

**UNIVERSIDAD DEL BIO-BIO**

**FACULTAD CIENCIAS EMPRESARIALES**

**DEPARTAMENTO DE SISTEMAS DE INFORMACIÓN**

**DESARROLLO DE SISTEMA ORIENTADO A OBJETOS  
PARA LA BIBLIOTECA DEL SEMINARIO  
METROPOLITANO DE CONCEPCIÓN**

AUTOR: CLAUDIO ULLOA MERINO  
PROFESOR GUÍA: PEDRO CAMPOS SOTO

**2007**

*Con mucho cariño a mis Padres.*

## AGRADECIMIENTOS

Al Señor que día a día me permite abrir los ojos, a mi Padre Martin por el cariño, apoyo y posibilidad que dio de estudiar, A mi Madre Miriam por su amor y cariño, a mis hermanos (Cristian, Katherine y Martincito) y a mi polola Johanna, por su comprensión, apoyo y amor incondicional en todo mi proceso Universitario.

# ÍNDICE

ÍNDICE.....	3
TABLA DE ILUSTRACIONES .....	7
ANEXOS .....	1
CAPÍTULO I: COMIENZO DEL PROYECTO .....	2
1.1 POR QUÉ EL SISTEMA .....	3
PROBLEMA .....	3
PROPUESTA DE SOLUCIÓN.....	4
ÁMBITO DE LA APLICACIÓN .....	7
JUSTIFICACIÓN DEL PROYECTO .....	7
1.2 POR QUÉ ORIENTACIÓN A OBJETOS.....	8
1.3 OBJETIVOS DEL PROYECTO .....	10
OBJETIVO GENERAL DEL PROYECTO.....	10
OBJETIVO ESPECÍFICOS DEL PROYECTO .....	11
1.4 PRESENTACIÓN DE CAPÍTULOS.....	12
CAPÍTULO II: MARCO TEÓRICO.....	14
2.1 CARACTERÍSTICAS DE LA OO .....	15
CONCEPTOS FUNDAMENTALES.....	15
CARACTERÍSTICAS DE LA POO .....	17
BENEFICIOS DE LA ORIENTACIÓN A OBJETOS .....	18
2.2 PROGRAMACIÓN CON JAVA Y JSP .....	19
CARACTERÍSTICAS .....	19
COMPONENTES CLAVES DE LAS JSP'S.....	21
FUNCIONAMIENTOS JSP .....	22
VENTAJAS JSP .....	22
2.3 BASE DE DATOS ORIENTADA A OBJETOS .....	23
HISTORIA DE LAS BD ORIENTADAS A OBJETOS .....	24
CRÍTICAS .....	24
CARACTERÍSTICAS .....	25

EJEMPLO DE BDOO .....	25
2.4 DOCUMENTACIÓN DE UML.....	26
ARQUITECTURA del UML .....	27
DIAGRAMAS .....	27
2.5 CÓDIGOS DE BARRA .....	29
NOMENCLATURA BÁSICA:.....	30
ESTRUCTURA.....	30
2.6 SISTEMA DE CLASIFICACIÓN DECIMAL DEWEY.....	32
HISTORIA.....	32
PRINCIPIOS GENERALES .....	33
BENEFICIOS .....	34
NOTACIÓN .....	34
2.7 TECNOLOGÍA AJAX .....	36
2.8 METODOLOGÍA A UTILIZAR .....	38
CAPÍTULO III: DESCRIPCIÓN SEMINARIO.....	40
3.1 RESEÑA HISTÓRICA .....	41
3.2 QUE ES UN SEMINARIO .....	42
3.3 DEFINICIÓN DEL SEMINARIO .....	42
VISIÓN.....	44
MISIÓN .....	44
OBJETIVOS ESTRATÉGICOS .....	44
ESTRUCTURA ORGANIZACIONAL.....	45
DESCRIPCIÓN DE LA BIBLIOTECA .....	47
CAPÍTULO IV: ESTUDIO DE FACTIBILIDAD.....	48
4.1 FACTIBILIDAD TÉCNICA.....	49
4.2 FACTIBILIDAD ECONÓMICA .....	53
FUNCIONAMIENTO SISTEMA ACTUAL.....	56
FUNCIONAMIENTO NUEVO SISTEMA.....	58
4.3 FACTIBILIDAD OPERACIONAL .....	60
4.4 CONCLUSIÓN .....	61
CAPÍTULO V: ANÁLISIS .....	62
5.1 DESCRIPCIÓN GLOBAL DEL PRODUCTO .....	63
OBJETIVO DEL SOFTWARE.....	63

OBJETIVOS ESPECÍFICOS DEL SOFTWARE .....	63
ALCANCES .....	64
INTERFAZ DE HARDWARE .....	65
5.2 PRINCIPALES ACTIVIDADES .....	66
REGISTRO Y CLASIFICACIÓN DE LOS LIBROS Y REVISTAS .....	66
PRÉSTAMOS DE LIBROS Y REVISTAS .....	67
DEVOLUCIÓN DE LIBROS Y REVISTAS .....	68
BÚSQUEDA DE MATERIAL .....	69
5.3 REQUERIMIENTOS ESPECÍFICOS .....	71
REQUERIMIENTOS FUNCIONALES DEL SISTEMA .....	71
DESCRIPCIÓN MEDIANTE CASOS DE USO .....	72
DIAGRAMA DE PAQUETES .....	73
CAPITULO VI: DISEÑO .....	115
6.1 DISEÑO ARQUITECTÓNICO .....	116
6.2 DIAGRAMAS DE COLABORACIÓN .....	118
6.4 JERARQUÍA DE MENÚ .....	137
APLICACIÓN DE ESCRITORIO .....	137
APLICACIÓN DE WEB .....	139
6.5 PANTALLAS DEL SISTEMA .....	141
CAPITULO VII: DETALLES DE IMPLEMENTACIÓN .....	146
7.1 CONTROL DE CONCURRENCIA EN DB40 .....	147
7.2 IMPLEMENTACIÓN CON AJAX .....	149
7.3 DETALLES DE DISEÑO CON SWING EN JAVA .....	152
JTABLE .....	152
JDIALOG y JFRAME .....	153
7.4 CÓDIGOS DE BARRA .....	155
IMPRESORA DE CÓDIGOS DE BARRA .....	155
LECTOR DE CÓDIGOS DE BARRA .....	157
CAPITULO VIII: PLAN DE PRUEBA, PUESTA EN MARCHA Y CAPACITACIÓN .....	158
8.1 PLAN DE PRUEBAS .....	159
8.2 PUESTA EN MARCHA .....	168
8.3 PLAN DE CAPACITACIÓN .....	170
CONCLUSIONES .....	172

BIBLIOGRAFÍA.....	175
-------------------	-----

# TABLA DE ILUSTRACIONES

Figura 1: Arquitectura Hardware de la propuesta de solución.....	5
Figura 2: Arquitectura del software de la propuesta de solución.....	5
Figura 3: Base de datos Orientada a Objetos con sus respectivos estados.....	24
Figura 4: Diagramas UML .....	28
Figura 5: Código de barras Lineal.....	29
Figura 6: Código de barras bidimensional.....	30
Figura 7: Estructura códigos de barra.....	31
Figura 8: Tecnologías agrupadas bajo el concepto de AJAX.....	36
Figura 9: Grafica el modelo Incremental.....	38
Figura 10: Estructura Organizacional del Seminario Metropolitano Concepción. ....	45
Figura 11: Registro y calificación de libros y revistas. ....	66
Figura 12: Préstamos de libros y revistas. ....	67
Figura 13: Devolución de libros y revistas.....	68
Figura 14: Búsqueda de Material.....	69
Figura 15: Paquetes para casos de uso. ....	73
Figura 16: Paquete catálogo libros y revistas.....	75
Figura 17: Modelo conceptual paquete Catálogo libros y revistas.....	90
Figura 18: Paquete Administración de usuarios.....	91
Figura 19: Modelo conceptual del paquete Administración de Usuarios.....	99
Figura 20: Paquete préstamos y devoluciones.....	100
Figura 21: Modelo conceptual del paquete Préstamos y devoluciones .....	108
Figura 22: Paquete Publicaciones digitales. ....	109
Figura 23: Modelo conceptual del paquete Publicaciones Digitales.....	110
Figura 24: Paquete sistema de mensajería.....	111
Figura 25: Modelo conceptual del paquete sistema de mensajería. ....	112
Figura 26: Paquete recomendación de búsqueda.....	113
Figura 27: Modelo conceptual del paquete recomendación de búsqueda. ....	114
Figura 28: Diseño Arquitectónico .....	116

Figura 29: Instancia en un diagrama de Colaboración.....	119
Figura 30: Ejemplo métodos get y set .....	119
Figura 31: Método setBD ocupado para guardar un objeto.....	120
Figura 32: Ejemplo setBD(). .....	120
Figura 33: Ejemplo consulta a la base de datos.....	121
Figura 34: Método BuscarPorTitulo de la clase Libro.....	121
Figura 35: Diagrama de clases, paquete catálogo de libros y revistas. ....	132
Figura 36: Diagrama de clases, paquete administración de usuarios. ....	133
Figura 37: Diagrama de clases, paquete préstamos y devoluciones.....	134
Figura 38: Diagrama de clases, paquete publicaciones digitales.....	135
Figura 39: Diagrama de clases, paquete sistema de mensajería. ....	135
Figura 41: Código con semáforos de DB4o. ....	147
Figura 42: Función que ocupa el sistema para recargar la Web.....	149
Figura 43: función objetoAjax().....	151
Figura 44: Método reescrito en el modelo de JTable. ....	152
Figura 45: Pruebas de los incrementos a las de integración. ....	159
Figura 46: Tiempo marcha blanca.....	169
Figura 47: Diagrama de paquetes. ....	251
Figura 48: Paquete catálogo libros y revistas. ....	252
Figura 49: Modelo conceptual paquete Catálogo libros y revistas.....	270
Figura 50: Paquete Administración de usuarios.....	271
Figura 51: Modelo conceptual del paquete Administración de Usuarios.....	287
Figura 53: Modelo conceptual del paquete Préstamos y devoluciones .....	299
Figura 54: Paquete Publicaciones digitales. ....	300
Figura 55: Modelo conceptual del paquete Publicaciones Digitales .....	309
Figura 56: Paquete sistema de mensajería.....	310
Figura 57: Modelo conceptual del paquete Publicaciones Digitales .....	315
Figura 58: Paquete recomendación de búsqueda.....	316
Figura 59: Modelo conceptual del paquete Recomendación de búsqueda.....	321

# ANEXOS

ANEXO 1 .....	178
MANUAL DE USUARIO DEL SISTEMA BIBLIOTECARIO .....	178
ANEXO 2 .....	249
CASOS DE USO Y DIAGRAMAS CONCEPTUALES DE TODO EL SISTEMA .....	249
ANEXO 3 .....	322
DOCUMENTO DE APOYO DB4O .....	322
ANEXO 4 .....	334
DOCUMENTO DE APOYO JSP .....	334
ANEXO 5 .....	343
DOCUMENTO DE APOYO NETBEANS.....	343

# CAPÍTULO I: COMIENZO DEL PROYECTO

“El desarrollo de un software que permita la automatización de las actividades relacionadas a la biblioteca del Seminario Metropolitano Concepción implementado en un 100% con tecnología orientada a objetos”

En este capítulo se abordará el problema que se origina en la biblioteca, una propuesta de solución, el porqué utilizar la tecnología orientada a objetos y finalmente una presentación de los capítulos posteriores.

