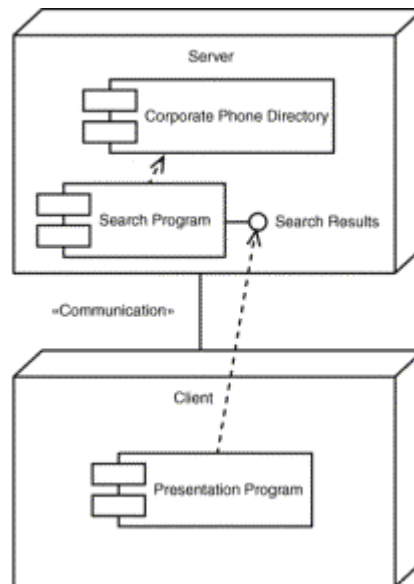
Por ejemplo, en la imagen anterior tenemos un componente del sistema de Windows. En el diagrama de componentes de Windows debe salir este componente, ya que sin el sistema no funcionaría.



En esta otra figura tenemos el mismo componente, pero indicando que dispone de una interfase. Al ser un Dll la interfase permite el acceso a su contenido. Lo anterior permite pensar que la representación anterior es incorrecta, pero no es así solo corresponde a un nivel diferente de detalle.

- **Diagrama de plataformas despliegue**

En el diagrama de despliegue se indica la situación física de los componentes lógicos desarrollados. Es decir se sitúa el software en el hardware que lo contiene. Cada Hardware se representa como un nodo.



Un nodo se representa como un cubo, un nodo es un elemento donde se ejecutan los componentes, representan el despliegue físico de estos componentes.

En la figura anterior tenemos dos nodos, el cliente y el servidor, cada uno de ellos contiene componentes. El componente del cliente utiliza una interfase de uno de los componentes del servidor. Se muestra la relación existente entre los dos Nodos. Esta relación se puede asociar a un estereotipo para indicar que tipo de conexión disponemos entre el cliente y el servidor, así como modificar su contenido, para indicar que soportamos diversos clientes.

Como los componentes pueden residir en mas de un nodo podemos situar el componente de forma independiente, sin que pertenezca a ningún nodo, y relacionarlo con los nodos en los que se sitúa.

Diagrama de Iteración

El diagrama de interacción, representa la forma en como un Cliente (Actor) u Objetos (Clases) se comunican entre si en petición a un evento. Esto implica recorrer toda la secuencia de llamadas, de donde se obtienen las responsabilidades claramente.

Muestran las interacciones entre un conjunto de objetos, ordenadas según el tiempo en que tienen lugar. En los diagramas de este tipo intervienen objetos, que tienen un significado parecido al de los objetos representados en los diagramas de colaboración, es decir son instancias concretas de una clase que participa en la interacción. El objeto puede existir sólo durante la ejecución de la interacción, se puede crear o puede ser destruido durante la ejecución de la interacción. Un diagrama de secuencia representa una forma de indicar el período durante el que un objeto está desarrollando una acción directamente o a través de un procedimiento.

En este tipo de diagramas también intervienen los mensajes, que son la forma en que se comunican los objetos: el objeto origen solicita (llama a) una operación del objeto destino. Existen distintos tipos de mensajes según cómo se producen en el tiempo: simples, síncronos, y asíncronos.

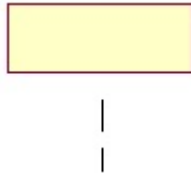
Los diagramas de secuencia permiten indicar cuál es el momento en el que se envía o se completa un mensaje mediante el tiempo de transición, que se especifica en el diagrama. Dicho diagrama puede ser obtenido de dos partes, desde el Diagrama Estático de Clases o el de Casos de Uso.

Los componentes de un diagrama de interacción son:

- Un Objeto o Actor.
- Mensaje de un objeto a otro objeto.
- Mensaje de un objeto a si mismo.

Elementos

Objeto/Actor:



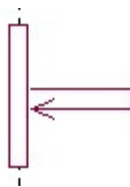
El rectángulo representa una instancia de un Objeto en particular, y la línea punteada representa las llamadas a métodos del objeto.

- **Mensaje a Otro Objeto:**



Se representa por una flecha entre un objeto y otro, representa la llamada de un método (operación) de un objeto en particular.

- **Mensaje al Mismo Objeto:**



No solo llamadas a métodos de objetos externos pueden realizarse, también es posible visualizar llamadas a métodos desde el mismo objeto.