## **1.1 POR QUÉ EL SISTEMA**

El “Desarrollo de sistema Orientado a Objetos para la Biblioteca del Seminario Metropolitano Concepción”, es un proyecto que busca dar una solución automatizada a las actividades que se realizan en la biblioteca del Seminario. Este sistema tiene la particularidad de ser totalmente orientado a objetos, y permitirá de una forma eficiente manejar todos los objetos que existen actualmente en la biblioteca.

## **PROBLEMA**

El Seminario Metropolitano Concepción, es una casa de estudios que se dedica a la formación de futuros sacerdotes. Un factor para el éxito de la institución, es la existencia de la biblioteca, cuya finalidad consiste en la adquisición, conservación, estudio y exposición de libros y documentos. En el Seminario, existe una biblioteca exclusiva para que los seminaristas puedan realizar sus estudios, ésta posee alrededor de 11.000 libros clasificados, pero debido al tiempo que requiere ingresar y clasificar el material, aún quedan unos 40.000 ejemplares que no han sido clasificados.

**A continuación se mencionan los problemas que existen actualmente en la biblioteca del Seminario.**

- **Préstamos de libros y revistas:** Los libros y revistas poseen una tarjeta que los identifica, es allí donde se registra el día correspondiente a la devolución y el nombre del usuario que solicitó el material. Este proceso se hace de forma manual, siendo muy difícil el control de la información, debido a esto el proceso se vuelve lento y tedioso.
- **Devolución de libros y revistas:** Consiste en dejar el material solicitado al bibliotecario, ya que es él quien vuelve a dejarlo en el lugar correspondiente. Al existir un atraso de un libro o revista, se hace difícil el registro de la falta, ya que no existen fichas de usuarios a los que se les pueda vincular.

- **Registro y clasificación de libros revistas:** Es sin duda el proceso más complicado de la biblioteca, ya que para registrarlos, se debe ingresar una serie de datos en una hoja especial, si es un libro se clasifica el material con el *sistema de Clasificación Decimal Dewey*, después de esto se crean una serie de tarjetas que permiten almacenar el autor, título, y materia. El proceso completo es bastante lento e ineficiente, ya que la mayoría de la información queda almacenada en los estantes y no es utilizada de una forma óptima.
- **Búsqueda de libros y revistas:** Para la búsqueda de libros, el seminarista o bibliotecario, debe revisar las tarjetas que se encuentran en los estantes, los criterios de búsqueda son: autor, título y materia. Una vez que el usuario encuentra lo que busca, debe anotar el código de clasificación, ya que con este código se puede ubicar fácilmente el material. Este proceso es aburrido y demasiado lento, ya que buscar entre unas 30.000 tarjetas es muy complejo. La búsqueda de una revista es un tanto complicada, ya que no existe un método que apoye ésta labor, las revistas son dejadas en los estantes por orden alfabético y de buscar algo específico, sólo queda leerlas una por una.

## **PROPUESTA DE SOLUCIÓN**

La propuesta de solución a la problemática de la biblioteca del Seminario, es realizar un sistema bibliotecario orientado a objetos que realice las operaciones básicas y además incorpore alternativas innovadoras para la búsqueda de los materiales existentes.

### **El sistema constará de 2 programas fundamentales.**

- **Administración de los objetos de la biblioteca:** Un programa de escritorio que ejecute las operaciones administrativas correspondientes al bibliotecario.
- **Plataforma Web:** Plataforma que usarán los usuarios para realizar la búsqueda del material existente y conocer la información relacionada al usuario, como préstamos vigentes y préstamos históricos.

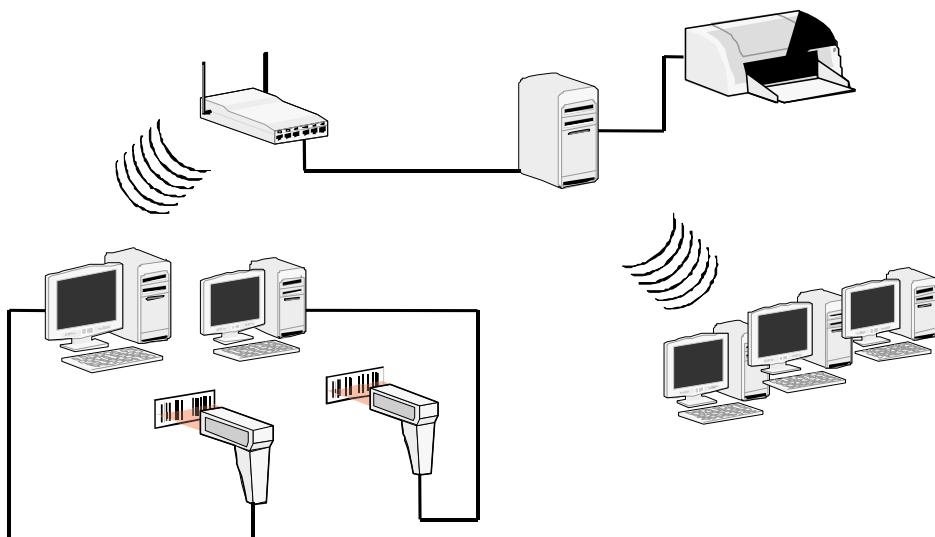


Figura 1: Arquitectura Hardware de la propuesta de solución.

En la Figura 1 se muestra la interacción del hardware con los dos programas que el sistema proveerá. La aplicación de escritorio (para la administración de objetos de la biblioteca) es la que permitirá al bibliotecario ingresar los libros y revistas, registrar a los usuarios y permitir los préstamos y devoluciones mediante códigos de barra. La aplicación Web por su parte es la que permitirá a los usuarios realizar la búsqueda de libros y revistas, además de conocer el detalle de los préstamos vigentes e históricos. La comunicación de las aplicaciones mencionadas, se efectuará por medio de una conexión inalámbrica que iluminara todo el sector físico de la biblioteca, permitiendo el acceso libre de cualquier usuario a la plataforma Web.

**A continuación se presenta arquitectura del software correspondiente al del sistema.**



Figura 2: Arquitectura del software de la propuesta de solución.

En la Figura 2 se pueden apreciar bloques que representan diversos subsistemas y el flujo de información que existe entre ellos, presentando una vista abstracta de todo el sistema. La arquitectura muestra que los subsistemas “Administración de usuarios” y “Catálogo libros y revistas” son totalmente independientes de otros subsistemas, en cambio “Sistema Mensajería” y “Publicaciones digitales” dependen de “Administración de usuarios”, por otro lado “Recomendación de búsqueda” y “Devolución y préstamos” dependen de “Administración de usuarios” y “Catálogo libros y revistas”. En el siguiente párrafo se menciona la funcionalidad de cada uno:

- **Administración de usuarios:** Este subsistema se encarga de administrar y proveer toda la información correspondiente a los usuarios de la biblioteca, permitiendo el ingreso, modificación y búsqueda de la información perteneciente a cada usuario.
- **Catálogo de libros y revistas:** Este subsistema tiene el control de los libros y revistas que pertenecen a la biblioteca, permitiendo el ingreso y modificación de información válida para el sistema, también es importante para éste subsistema el controlar la búsqueda de un libro o revista.
- **Devolución y préstamos:** Subsistema que tiene por objetivo, interconectar información entre los subsistemas “Catálogo de libros y revistas” y “Administración de usuarios” para lograr que un préstamo sea asociado a un material y a un usuario en particular. Este subsistema también es el encargado de registrar la devolución del material.
- **Publicaciones digitales:** Subsistema que permite a un seminarista almacenar documentos digitales vía Web con el objetivo de que todos los seminaristas tengan acceso a él.

*Subsistema que solo existe en la plataforma Web.*

- **Recomendación de búsqueda:** Subsistema que tiene una estrecha relación con “Catálogo de libros y revistas”, ya que ordenará los resultados de una búsqueda de libros de acuerdo a las recomendaciones que los mismos seminaristas han realizado.

*Subsistema que solo existe en la plataforma Web.*

- **Sistema de mensajería:** subsistema que permite la comunicación entre seminaristas y bibliotecarios a través de la cuenta que cada uno posee.

## ÁMBITO DE LA APLICACIÓN

El Sistema Orientado a Objetos para Biblioteca del Seminario Concepción, será aplicado en la Biblioteca existente en el Seminario, este sistema reunirá información de: Seminaristas, Libros, Revistas, Publicaciones digitales, préstamos y devoluciones. Esta información será ingresada por medio del propio sistema, ya que no existe un sistema actual con el cual pueda interactuar.

La información al ser relacionada entre sí, dará origen a un sistema de biblioteca, el cual permitirá el control y gestión de la información referente a los préstamos y devoluciones tanto de libros y revistas, además cada usuario podrá publicar documentos digitales, enviar menajes y retroalimentar recomendaciones para facilitar la búsqueda de los libros.

## JUSTIFICACIÓN DEL PROYECTO

El proyecto “Desarrollo de sistema Orientado a Objetos para la Biblioteca del Seminario Metropolitano Concepción” se justifica ampliamente, ya que en la actualidad la biblioteca del Seminario opera totalmente de forma manual, haciendo que las actividades se realicen de forma lenta e inefficiente, por esto mismo, la productividad de la biblioteca se ve bastante empobrecida y limitada.

Una de las actividades más importantes que se realiza dentro de la biblioteca, es el registro y clasificación del material que existe, ésta actividad permite que los libros puedan ser utilizados por los usuarios. Lamentablemente existen demasiados libros que no pueden ser utilizados, debido a la falta de tiempo existente para su clasificación. El nuevo sistema permitirá que se ingresen y clasifiquen más libros en un tiempo muy corto, ya que bastará ingresar los datos del material a un formulario digital para que estos queden registrados.

Para los seminaristas, la búsqueda de libros es tediosa y muchas veces incierta, ya que ésta es a través de tres tipos de tarjetas, unas contienen el título del libro, otras el autor y las últimas las materias respectivas. La búsqueda sobre tarjetas, se hace muy compleja debido a la gran cantidad

de libros que existen. El sistema permitirá una búsqueda inmediata y exacta, lo que hará que el seminarista pueda encontrar realmente lo que está buscando.

Por último, el proceso de préstamos y devolución del material, deja mucho que desear, ya que prácticamente no existe un control sobre ésta información. El nuevo sistema permitirá un control tanto de los usuarios, como del material vinculado a los préstamos y devoluciones.

## 1.2 POR QUÉ ORIENTACIÓN A OBJETOS

El uso de la tecnología orientada a objetos será fundamental para el proyecto en cuestión. A continuación, se mencionan las razones por la cual se utilizará este paradigma.

La orientación a objetos, facilita la creación de *software de calidad* por sus factores que potencian el *mantenimiento*, la *extensión* y la *reutilización* del software generado bajo este paradigma.

La programación orientada a objetos trata de amoldarse al *modo de pensar del hombre* y no al modo de operar de las máquinas. Esto es posible gracias a la forma racional con la que se manejan las abstracciones que representan las entidades del dominio del problema, y a propiedades como la jerarquía o el encapsulamiento.

El elemento básico de este paradigma no es la función (elemento básico de la programación estructurada), sino un ente denominado objeto. Un objeto es la representación de un concepto para un programa, y contiene toda la información necesaria para abstraer dicho concepto: los datos describen su estado y las operaciones pueden modificar dicho estado determinando las capacidades del objeto.

Los beneficios de la tecnología orientada a objeto se fortalecen si se usa antes y durante el proceso de ingeniería del software. Un simple uso de programación orientada a objetos (POO) no brindará los mejores resultados. Los ingenieros del software y sus directores deben considerar tales elementos: el análisis de requisitos orientado a objetos (AROO), el diseño orientado a objetos (DOO), el análisis de dominio orientado a objetos (ADOO), sistema de gestión de bases de datos orientados a objetos (SGBDOO) y la ingeniería del software orientado a objetos asistida por computadora (ISOOAC.) (Pressman, 2002)

Las Técnicas Orientadas a Objetos modifican el punto de vista de los analistas de sistemas de información acerca del mundo, en vez de pensar en los procesos y su descomposición, piensan en Objetos y su comportamiento (Martin, y otros, 1994).

Además, el enfoque Orientado a objetos es ambicioso: abarca todo el ciclo de vida de desarrollo software. Cuando se examinan soluciones orientadas a objetos se debe comprobar que el método y el lenguaje, así como las herramientas de apoyo, sean aplicables al análisis y al diseño al igual que la implementación y el mantenimiento (Meyer, 1998).

Estas características hacen de la Orientación a Objetos la mejor opción a considerar en la implementación del sistema de biblioteca del Seminario Metropolitano Concepción, permitiendo la reutilización del código en futuras aplicaciones y una rápida ampliación del sistema.

## **1.3 OBJETIVOS DEL PROYECTO**

### **OBJETIVO GENERAL DEL PROYECTO**

El objetivo general del proyecto es el desarrollo de un software para la administración del Seminario Metropolitano Concepción, con la característica distintiva de ser desarrollado íntegramente con “*Tecnología orientada a objetos*”, vale decir, un análisis, diseño, codificación y utilización de una base de datos apropiada (orientada a objetos).

El propósito central del proyecto “Sistema Orientado a Objetos para Biblioteca del Seminario Metropolitano Concepción”, es la automatización de los procesos relacionados a la administración y control de los recursos internos de la biblioteca (Libros, revistas, publicaciones digitales).

El uso de tecnología también es parte del objetivo, ya que se utilizará para dar una mayor eficacia al proceso en general, gracias a las ventajas propias del paradigma, ya sea mantención y la posibilidad de reutilización.

En resumen, el objetivo del proyecto se puede definir en: “La automatización de los procesos relacionados a la Biblioteca del Seminario Metropolitano Concepción haciendo uso de la tecnología Orientada a Objetos”.

## **OBJETIVO ESPECÍFICOS DEL PROYECTO**

Para el cumplimiento del objetivo general mencionado anteriormente, se plantean los siguientes objetivos específicos:

- Estudio detallado de los requerimientos de una biblioteca, específicamente, la del Seminario Metropolitano Concepción.
- Estudio a fondo del Paradigma Orientado a Objetos, dando énfasis en las bases de datos y el uso de UML para su documentación.
- Estudio del sistema de Clasificación Decimal Dewey (CDD) y un sistema de codificación mediante códigos de barra para identificar el material bibliográfico.
- Implementar nuevas formas de búsqueda, apoyadas por recomendaciones correspondientes a evaluaciones y comentarios de personas que han utilizado el material.
- Implementar terminales clientes que permitan atender a las consultas de usuarios, utilizando una red local para la comunicación con los servidores.
- El sistema debe ser diseñado para permitir el acceso de otros seminarios.
- Realizar un diseño que permita la reutilización del código construido, en otros proyectos.

## **1.4 PRESENTACIÓN DE CAPÍTULOS**

El Proyecto “Sistema Orientado a Objetos para Biblioteca del Seminario Concepción” es presentado en ocho capítulos, los cuales van desde una introducción, hasta los planes de capacitación que los usuarios necesitan para utilizar el software.

En este capítulo introductorio se comienzan a detallar los primeros pasos del proyecto, dando respuesta a interrogantes como ¿Por qué realizar el sistema?, ¿A quién va dirigido?, ¿Qué es lo que soluciona? , ¿Cómo lo soluciona? y ¿Por qué se utilizará la tecnología a objetos?

Luego, en el segundo capítulo, se dan a conocer teorías que son necesarias para el entendimiento de los capítulos posteriores. El marco teórico se centra en : Características de la Orientación a Objetos, JAVA, JSP, bases de datos orientadas a objetos, UML, códigos de barra, sistema de clasificación Dewey, tecnología Ajax y la metodología utilizada para el desarrollo del software.

Más tarde, en el tercer capítulo, se da una definición clara de que es un Seminario, una descripción específica de la organización, cuales son los objetivos y finalmente se detalla el lugar específico donde se implementará el proyecto, “La Biblioteca del Seminario”.

En el cuarto capítulo, se documenta el estudio de factibilidad realizado para el proyecto, pasando por los estudios de factibilidad técnica, económica y operacional.

El análisis, es parte del quinto capítulo, y en él se encuentran los objetivos que el software posee, los alcances, las principales actividades que se realizan (diagramas de actividad) y los requerimientos del sistema. Posteriormente, se describen los requerimientos mediante los principales casos de uso y diagramas conceptuales de cada paquete.

En el sexto capítulo, se llevará a cabo el diseño del software, en el cual se define, el diseño arquitectónico, el diseño de implementación (principales diagramas de colaboración), los diagramas de clases de cada paquete, la jerarquía de menús y finalmente el diseño de algunas interfaces.

Después del diseño, en la codificación se presentan detalles que son interesantes conocer, como la utilización de códigos de barra en el software, el manejo de la concurrencia en una base de datos orientada a objetos, el uso de la tecnología Ajax y el uso de componentes Swing.

Finalmente, en el último capítulo, se mencionan los principales casos de prueba, la puesta en marcha (marcha blanca) y la capacitación necesaria que los usuarios necesitan para la utilización de sistema.

## CAPÍTULO II: MARCO TEÓRICO

En este segundo capítulo denominado “*Marco teórico*” se definen materias importantes que se trabajarán en el proyecto. Como el título lo menciona, se describirán las teorías que apoyan las decisiones que se han tomado en este proyecto.

- Características de la Orientación a objetos.
- La tecnología JAVA y JSP.
- Base de datos orientada a objetos.
- UML.
- Códigos de barras.
- Sistema de clasificación Dewey.
- Tecnología Ajax.
- Metodología incremental.

## 2.1 CARACTERÍSTICAS DE LA OO

Como ya se mencionó en el objetivo, la orientación a objetos (OO) es la principal cualidad de este proyecto. A continuación se describen las características de este paradigma.

*“Las técnicas orientadas a objetos permite que el software se construya a partir de objetos de comportamiento específico. Los propios objetos se pueden construir a partir de otros, que a su vez pueden estar formados por otros objetos” (Martin, y otros, 1994).*

## CONCEPTOS FUNDAMENTALES

La programación orientada a objetos introduce nuevos conceptos que superan y amplían conceptos antiguos ya conocidos. Entre ellos destacan los siguientes (Wikipedia POO, 2007):

- **Objeto:** Entidad provista de un conjunto de propiedades (atributo o dato) y de comportamiento (métodos). Corresponden a los objetos reales del mundo que nos rodea, o a objetos internos del sistema.
- **Clase:** Definiciones de las propiedades y comportamientos de un tipo de objeto. La instanciación es la lectura de estas definiciones y la creación de un objeto a partir de ellas.
- **Método:** Algoritmo asociado a un objeto (o a una clase), cuya ejecución se desencadena tras la recepción de un *mensaje*. Desde el punto de vista del comportamiento, es lo que el objeto puede hacer. Un método puede producir un cambio en las propiedades del objeto o la generación de un *evento* con un nuevo mensaje para otro objeto del sistema.
- **Evento:** Un suceso en el sistema (tal como una interacción del usuario con la máquina, o un mensaje enviado por un objeto). El sistema maneja el evento enviando el mensaje adecuado al objeto pertinente. También se puede definir como evento, a la reacción que puede desencadenar un objeto, es decir, la acción que genera.
- **Mensaje:** Una comunicación dirigida a un objeto, que le ordena que ejecute uno de sus métodos con ciertos parámetros asociados al evento que lo generó.

- **Propiedad o atributo:** Contenedor de un tipo de datos asociados a un objeto (o a una clase de objetos), que hace los datos visibles desde fuera del objeto, y cuyo valor puede ser alterado por la ejecución de algún método.
- **Estado interno:** Es una propiedad invisible de los objetos, que puede ser únicamente accedida y alterada por un método del objeto, y que se utiliza para indicar distintas situaciones posibles para el objeto (o clase de objetos).
- **Componentes de un objeto:** Atributos, identidad, relaciones y métodos.
- **Representación de un objeto:** Un objeto se representa por medio de una tabla o entidad que esté compuesta por sus atributos y funciones correspondientes.
- **Tipo:** Conjunto de firma de métodos con un nombre que lo identifica. Un tipo puede ser definido a través de una clase o una interface.

## CARACTERÍSTICAS DE LA POO

Las características esenciales del paradigma orientado a objetos son las que se describen a continuación (Wikipedia POO, 2007):

- **Abstracción:** Capacidad de analizar y representar las características esenciales de fenómenos complejos.
- **Encapsulamiento:** Técnica para proteger y ocultar el estado interno y *conocimiento* de una entidad.
- **Polimorfismo:** Comportamientos diferentes, asociados a objetos distintos, pueden compartir el mismo nombre, al llamarlos por ese nombre se utilizará el comportamiento correspondiente al objeto que se esté usando.
- **Herencia:** Las clases no están aisladas, sino que se relacionan entre sí, formando una jerarquía de clasificación. Los objetos heredan las propiedades y el comportamiento de todas las clases a las que pertenecen. La herencia organiza y facilita el polimorfismo y el encapsulamiento permitiendo a los objetos ser definidos y creados como tipos especializados de objetos preexistentes.

## BENEFICIOS DE LA ORIENTACIÓN A OBJETOS

A continuación se listan los beneficios que la tecnología orientada a objetos ofrece al proyecto en cuestión (Martin, y otros, 1994).

- **Reutilización:** Las clases están diseñadas para que se reutilicen en muchos sistemas. Para maximizar la reutilización, las clases se construyen de modo que se puedan adaptar. Es probable que las bibliotecas de clases crezcan rápidamente.
- **Estabilidad:** Las clases diseñadas para una reutilización repetida se vuelven estables, de la misma manera que los microprocesadores y otros chips se hacen estables.
- **Confiabilidad:** Es probable que el software construido a partir de las clases estables ya probadas tengan menos fallas que el software elaborado a partir de cero.
- **Diseño más rápido:** Las aplicaciones se crean a través de componentes ya existentes. Muchos de los componentes están construidos de modo que se puedan adaptar para un diseño particular. Los componentes se pueden ver, adaptar y ligar entre sí en la pantalla de herramientas CASE.
- **Diseño de mayor calidad:** Los diseños suelen tener mayor calidad, puesto que se integran a partir de componentes probados, que han sido verificados y pulidos varias veces.
- **Programación más sencilla:** Los programas se unen a partir de piezas pequeñas, cada una de las cuales, en general, se pueden crear fácilmente. El programador crea un método para una clase a la vez.
- **Mantenimiento más sencillo:** El programador encargado del mantenimiento cambia un método de clase a la vez. Cada clase efectúa sus funciones independientemente de los demás.
- **Independencia del diseño:** Las clases están diseñadas para ser independientes del ambiente de plataformas, hardware y software. Utilizan solicitudes y respuestas con formato estándar. Esto les permite ser utilizadas en múltiples sistemas operativos, controladores de base de datos, controladores de redes, interfaces Usuario-Gráficas, etc.