Diagramas 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).

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

- Actor.
- Casos de Uso.
- Relaciones de Uso, Herencia y Comunicación.

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 a una persona en particular, sino más bien la labor que realiza frente al sistema.

Como ejemplo a la definición anterior, tenemos el caso de un sistema de ventas en que el rol de Vendedor con respecto al sistema puede ser realizado por un Vendedor o bien por el Jefe de Local.

- **Caso 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.

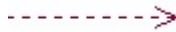
Relaciones:

- **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 Instancia**



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.

Diagramas de Clases

Un diagrama de clases sirve para visualizar las relaciones entre las clases que involucran el sistema, las cuales pueden ser asociativas, de herencia, de uso y de contenimiento.

Un diagrama de clases está compuesto por los siguientes elementos:

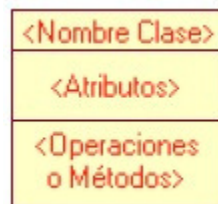
- **Clase:** atributos, métodos y visibilidad.
- **Relaciones:** Herencia, Composición, Agregación, Asociación y Uso.

Elementos

Clase

Es la unidad básica que encapsula toda la información de un Objeto (un objeto es una instancia de una clase). A través de ella podemos modelar el entorno en estudio (una Casa, un Auto, una Cuenta Corriente, etc.).

En UML, una clase es representada por un rectángulo que posee tres divisiones:




En donde:

- **Superior:** Contiene el nombre de la Clase
- **Intermedio:** Contiene los atributos (o variables de instancia) que caracterizan a la Clase (pueden ser private, protected o public).
- **Inferior:** Contiene los métodos u operaciones, los cuales son la forma como interactúa el objeto con su entorno (dependiendo de la visibilidad: private, protected o public).


Atributos y Métodos:

Atributos:

Los atributos o características de una Clase pueden ser de tres tipos, los que definen el grado de comunicación y visibilidad de ellos con el entorno, estos son:


- public (+, 

113

- private (-, 

Métodos:

Los métodos u operaciones de una clase son la forma en como ésta interactúa con su entorno, éstos pueden tener las características:




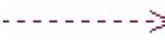
- public (+, 

Relaciones entre Clases:

Ahora ya definido el concepto de Clase, es necesario explicar como se pueden interrelacionar dos o más clases (cada uno con características y objetivos diferentes).

Antes es necesario explicar el concepto de cardinalidad de relaciones: En UML, la cardinalidad de las relaciones indica el grado y nivel de dependencia, se anotan en cada extremo de la relación y éstas pueden ser:

- Uno o muchos: 1..* (1..n)
- 0 o muchos: 0..* (0..n)
- Número fijo: m (m denota el número).

- Herencia (Especialización/Generalización)  : Indica que una subclase hereda los métodos y atributos especificados por una Super Clase, por ende la Subclase además de poseer sus propios métodos y atributos, poseerá las características y atributos visibles de la Super Clase (public y protected).
- **Agregación**  : Para modelar objetos complejos, no bastan los tipos de datos básicos que proveen los lenguajes: enteros, reales y secuencias de caracteres. Cuando se requiere componer objetos que son instancias de clases definidas por el desarrollador de la aplicación, tenemos dos posibilidades:
 - **Por Valor:** Es un tipo de relación estática, en donde el tiempo de vida del objeto incluido esta condicionado por el tiempo de vida del que lo incluye. Este tipo de relación es comúnmente llamada **Composición** (el Objeto base se construye a partir del objeto incluido, es decir, es "parte/todo").
 - **Por Referencia:** Es un tipo de relación dinámica, en donde el tiempo de vida del objeto incluido es independiente del que lo incluye. Este tipo de relación es comúnmente llamada **Agregación** (el objeto base utiliza al incluido para su funcionamiento).
- **Asociación**  : La relación entre clases conocida como Asociación, permite asociar objetos que colaboran entre si. Cabe destacar que no es una relación fuerte, es decir, el tiempo de vida de un objeto no depende del otro.
- **Dependencia o Instanciación (uso)**  : Representa un tipo de relación muy particular, en la que una clase es instanciada (su instanciación es dependiente de otro objeto/clase). Se denota por una flecha punteada. El uso más particular de este tipo de relación es para denotar la dependencia que tiene una clase de otra, como por ejemplo una aplicación grafica que

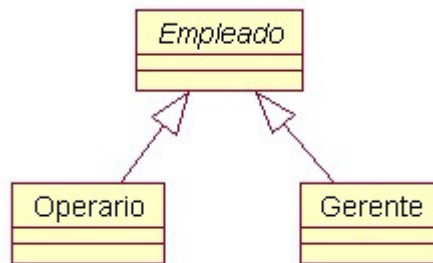
instanciar una ventana (la creación del Objeto Ventana está condicionada a la instanciación proveniente desde el objeto Aplicación):



Cabe destacar que el objeto creado (en este caso la Ventana gráfica) no se almacena dentro del objeto que lo crea (en este caso la Aplicación).