- **Modelado más realista:** El análisis OO modela la empresa o área de aplicación de manera que sea lo más cercana posible a la realidad de lo que se logra con el análisis convencional. El análisis se traduce de manera directa en el diseño y la implantación.

## 2.2 PROGRAMACIÓN CON JAVA Y JSP

El software que operará el bibliotecario para la administración de los objetos de la biblioteca, será escrito en el lenguaje de programación orientado a objetos JAVA. Por lo anterior, a continuación se da una descripción del lenguaje y las características más importantes.

**JAVA**, es un lenguaje de programación orientado a objetos desarrollado por Sun Microsystems a principios de la década de 1990. Las aplicaciones Java están típicamente compiladas en un bytecode<sup>1</sup>, aunque la compilación en código máquina nativo también es posible. En el tiempo de ejecución, el bytecode es normalmente interpretado o compilado a código nativo para la ejecución, aunque la ejecución directa por hardware del bytecode por un procesador Java también es posible. (Wikipedia JAVA)

## CARACTERÍSTICAS

A continuación se listan las características principales del lenguaje de programación JAVA (Wikipedia Lenguaje Java):

**Orientado a Objetos:** se refiere a un método de programación y al diseño del lenguaje. Aunque hay muchas interpretaciones para OO, una primera idea es diseñar el software de forma que los distintos tipos de datos que use estén unidos a sus operaciones. Así, los datos y el código (funciones o métodos) se combinan en entidades llamadas objetos.

**Independencia de la plataforma:** La segunda característica significa que programas escritos en el lenguaje Java pueden ejecutarse igualmente en cualquier tipo de hardware. Esto significa ser

---

<sup>1</sup> Código intermedio más abstracto que el código máquina. Habitualmente se lo trata como a un archivo binario que contiene código máquina producido por el compilador.

capaz de escribir un programa una vez y que pueda ejecutarse en cualquier dispositivo. Se compila el código fuente escrito en lenguaje Java, para generar un código bytecode.

**El recolector de basura:** Un argumento en contra de lenguajes como C++, es que los programadores se encuentran con la carga añadida de tener que administrar la memoria solicitada dinámicamente de forma manual. En Java, este problema potencial es evitado en gran medida por el recolector automático de basura (*automatic garbage collector*). El programador determina cuándo se crean los objetos, y el entorno en tiempo de ejecución de Java (Java runtime) es el responsable de gestionar el ciclo de vida de los objetos. El programa u otros objetos pueden tener localizado un objeto mediante una referencia a éste (que, desde un punto de vista de bajo nivel es una dirección de memoria). Cuando no quedan referencias a un objeto, el recolector de basura de Java borra el objeto, liberando así la memoria que ocupaba previniendo posibles fugas (ejemplo: un objeto creado y únicamente usado dentro de un método sólo existe dentro de éste; al salir del método el objeto es eliminado).

Para la búsqueda del material de la biblioteca, es necesario un programa orientado a la Web, que pueda ser utilizado por los usuarios del Seminario. Para efectuar la búsqueda de los materiales es muy recomendable aprovechar las clases que se han ocupado para el programa de administración, por lo cual JSP es el lenguaje indicado para el programa de búsqueda del material.

### **De ésta forma, se dará una definición de JSP, su funcionamiento y sus ventajas principales.**

**JSP**, es un acrónimo de **Java Server Pages** (Páginas de Servidor Java). Permiten a los programadores de aplicaciones Web crear contenido dinámico mediante la reutilización de componentes predefinidos, y mediante la interacción con los componentes utilizando secuencia de comandos del lado del servidor (Deitel, 2004).

Las JSP's permiten la utilización de código Java mediante scripts. Además, es posible utilizar algunas acciones JSP predefinidas mediante etiquetas. Estas etiquetas pueden ser enriquecidas

mediante la utilización de Librerías de Etiquetas (TagLibs o Tag Libraries) externas e incluso personalizadas. (Wikipedia JSP)

## COMPONENTES CLAVES DE LAS JSP'S

Existen 4 componentes claves de las JSP's (Deitel, 2004) :

- **Directivas:** Son mensajes para el contenedor de JSP's, que permiten al programador especificar configuraciones de páginas, incluir contenidos de otros recursos y especificar bibliotecas de marcas personalizadas para usarlas en una JSP.
- **Acciones:** las acciones encapsulan la funcionalidad en marcas predefinidas que los programadores pueden incrustar en una JSP. A menudo las acciones se realizan con base en la información que se envíe al servidor como parte de una petición de cliente específica. También pueden crear objetos de JAVA para utilizarlos en Scriptlets de JSP.
- **Scriptlets:** Los scriptlets o elementos de secuencias de comandos, permiten a los programadores insertar código de JAVA que interactúe con los componentes de una JSP (y posiblemente con otros componentes de la aplicación web) para realizar el procesamiento de las peticiones.
- **Bibliotecas de marcas:** Las bibliotecas de marcas son parte del mecanismo de extensión de marcas que permite a los programadores crear marcas personalizadas. Dichas marcas permiten a los programadores manipular el contenido de una JSP.

## **FUNCIONAMIENTOS JSP**

**JSP** no se puede considerar un script al 100%, ya que antes de ejecutarse el Servidor de Aplicaciones<sup>2</sup> compila el contenido del documento JSP (script y etiquetas) y genera una clase Servlet<sup>3</sup>. Por lo tanto, se puede decir que aunque este proceso sea transparente para el programador no deja de ser una tecnología compilada. (Wikipedia JSP)

La principal ventaja de **JSP** frente a otros lenguajes, es que permite integrarse con clases Java (.class), lo que permite separar en niveles las aplicaciones web, almacenando en clases java las partes que consumen más recursos (así como las que requieren más seguridad) y dejando la parte encargada de formatear el documento HTML en el archivo JSP. La idea fundamental detrás de este criterio, es el de separar la lógica del negocio de la presentación de la información. (Wikipedia JSP)

Independientemente de la certeza de la aseveración, Java es conocido por ser un lenguaje muy portable (su lema publicitario reza: *escribelo una vez, córrelo donde sea*) y sumado a las capacidades de JSP se hace una combinación muy atractiva. (Wikipedia JSP)

## **VENTAJAS JSP**

El uso de la tecnología JSP, tiene fuertes ventajas en las aplicaciones Web sobre otras tecnologías como ASP, PHP, etc. Estas se mencionan a continuación. (Natxo, 2007)

**Compatibilidad con servidores:** JSP sigue la filosofía de la arquitectura JAVA de "escribe una vez ejecuta donde quieras". JSP se puede ejecutar en los sistemas operativos y servidores web más populares, como por ejemplo: Apache Tomcat, JSWDK, Sun Java System Web Server, Macromedia JRun, ServletExec, etc.

---

<sup>2</sup> Servidores Web de nueva generación que proporcionan la lógica de negocio sobre la que se construyen las aplicaciones.

<sup>3</sup> Es un objeto que se ejecuta en un servidor o contenedor JEE, fue especialmente diseñado para ofrecer contenido dinámico desde un servidor web.

**Proceso de desarrollo abierto (open source):** El API JSP se beneficia de la extendida comunidad JAVA existente.

**TAGS:** Los desarrolladores pueden crear nuevos tags y no depender tanto de los scripts.

- **Reusabilidad entre plataformas:** Los componentes JSP son reusables en distintas plataformas (UNIX, Windows).
- **La ventaja Java:** La tecnología JSP usa Java como lenguaje de Script. Java es un lenguaje muy potente y escalable, a diferencia de los lenguajes de Script. Las páginas JSP son compilados en Servlets, por lo que actúan como una puerta a todos los servicios Java de Servidor y librerías Java para aplicaciones http. Java hace el trabajo del desarrollador más fácil p. e., ayuda a proteger el sistema contra las “caídas.

## 2.3 BASE DE DATOS ORIENTADA A OBJETOS

Una base de datos orientada a objetos (BDOO) es necesaria para dar cumplimiento al objetivo del proyecto.

Las bases de datos orientadas a objetos permiten el desarrollo y mantenimiento de aplicaciones complejas con un costo significativamente menor. Permiten que el mismo modelo conceptual se aplique al análisis, diseño, programación, definición y acceso a la base de datos. (Martin, y otros, 1994)

Las bases de datos orientadas a objetos toman la idea de las bases inteligentes de datos a su conclusión lógica. No se tiene acceso a dato alguno sino es a través de los métodos almacenados en la base de datos. Estos métodos están listos para entrar en acción al momento en que reciben una solicitud. Los datos de todos los objetos quedan entonces encapsulados. En general los datos son activos más que pasivos. (Martin, y otros, 1994)

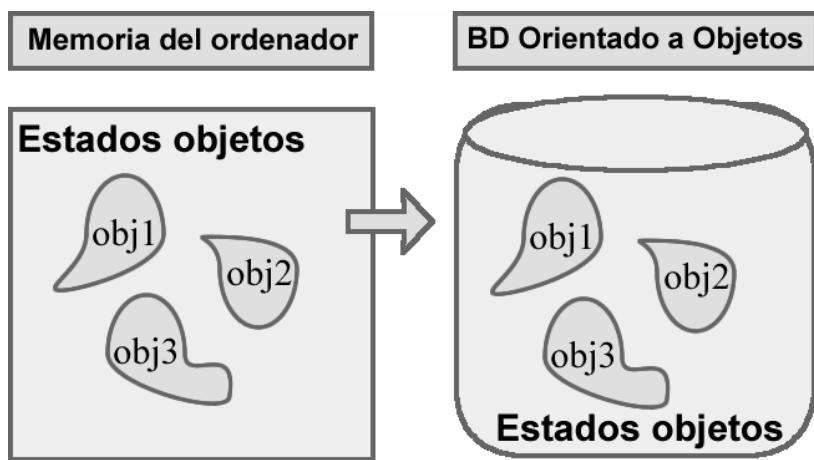


Figura 3: Base de datos Orientada a Objetos con sus respectivos estados.

En la Figura 3 se muestra como los objetos son guardados de en la base de datos, para mantener la persistencia (datos que permanecen después de terminado un proceso).

## HISTORIA DE LAS BD ORIENTADAS A OBJETOS

Las bases de datos orientadas a objetos surgieron en un principio para soportar la programación orientada a objetos. Los programadores de Smalltalk y C++ necesitaban almacenar lo que llamaban datos persistentes. Las BDOO se volvieron importantes para cierto tipo de aplicaciones con datos complejos, como CAD (diseño apoyado por computadora) y CAE (Ingeniería apoyada por computadora) (Martin, y otros, 1994).

## CRÍTICAS

Comparadas con las bases de datos relacionales, las bases de datos orientadas a objetos han sido tema de ciertas críticas:

El modelo navegacional de computación ha sido criticado por que parece como un paso atrás en el tiempo hacia las bases de datos de redes y jerárquicas. Sin embargo, hay aplicaciones de CAD y de inteligencia artificial para las cuales es absolutamente esencial navegar a través de los datos, y para las que estructura anidada de los objetos es sólo un aspecto del modelo de objetos.

Otra crítica común es que el modelo de objetos aún no tiene una teoría matemática coherente que le sirva de base. Sin embargo, debe también señalarse que ni el álgebra ni el cálculo relacional manejan otros aspectos de la tecnología de las bases de datos como la autorización, el control de concurrencia, la recuperación o el control de integridad. (Bertino, y otros, 1995)

## CARACTERÍSTICAS

Hay una serie de características que las bases de datos orientadas a objetos poseen y nos permiten conocer algunos detalles de la tecnología. Las BDOO no sólo permiten trabajar transparentemente con un entorno de programación basado en objetos, sino que soportan todos los conceptos de la tecnología de objetos, tales como: Abstracción, encapsulamiento, herencia y polimorfismo, tal como se describió en el capítulo 3.1

## EJEMPLO DE BDOO

DB4o es un novedoso motor de base de datos orientada a objetos. Sus siglas corresponden a la expresión "**D**ata**B**ase **4** (for) **O**bjects", que a su vez es el nombre de la compañía que lo desarrolla: *db4objects, Inc.*

DB4o es una solución de persistencia ideal cuando se necesita una base transparente y fácil de utilizar, ésta se puede incorporar rápidamente en un proyecto Java o .NET sin sacrificar desempeño. Logra acortar los tiempos de entrega de las aplicaciones de forma significativa, ya que permite concentrarse directamente en el dominio del problema (los objetos). Además, ofrece características avanzadas, como soporte transparente a cambios en las versiones de los objetos (evolución del esquema), cero administración (no requiere DBA) y consultas nativas. (Lavintman, y otros, 2007)

## 2.4 DOCUMENTACIÓN DE UML

Para el ciclo de desarrollo se considerará el lenguaje de modelado UML, por el soporte y universalidad a la orientación a objetos.

**UML** (*Unified Modeling Language*) es un lenguaje para especificar, visualizar, construir y documentar los artefactos de los sistemas software, así como para el modelado del negocio y otros sistemas no software [OMG01].

UML se ha convertido en la notación visual estándar de *facto y de iure* (negocio y de derecho) para el modelado orientado a objetos. Comenzó como una iniciativa de Grady Booch y Jimm Rumbaugh en 1994 para combinar las notaciones visuales de sus dos populares métodos –los métodos de Booch y OMT (*Object Modeling Technique*). Más tarde se les unió Ivar Jacobson, el creador del método Objectory, y el grupo comenzó a ser conocido como tres amigos. Muchos otros contribuyeron a UML, quizás más notablemente Cris Kobryn, que lidera el proceso de refinamiento que todavía continúa.

UML fue adoptado en 1997 como estándar por OMG (*Object Management Group*, organización que promueve estándares para la industria), y continúa siendo refinado en nuevas versiones. (Larman, 2003)

## **ARQUITECTURA del UML**

UML está basado en una arquitectura de cuatro capas, definidas a fin de cumplir con la especificación Meta Object Facility del OMG (creangel UML, 2007):

- **Meta-metamodelo:** define el lenguaje para especificar meta-modelos.
- **Metamodelo:** define el lenguaje para especificar modelos.
- **Modelo:** define el lenguaje para describir un dominio de información.
- **Objetos de usuario:** define un dominio de información específico.

## **DIAGRAMAS**

Un Diagrama es una representación gráfica de una colección de elementos de modelado, a menudo dibujada como un grafo conexo de arcos (relaciones) y vértices (otros elementos del modelo). Un diagrama no es un elemento semántico, un diagrama muestra representaciones de elementos semánticos del modelo, pero su significado no se ve afectado por la forma en que son representados.

Un diagrama está contenido dentro de un paquete. La mayoría de los diagramas de UML y algunos símbolos complejos son grafos que contienen formas conectadas por rutas. La información está sobre todo en la topología, no en el tamaño o la colocación de los símbolos (hay algunas excepciones como el diagrama de secuencia con un eje métrico de tiempo). Hay tres clases importantes de relaciones visuales: conexión (generalmente de líneas a formas de dos dimensiones), contención (de símbolos por formas cerradas de dos dimensiones), y adhesión visual (un símbolo que está "cerca" de otro en un diagrama). Estas relaciones geométricas se reasignan a conexiones entre nodos en un gráfico en la forma analizada de la notación.

La notación de UML está pensada para ser dibujada en superficies bidimensionales. Algunas formas bidimensionales son proyecciones de formas tridimensionales, tales como cubos, pero todavía se representan como íconos en una superficie bidimensional. (creangel UML, 2007)

Área	Vista	Diagramas	Conceptos principales
Estructural	Vista estática	Diagrama de Clases	Clase, asociación, generalización, dependencia, realización, interfaz.
	Vista de Casos de uso	Diagrama de Casos de Uso	Caso de Uso, Actor, asociación, extensión, generalización.
	Vista de implementación	Diagrama de Componentes	Componente, interfaz, dependencia, reutilización.
	Vista de despliegue	Diagrama de Despliegue	Nodo, componente, dependencia, localización.
Dinámica	Vista de estado de máquina	Diagrama de Estados	Estado, evento, transición, acción.
	Vista de actividad	Diagramas de Actividad	Estado, actividad, transición, determinación, división, unión.
	Vista de interacción	Diagramas de Secuencia	Interacción, objeto, mensaje, activación.
		Diagramas de Colaboración	Colaboración, interacción, rol de colaboración, mensaje.
Administración o Gestión de modelo	Vista de gestión de modelo	Diagramas de Clase	Paquete, subsistema, modelo.
Extensión de UML	Todas	Todos	Restricción, estereotipo, valores, etiquetados.

Figura 4: Diagramas UML

En la Figura 4 se muestra una tabla todos los diagramas que conforman UML, además del área al que pertenece, la vista y los conceptos principales.

## 2.5 CÓDIGOS DE BARRA

Para el ingreso de los datos de entrada en el sistema, se ocupara códigos de barra, por la rapidez y facilidad a la hora de leer los datos. A continuación, se da una descripción de códigos de barras.

El **código de barras** es un código basado en la representación mediante un conjunto de líneas paralelas verticales de distinto grosor y espaciado, que en su conjunto contienen una determinada información. De este modo, el código de barras permite reconocer rápidamente un artículo en un punto de la cadena logística y así poder realizar inventario o consultar sus características asociadas. Actualmente, el código de barras está implantado masivamente de forma global. (Wikipedia codigos Barra, 2007)

La correspondencia o mapeo entre la información y el código que la representa se denomina simbología. Estas simbologías pueden ser clasificadas en dos grupos atendiendo a dos criterios diferentes:

**Código de una dimensión:** Al efectuar la lectura de éstos códigos se tiene en cuenta el ancho de las barras y los espacios entre ellas. La altura de las barras no otorga dato alguno. Sólo codifican a no más de una docena de caracteres, y representan la clave para acceder un registro de alguna base de datos, en la Figura 5 se muestra un ejemplo para éste tipo de código.

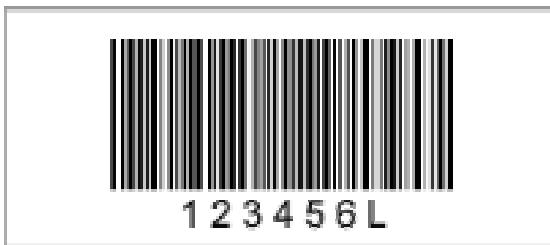


Figura 5: Código de barras Lineal.

**Códigos de dos dimensiones:** Los datos están codificados en la altura y longitud del símbolo, y en éstos códigos la información no se reduce sólo al código del artículo, sino que puede almacenar gran cantidad datos. Un código del tamaño de una estampilla postal, se pueden almacenar más de mil caracteres alfanuméricos. En la Figura 6 se muestra un ejemplo.



Figura 6: Código de barras bidimensional.

### NOMENCLATURA BÁSICA:

- **Módulo:** Es la unidad mínima o básica de un código. Las barras y espacios están formados por un conjunto de módulos.
- **Barra:** Elemento “oscuro” dentro del código. Se hace corresponder con el valor binario 1.
- **Espacio:** El elemento “claro” dentro del código. Se hace corresponder con el valor binario 0.
- **Carácter:** Formado por barras y espacios. Normalmente se corresponde con un carácter alfanumérico.

### ESTRUCTURA

Como muestra analizaremos la estructura general de un código de barras lineal, sin embargo, los conceptos se aplican también a los códigos de dos dimensiones.

Cabe hacer mención que el ancho de las barras y los espacios, así como el número de cada uno de éstos varía para cada simbología.

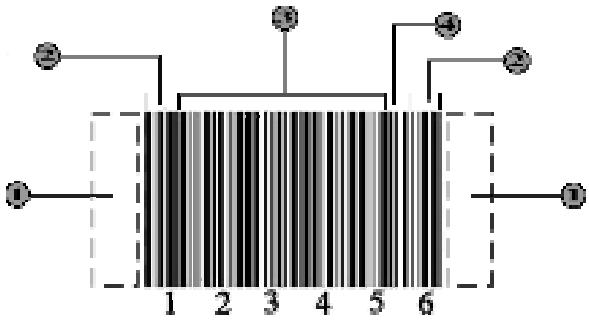


Figura 7: Estructura códigos de barra.

En la Figura 7 se puede apreciar la estructura de un código de barras, a continuación se explica en detalle la figura. (Sabater, 2007)

1. **Quiet Zone:** zona libre que rodea al código y permite al lector óptico distinguir entre el código y el resto de información contenida en el documento o en la etiqueta del producto.
2. **Caracteres de inicio y terminación:** Marcas predefinidas de barras y espacios específicos para cada simbología. Marcan el inicio y terminación de un código.
3. **Caracteres de datos:** Contienen los números o letras particulares del símbolo.
4. **Checksum:** Es una referencia incluida en el símbolo, cuyo valor es calculado de forma matemática con información de otros caracteres del mismo código. Se utiliza para ejecutar un chequeo matemático que valida los datos del código de barras. Aunque puede ser importante en cualquier simbología, no son requeridos en todas ellas.

## **2.6 SISTEMA DE CLASIFICACIÓN DECIMAL DEWEY**

Para la clasificación de los libros de la biblioteca, se utilizará el sistema de clasificación decimal Dewey que a continuación se describe:

### **HISTORIA.**

Melvil Dewey era bibliotecario en Amherst College en Massachusetts cuando tuvo la idea de crear un sistema de clasificación que respondiera a las necesidades de la biblioteca del colegio.

La primera edición del sistema decimal de Dewey con el título “A classification and Subject Index for Cataloguing and Arranging the Books and Pamphlets of a Library” no contenía en sí más de 921 categorías divididas en 10 clases principales del 000 al 999. Con el esquema iba un registro de materias que contenía más de 2500 entradas. (Miksa, 1998)

La Biblioteca del Congreso de Estados Unidos es la responsable del mantenimiento y renovación del esquema y las tablas.