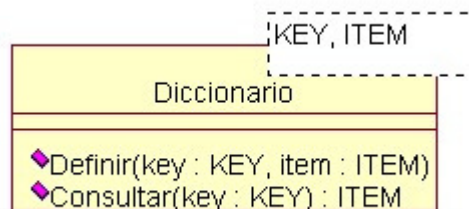
Casos Particulares:

➤ Clase Abstracta:



Una clase abstracta se denota con el nombre de la clase y de los métodos con letra "itálica". Esto indica que la clase definida no puede ser instanciada pues posee métodos abstractos (aún no han sido definidos, es decir, sin implementación). La única forma de utilizarla es definiendo subclases, que implementan los métodos abstractos definidos.

➤ Clase parametrizada:



Una clase parametrizada se denota con un sub cuadro en el extremo superior de la clase, en donde se especifican los parámetros que deben ser pasados a la clase para que esta pueda ser instanciada. El ejemplo más típico es el caso de un Diccionario en donde una llave o palabra tiene asociado un significado, pero en este caso las llaves y elementos pueden ser genéricos. La genericidad puede venir dada de un Template (como en el caso de C++) o bien de alguna estructura predefinida (especialización a través de clases).

En el ejemplo no se especificaron los atributos del Diccionario, pues ellos dependerán exclusivamente de la implementación que se le quiera dar.

Anexo C

Entrevista a bibliotecaria

Objetivo: Realizar una entrevista a la persona encargada de la biblioteca de la casa de la cultura, con el fin de entender el proceso de funcionamiento actual de la biblioteca.

Entrevistado:

Bibliotecaria: Ana Jazmín Aguilar Morales.

¿Cuántas personas utilizan la biblioteca actualmente?

R/

Desde el apogeo del Internet, la gente ha dejado de utilizar las bibliotecas, hace unos 10 años venían alrededor de 20 personas diarias, pero ahora la cantidad ha disminuido de manera significativa, ahora con suerte vienen dos personas al día y hay días que no viene nadie.

¿Cuál es la edad de las personas que frecuentan la biblioteca?

R/

La edad varía, pero el promedio anda por los 12 años, los jóvenes vienen cuando buscan información específica cultural e histórica del país, ya que esta clase de material no lo encuentran con la misma disponibilidad que otros materiales en Internet.

¿Con qué tipo de materiales cuenta la biblioteca?

R/

La biblioteca cuenta con recursos bibliográficos, libros, revistas, enciclopedias, ya en estos podemos hacer diferencias, ya que estos divididos por áreas, similares a las materias del colegio, matemáticas, lenguajes, sociales, ciencias naturales,

todo esto con el fin de que los jóvenes puedan encontrar de forma mas inmediata los materiales.

¿Cuál es el proceso que debe seguir una persona, para poder prestar materiales?

R/

El visitante debe ser una persona con carné, este cuesta 75 centavos de dólar, y le permite tener acceso a los materiales de la biblioteca por un año, el uso de los libros es de uso interno, debido a que las personas no regresan los materiales si se les presta por más tiempo.

¿Cuál es el proceso que se sigue, cuando el usuario visita la biblioteca para prestar materiales?

R/

Las personas hablan directamente conmigo, normalmente me dicen el nombre del libro y yo de acuerdo a mi experiencia busco los materiales en los estantes, existen fichas, pero las personas no pueden utilizarlas.

¿Cuántas personas están encargadas de la biblioteca?

R/

La directora de la biblioteca y yo, pero la que trabaja con las personas soy yo. El horario es de lunes a sábado de 8-12 y de 2-5.

¿Cuál es el equipo de oficina con el que cuentan?

R/

Una maquina de escribir y un teléfono.

¿Cómo se lleva el control de inventario de los libros?

R/

Usamos fichas, para llevar un control de los libros con los que contamos. Pero normalmente no son usados para generar reportes para nadie, por la dificultad que nos generaría hacer eso.