Además del personal de la Biblioteca del Congreso, existe un consejo con representantes de bibliotecarios y profesores de bibliotecología. El hecho que la Biblioteca del Congreso es la garante de la revisión del sistema hace que este sistema tenga más aceptación que el sistema decimal universal CDU<sup>4</sup>, que carece de una plataforma tan firme. (Benito, 1999)

---

<sup>4</sup> Clasificación Decimal Universal, es un sistema de clasificación para ordenar y catalogar libros.

## **PRINCIPIOS GENERALES**

El sistema de clasificación decimal Dewey realiza una división de las ciencias en nueve clases principales: filosofía, religión, ciencias sociales, filología, ciencias naturales, técnica y ciencias prácticas, arte, literatura e historia. Esta división es la que existía en el siglo XIX, y que a fines del siglo XX no corresponde a la división del saber tal como lo vemos hoy, por lo que hay un desbalance entre las disciplinas.

El sistema Dewey, ha vivido tres períodos diferentes en la visión de las ciencias. El primer período, durante la vida de Dewey, en el que había una promesa explícita de no alterar los signos dados a una materia, para que las bibliotecas no se vieran obligadas a hacer cambios en el orden de sus depósitos. El segundo período, de la edición 14 a la 17, en la que se hicieron muchos cambios parciales en todas las disciplinas sin considerar el resultado, y el tercer período, el actual, en el que la remodelación de una disciplina se puede hacer completa, pero dejando las demás disciplinas intactas. (Benito, 1999).

## **BENEFICIOS**

- Ofrece un sistema lógico para organizar cada componente de la colección de su biblioteca.
- Ofrece a los usuarios de biblioteca la confianza y consistencia de un sistema de clasificación consagrado, usado formalmente en 200 000 bibliotecas de todo el mundo.

## **NOTACIÓN**

El sistema de clasificación se basa en números árabes. Estos tienen la ventaja de ser casi universales, a diferencia de las letras, que tenemos varios alfabetos, además de otras formas de representación.

Dewey decidió que todas las materias deben de tener por lo menos tres decimales. Esto quiere decir que si tenemos una materia principal con un número básico de solo una o dos cifras añadimos un cero o dos para completar. Por ejemplo usamos 200 para religión y 220 para la Biblia. Si necesitamos más de tres cifras sepáramos las cifras siguientes con un punto, por ejemplo 224.94.

En el esquema se escriben las tres cifras primeras al comienzo de la materia respectiva, para luego escribirlas solamente al comienzo de cada página. Un punto señala que las tres primeras cifras se han suprimido, con ello se hace más fácil la lectura. En la subsiguiente subdivisión se deja un espacio libre por cada tres cifras con el mismo fin. (Benito, 1999)

**El sistema se hace de diez categorías ascendentes que se siguen descomponiendo:**

- 0 Computadoras, información y obras generales
- 1 Filosofía y psicología
- 2 Religión
- 3 Ciencias sociales
- 4 Lingüística
- 5 Ciencia y matemáticas
- 6 Tecnología
- 7 Arte y recreación
- 8 Literatura
- 9 Historia y geografía

## 2.7 TECNOLOGÍA AJAX

El término AJAX se acuñó por primera vez en el artículo “Ajax: A New Approach to Web Applications” publicado por Jesse James Garrett el 18 de Febrero de 2005. Hasta ese momento, no existía un término normalizado que hiciera referencia a un nuevo tipo de aplicación web que estaba apareciendo. En realidad, el término AJAX es un acrónimo de Asynchronous JavaScript + XML, que se puede traducir como “JavaScript asíncrono + XML”.

El artículo define AJAX de la siguiente forma: “Ajax no es una tecnología en sí mismo, en realidad, se trata de la unión de varias tecnologías que se desarrollan de forma autónoma y que se unen de formas nuevas y sorprendentes”. (Pérez, 2007)

### Las tecnologías que forman AJAX son:

- XHTML y CSS, para crear una presentación basada en estándares.
- DOM, para la interacción y manipulación dinámica de la presentación.
- XML, XSLT y JSON, para el intercambio y la manipulación de información.
- XMLHttpRequest, para el intercambio asíncrono de información.
- JavaScript, para unir todas las demás tecnologías. (Pérez, 2007)

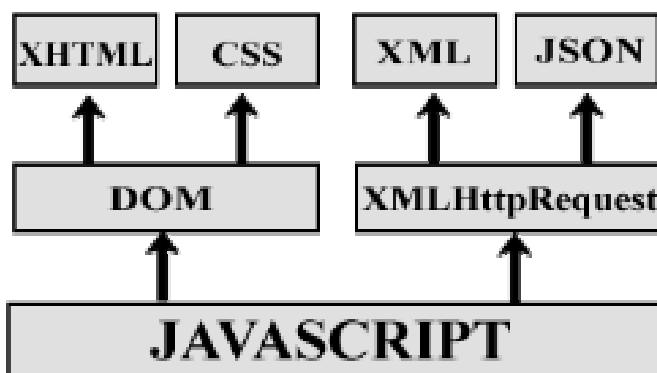


Figura 8: Tecnologías agrupadas bajo el concepto de AJAX

En la Figura 8 se muestran las tecnologías que en su conjunto forman Ajax. XHTML, CSS, se utiliza para el diseño que acompaña la información que visualizará el usuario. DOM para representar los elementos de un documento estructurado (Pagina Web). XML y JSON se encargan del formato utilizado para el retorno desde el servidor que genera la interfaz XMLHttpRequest, todo esto mediante el lenguaje JavaScript. Por ejemplo, un usuario genera una acción en un documento XML, el que envía una petición al servidor Web mediante XMLHttpRequest por medio del lenguaje JavaScript, el que recibe la petición y mediante DOM cambia el contenido de la pagina Web que contiene un diseño CSS.

Las técnicas tradicionales que se ocupan para crear aplicaciones web funcionan correctamente, pero no crea una buena sensación al usuario. Al realizar peticiones continuas al servidor, el usuario debe esperar a que se recargue la página con los cambios solicitados. Si la aplicación debe realizar peticiones continuas, la aplicación web se convierte en algo más molesto que útil.

AJAX permite mejorar completamente la interacción del usuario con la aplicación, evitando las recargas constantes de la página, ya que el intercambio de información con el servidor se produce en un segundo plano.

Las aplicaciones construidas con AJAX eliminan la recarga constante de páginas mediante la creación de un elemento intermedio entre el usuario y el servidor. La nueva capa intermedia de AJAX mejora la respuesta de la aplicación, ya que el usuario nunca se encuentra con una ventana del navegador vacía esperando la respuesta del servidor. (Pérez, 2007)

## 2.8 METODOLOGÍA A UTILIZAR

El modelo Incremental combina elementos del modelo lineal secuencial con la filosofía interactiva de construcción de prototipos. El modelo incremental aplica secuencias lineales de forma sorprendente de la misma forma que progresá el tiempo en el calendario. Cada secuencia lineal produce un <<incremento>> del software.

A diferencia de la construcción de prototipos, el modelo incremental se centra en la entrega de un producto operacional con cada incremento. (Pressman, 2002)

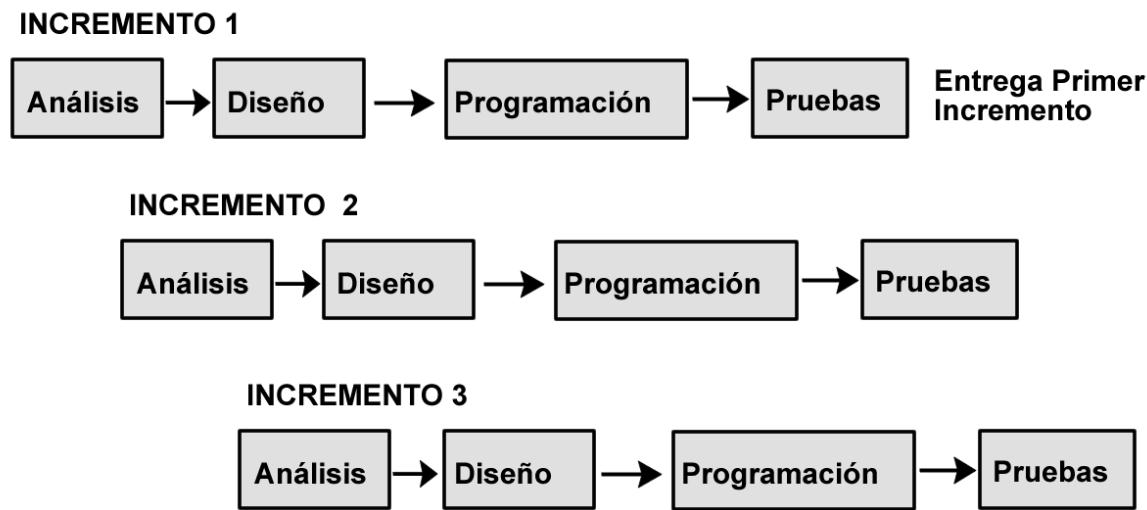


Figura 9: Grafica el modelo Incremental.

En la Figura 9 se muestran 3 incrementos de un software determinado, en cada incremento se aprecia, el análisis, diseño, programación y pruebas, finalizando con un producto acabado, para hacer uso de él desde el término del incremento, el término total del software ocurre con el último incremento que completa la construcción del sistema.

Esta metodología se ha escogido por la magnitud del proyecto, en relación a que trabajará solo una persona. La cual podrá centrarse solo en el incremento necesario, el que no será muy extenso y dará el tiempo necesario tanto para su análisis, diseño, programación y pruebas necesarias del incremento.

Además, permitirá tener en operación en un corto plazo el núcleo del software, el que permitirá desde su construcción el ingreso de los datos necesarios para su utilización (alrededor de 6.000 libros, entre otros datos). Los que permitirán realizar mejores pruebas en los incrementos posteriores.

## CAPÍTULO III: DESCRIPCIÓN SEMINARIO

En este tercer capítulo se describe el lugar físico en que se implementará el proyecto en cuestión, el Seminario Metropolitano Concepción.

Se describe, una reseña histórica, los objetivos y la estructura organizacional que este posee culminando con una definición específica de la biblioteca.

### **3.1 RESEÑA HISTÓRICA**

El Seminario de Concepción es el "colegio" más antiguo de Chile. Fundado en 1568, en Imperial y trasladado pocos años después a la "Antigua Concepción", hoy Penco; denominado por los Padres Jesuitas "Convictorio de San José", allí permaneció hasta el traslado de la ciudad.

Al trasladarse la ciudad a su ubicación actual, siguió bajo el alero de la Compañía de Jesús, permaneciendo junto a ellos hasta que fueron expulsados de América. El Seminario, por ésta causa, estuvo cerrado hasta 1777, en cuya ocasión se reabrió bajo el nombre de "Seminario de San Carlos", el cual fue cerrado en el 1814, por los azares de la lucha de independencia.

Pasaron un par de años antes de su reapertura. En dicha ocasión funcionó bajo el amparo del colegio seglar denominado "Instituto Literario".

El colegio fue destruido por el terremoto de 1835, pero luego reabrió sus puertas bajo el nombre de "Colegio Provincial". Despues de muchos años se sugirió el desglose del Seminario del colegio provincial, ésta separación se realizó bajo el mandato de Monseñor Hipólito Salas en 1855, dándole la forma definitiva.

El "Seminario Conciliar de Concepción ", se encontraba dividido en dos secciones: la eclesiástica y la seglar. La sección seglar se clausuró en 1937 y la eclesiástica en 1965.

El Seminario fue reabierto en 1977, en la ciudad de Concepción para luego ser trasladado en 1981 a su actual ubicación en la comuna de Chiguayante.

## **3.2 QUE ES UN SEMINARIO**

Un Seminario, es una casa de formación para adultos que de manera voluntaria y aceptados por las autoridades eclesiásticas de la Iglesia Católica, inician un itinerario de estudios que los conduce al ministerio sacerdotal. Los seminarios pertenecen jurídicamente a la diócesis<sup>5</sup> bajo la autoridad del obispo. En adición a esto, las iglesias que poseen este sistema de formación sacerdotal, cuentan también con la forma de “Seminario Menor” que no es otra cosa que un colegio que puede ir desde la educación básica a la secundaria para menores de edad, bajo la tutela de la diócesis con proyección sacerdotal.

## **3.3 DEFINICIÓN DEL SEMINARIO**

**NOMBRE** : Seminario Metropolitano Concepción.

**DIRECCIÓN** : O'Higgins 1487, comuna de Chiguayante, Concepción, región Bío-Bío.

**WEB** : <http://www.seminarioconcepcion.cl>

**FONO** : 41-2361463.

**EMAIL** : [seminarioconcepcion@gmail.com](mailto:seminarioconcepcion@gmail.com)

---

<sup>5</sup> Distrito o territorio cristiano en que tiene y ejerce jurisdicción eclesiástica el arzobispo u obispo.

El Seminario Metropolitano Concepción hoy cuenta con 41 seminaristas, los que ingresan tras pasar por un proceso de discernimiento, que va desde pláticas con algún sacerdote hasta jornadas vocacionales<sup>6</sup>. Los seminaristas viven en casas dentro del Seminario y cada una de ellas representa una comunidad que hace referencia al año en que se encuentran del proceso de formación. El primer año en que ingresan, participan de la comunidad “Propedéutico” (año de nivelación), las clases son impartidas en salas que se encuentran al interior de la institución.

Los seminaristas, tienen tanto actividades generales, como propias de su comunidad, por ejemplo el desayuno y la once es un momento propio de cada comunidad, y el almuerzo es una actividad en que se reúnen todas las comunidades en un comedor común.

Hoy en día el Seminario atiende a jóvenes seminaristas de las diócesis de Chillán, Los Ángeles y Concepción, para ello cada diócesis aporta con becas para que los seminaristas puedan vivir en el Seminario y entrar en el proceso de formación.

---

<sup>6</sup> Encuentro en el cual sacerdotes y sicólogos ayudan al joven a discernir su vocación y ver si son aptos para entrar en el proceso de formación.

## **VISIÓN**

Ser una comunidad educativa en camino: la comunidad promovida por el Obispo para ofrecer, a quien es llamado por el Señor para el servicio apostólico, la posibilidad de revivir la experiencia formativa que el Señor dedicó a los Doce.

## **MISIÓN**

Formar a pastores bajo el ejemplo de Jesús Sacerdote y Buen Pastor.

## **OBJETIVOS ESTRATÉGICOS**

- Remodelar las estructuras físicas en las cuales estudian y viven los seminaristas.
- Mejora de la administración y gestión de la Biblioteca.
- Contratación de más profesores para la formación.

## ESTRUCTURA ORGANIZACIONAL

Un Seminario Mayor, para ser regular, debe contar como mínimo con el siguiente personal *siempre elegido por el obispo*:

- **Un rector:** Un sacerdote que rige la vida del Seminario, vela por el cumplimiento objetivo de los estudios, hace un seguimiento en conciencia de los candidatos y presenta reportes claros al obispo.
- **Un director espiritual:** Un sacerdote dispuesto a escuchar y a acompañar las inquietudes de los estudiantes.
- **Un confesor:** Un sacerdote dispuesto a ejercer el sacramento de la Confesión.

El Seminario Metropolitano Concepción tiene su estructura como se muestra en la siguiente figura.

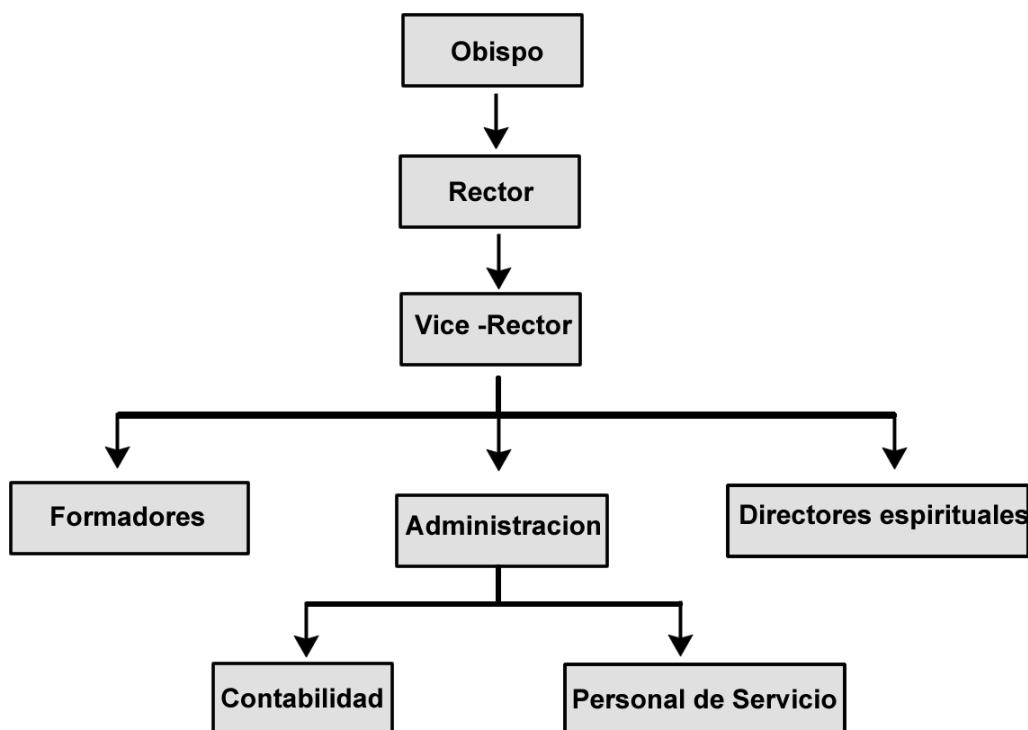


Figura 10: Estructura Organizacional del Seminario Metropolitano Concepción.

En la Figura 10 se muestra como están organizados jerárquicamente los cargos y departamentos del Seminario Metropolitano Concepción, en ella se puede apreciar que el Seminario está al mando del obispo y que para las decisiones inmediatas existe el rector, quien es el que vela conjuntamente con las demás entidades para el buen funcionamiento de la institución.

## **DESCRIPCIÓN DE LA BIBLIOTECA**

El Seminario de Concepción tiene el honor de ser el colegio más antiguo de Chile. Sus alumnos han ocupado los puestos más altos de la Iglesia y la Nación, de sus aulas han salido tres presidentes de la República, Obispos, Ministros, Senadores, Diputados, etc.

Ya desde el comienzo, del Seminario cuenta con colecciones de mapas, cuadros murales y toda clase de enseñanza. Tiene una esplendida biblioteca, obsequio en gran parte del Ilustre Sr. Salas. Además, está suscrita a las más importantes revistas europeas.

Actualmente la biblioteca cuenta con 11.000 libros clasificados y unos 40.000 sin clasificar, entre los que destacan libros de filosofía, teología, historia de la iglesia, vida de santos y las sagradas escrituras, además de poseer la primera edición de la Araucana, ediciones canónicas, archivos del arzobispado, archivo fotográfico del arzobispado, entre otras. Estos en su mayoría han sido donados y otros adquiridos por cuenta propia.

La biblioteca está conformada por dos edificios, el primero (de dos pisos) posee los libros más usados y es donde se realiza el proceso de préstamos, búsqueda y devolución de libros. El segundo edificio (de tres pisos) es en donde se encuentra el resto del material mantenido por un sistema de conservación, que permite a los libros permanecer en buen estado, debido a que impide que el aire húmedo penetre en ellos.

La biblioteca opera de forma totalmente manual, cada libro posee una ficha en donde se anota la fecha de devolución. El sistema de búsqueda es a través de fichas catalográficas<sup>7</sup> que se encuentran disponibles en estantes para que cada usuario haga uso de ellas.

---

<sup>7</sup> Tarjeta de papel o cartulina de forma rectangular y dimensiones pequeñas en la que se fija de forma sintetizada el contenido y los caracteres externos de un documento o libro.

## CAPÍTULO IV: ESTUDIO DE FACTIBILIDAD

En este cuarto capítulo se documenta el estudio de factibilidad, en el cual se determinó si el proyecto es viable para el Seminario o si es recomendable que se lleve a cabo. El estudio consta de tres partes: factibilidad económica, técnica y operacional.

Finalmente, se entrega una conclusión de los tres estudios, en la que se determina la factibilidad del proyecto.

## **4.1 FACTIBILIDAD TÉCNICA**

Para llevar a cabo el análisis de factibilidad técnica se consideran todos los recursos que se necesitan para el proyecto.

### **Recursos Humanos, con conocimiento en.**

- Tecnología Orientada a Objetos.
- Plataforma JAVA.
- Conocimientos del funcionamiento de una biblioteca.

### **Recursos Hardware.**

- 1 PC para el desarrollo del Proyecto.
  - 512 Memoria.
  - Procesador 1 GHz.
  - 20 GB Disco duro.
  - Tarjeta de red inalámbrica.
- 1 PC para Servidor
  - 2 GB Memoria.
  - 300 GB Disco duro.
  - Tarjeta de red inalámbrica estándar IEEE 802.11g.
  - Procesador 1Ghz.
- 2 PC para clientes.
  - Procesador 500 MHz.

- 512 MB Memoria.
  - 10 GB Disco duro.
  - Tarjeta de red inalámbrica estándar IEEE 802.11g.
- 
- 1 Router Inalámbrico estándar IEEE 802.11g.
  - 1 Impresora códigos de Barras Zebra LP 2824.
  - 1 Lector Código de Barras.
  - 1 Impresora Multifuncional.

### **Recursos Software.**

- Software de diseño.
  - Programa StarUML.
  - Editor de Texto.
- Software de Programación.
  - NetBeans IDE 5.
  - Java Development Kit, entorno de desarrollo Java.
  - Apache Tomcat 6.0.
  - Base de Datos Bb4o.
- Sistema Operativo.
  - 1 Licencia Windows XP pro.
  - Linux Ubuntu.

La tecnología que se necesita se puede adquirir en su totalidad, a continuación se detalla cada punto.

**Recursos Humanos:**

El recurso humano corresponde al alumno que desarrolla este proyecto, quién posee los conocimientos que son necesarios, como lo es la tecnología orientada a objetos, plataforma JAVA y en los requerimientos de la biblioteca.

**Recursos Hardware:**

Será adquirido por completo, menos el PC para el desarrollo ya que se cuenta con él. Los costos se detallan en el estudio de factibilidad económica.

## **Recursos Software:**

El recurso software en su totalidad es software gratuito o freeware, que quiere decir, que se distribuye sin costo, y por tiempo ilimitado; en contraposición al shareware (en el que la meta es lograr que usuario pague, usualmente luego de un tiempo "trial" limitado y con la finalidad de habilitar toda la funcionalidad). A veces se incluye el código fuente, pero no es lo usual. El freeware suele incluir una licencia de uso, que permite su redistribución pero con algunas restricciones, como no modificar la aplicación en sí, ni venderla, y dar cuenta de su autor. También puede desautorizar el uso en una compañía con fines comerciales o en una entidad gubernamental.

Los equipos que se proponen para el proyecto, son aptos para soportar los objetos que el sistema necesita para operar, y en su conjunto ofrecen una fácil incorporación de nuevos computadores en la biblioteca, sólo bastará de una tarjeta de red inalámbrica para que se conecte al sistema, permitiendo un grado alto de escalabilidad. En caso, que el sistema requiera ser utilizado desde otro lugar ajeno a la biblioteca, sólo bastará de una conexión a internet por parte del servidor.

## **4.2 FACTIBILIDAD ECONÓMICA**

Para comenzar con el estudio de factibilidad económica, se listan todos los recursos necesarios para que el sistema pueda operar de manera óptima.

### **Recursos Humanos.**

1 Ingeniero (e) en computación e informática. 3 hrs. Diarias, 5 días semanales, durante 5 meses. \$ 3.500 por hora.

**TOTAL: \$1.050.000**

### **Recursos Hardware.**

**NOTA:** Los siguientes valores fueron cotizados en PC Factory, a excepción de la impresora de códigos de barra cotizada en Tecnologística LTDA.

- PC Servidor:
  - AMD CPU Sempron 3000+ Box (AM2)
  - PC-Chips M/B AMD A13G Pci-E A/V/L (AM2)
  - Samsung Disco Duro Sata2 160Gb 7200 rpm
  - Multimarca DDR2 1GB 533Mhz PC4200
  - Monitor LCD 15" 540N S/Black Samsung

**VALOR: \$274.590**

- 2 pc:
  - AMD CPU Sempron 3000+ Box (AM2)
  - PC-Chips M/B AMD A13G Pci-E A/V/L (AM2)
  - Samsung Disco Duro 80Gb 7200rpm
  - Multimarca DDR2 512Mb 533Mhz PC4200
  - Monitor LCD 15" 540N S/Black Samsung

**VALOR: \$224.590**

- Router Inalámbrico AWR-MIMO-54RA 4P.

**VALOR: \$47.000**

- Impresora códigos de Barras Zebra LP 2824.

**VALOR: \$250.000**

- POS Lector Código de Barras Laser USB

**VALOR: \$76.800**

- Multifuncional Pixma MP160

**VALOR: \$49.100.**

### **Recursos Software.**

- Software de diseño.
  - Programa StarUML.
  - Editor de Texto.
- Software de Programación.
  - NetBeans IDE 5.
  - Java Development Kit, entorno de desarrollo Java.
  - Apache Tomcat 6.0.
  - Base de Datos Bb4o.
- Sistema Operativo.
  - 1 Licencia Windows XP pro. \$97.500
  - Linux Ubuntu.

### **Costos Fijos**

- Costos de Mantención.
  - Luz eléctrica: 330 W (aprox). Todo el sistema.

**VALOR: \$5.000 aprox.**

El valor total del proyecto es:

1.050.000	Recursos humanos
274.590	PC Servidor
449.180	2 PC clientes
47.000	Router Inalámbrico
250.000	Impresora de códigos de barra
76.800	Lector Códigos de Barra
49.100	Impresora Multifuncional
97.500	Licencia Windows

---

**2.294.170 TOTAL**

El valor total de todos los recursos necesarios para este nuevo sistema es de: \$2.294.170, pero teniendo en consideración que se trata de un proyecto de título el valor del sistema será de **\$1.244.170** descontando los valores del recurso humano.

## **FUNCIONAMIENTO SISTEMA ACTUAL.**

En la actualidad, la biblioteca cuenta con un bibliotecario que trabaja un total de 45 horas semanales con un sueldo mensual de \$220.694 (líquido), las actividades que realiza el bibliotecario se detallan a continuación.

<b>ACTIVIDAD</b>	<b>Horas/semanales</b>	<b>Material</b>
Ingrera y clasifica Libros	32 hrs. semanales	60 libros y/o revistas
Préstamo y devolución del material	3 hrs. semanales	25 libros y/o revistas
Aseo y mantención de la biblioteca	8 hrs. semanales	0
Búsqueda de libros.	2 hrs. semanales	35 libros y/o revistas
<b>TOTAL</b>	<b>45 hrs. Semanales</b>	

Las actividades anteriores suman un total de 45 horas semanales y 180 horas al mes, por lo que el costo de trabajo del bibliotecario es de: \$220.694/180 hrs. =\\$ 1.226/hr.

## **FUNCIONAMIENTO NUEVO SISTEMA.**

Con el nuevo sistema y considerando el mismo material, el bibliotecario debiera tardar los siguientes tiempos.

<b>ACTIVIDAD</b>	<b>Horas/semanales</b>	<b>Material</b>
Ingrera y clasifica Libros	12 hrs. semanales	60 libros y/o revistas
Préstamo y devolución del material	1/2 hrs. semanales	25 libros y/o revistas
Aseo y mantención de la biblioteca	8 hrs. semanales	0
Búsqueda de libros.	1/2 hrs. semanales	35 libros y/o revistas
<b>TOTAL</b>	<b>21 hrs. Semanales</b>	

Las actividades anteriores suman un total 84 hrs de trabajo al mes, por lo que nos quedan 96 horas que se pueden dedicar, por ejemplo al ingreso y clasificación de libros lo que nos significa un aumento en la productividad. El aumento de producción por el mes puede quedar especificado de la siguiente manera.

$$\$1.226/\text{hrs} * 96/\text{hrs} = \$117.696$$

Además, se considera un gasto de \$5.000 en luz eléctrica, el costo mensual de operación del sistema seria de:

$$\$220.694 + \$5.000 = \$225.695$$

Pero si se considera que mensualmente, existe un aumento en la productividad por parte del bibliotecario de \$117.696. Se puede deducir que sistema actual traerá beneficios a partir del undécimo mes. Ya que es en este mes, donde el aumento de producción supera los costos del proyecto.

$$11 * \$117.696 = \$1.294.656 > \mathbf{\$1.244.170}$$

## **4.3 FACTIBILIDAD OPERACIONAL**

El “Sistema Orientado a Objetos para biblioteca del Seminario Metropolitano Concepción” será utilizado por los seminaristas y el bibliotecario del Seminario. Tanto el bibliotecario como los seminaristas, son usuarios aptos para manipular el sistema, ya que poseen los conocimientos básicos en computación, que se requieren para utilizar el sistema.

Por otra parte, los usuarios esperan con entusiasmo que la Biblioteca opere de forma automatizada, ya que se beneficia tanto el bibliotecario como los seminaristas. También es importante, que los aspectos principales del sistema que actualmente funcionan, permanezcan, como es el sistema de clasificación Dewey y los criterios de búsqueda, esto permitirá que el cambio no sea muy repentino y se evite una resistencia a un cambio muy brusco.

El presbítero Francisco Steigmer, rector del Seminario, se encuentra muy entusiasmado por el proyecto, ya que entiende que el sistema permitirá proveer bastante información para la toma de decisiones. También, los usuarios esperan con ansias que el sistema se encuentre operando, ya que saben los grandes beneficios que éste les brindará. De lo anterior, podemos apreciar que por parte de la administración y de los usuarios existe apoyo para el proyecto.

El rector, el bibliotecario y algunos seminaristas, han colaborado de forma constante en la planificación del proyecto, mediante reuniones en las cuales se han pronunciado acerca de los requerimientos y el funcionamiento actual que posee la biblioteca. Además, el sistema no presentará mayor dificultad al hacer uso de él, bastará seleccionar en pantalla un par de opciones para que la información esté en forma inmediata. También es pertinente mencionar si ante una falla o falta de alimentación el sistema aborta, la información no podrá ser utilizada hasta que la alimentación se restablezca.

En la biblioteca del Seminario, la mayor parte de los libros no se encuentran clasificados, esto se debe al tiempo que requiere el ingresar y clasificar un libro, impidiendo el uso de este material. El sistema que actualmente se evalúa, permitirá una mayor productividad en el ingreso, clasificación, búsqueda, préstamos y devolución de los libros y revistas.

## **4.4 CONCLUSIÓN**

Después de realizar el estudio de factibilidad económica, técnica y operacional se puede concluir que la realización del proyecto es factible, ya que se cuenta con el capital necesario para su realización, recuperando la inversión en un tiempo de once meses para empezar a aumentar la producción en un 115% aproximadamente.

Además, el recurso hardware y humano que se propone son totalmente acordes con las necesidades del proyecto.

Para finalizar, los usuarios que operarán el nuevo sistema sólo les bastará una preparación para que puedan utilizar el sistema, ya que los conocimientos básicos que se necesitan estos ya los poseen.

En vista de todo lo anterior, se decidió realizar el proyecto.

## CAPÍTULO V: ANÁLISIS

La popularidad de las tecnologías de objetos, ha generado docenas de métodos de análisis orientado a objetos desde finales de los 80 y durante los 90. Cada uno de ellos introduce un proceso para el análisis de un producto o sistema, un conjunto de modelos que evoluciona fuera del proceso, y una notación que posibilita al ingeniero del software crear cada modelo de una manera consistente. (Pressman, 2002)

En este capítulo se definirán los objetivos del software, el alcance de este, la interfaz de hardware, las principales actividades que la biblioteca posee mediante diagramas de actividad, los paquetes que formarán parte de la lógica de negocio, los principales casos de uso y el diagrama conceptual correspondiente a cada paquete.

## **5.1 DESCRIPCIÓN GLOBAL DEL PRODUCTO**

### **OBJETIVO DEL SOFTWARE.**

El software a desarrollar tiene como objetivo general: ser una herramienta de utilidad, para la administración y control de la biblioteca del Seminario Metropolitano Concepción, con el fin de automatizar todas las actividades relacionadas con los objetos que interactúan en ésta, y así dar una mayor eficiencia en los préstamos y devoluciones del material existente, tales como libros y revistas. También es parte del objetivo, dar un mayor rendimiento en la clasificación e ingreso de los libros y revistas, actividades que se realizan de forma constante.

### **OBJETIVOS ESPECÍFICOS DEL SOFTWARE.**

- Contar con una base de datos orientada a objetos.
- Permitir la administración de usuarios y del material existente (Libros y revistas).
- El software debe ser compatible con el ingreso de datos mediante código de barras.
- El software deberá permitir que los libros sean clasificados con la clasificación Decimal Dewey.
- El software deberá ordenar los resultados de una búsqueda (de libros) tomando en cuenta recomendaciones de otros usuarios.
- El software deberá permitir la emisión de informes para la toma de decisiones.

## **ALCANCES**

El software permitirá mantener información actualizada de los objetos que se manipulan en la biblioteca del Seminario, mediante el ingreso, modificación y bajas que se producen en el transcurso del tiempo. También permitirá, al bibliotecario, ejecutar las acciones propias que el software maneja, como lo son los préstamos y manipulaciones del material. A continuación se listan las funciones principales del sistema:

- Ingreso de libros y revistas.
- Préstamos y devoluciones del material existente a usuarios registrados.
- Comunicación de los usuarios del sistema a través de las cuentas.
- Visualización y administración de documentos digitales por parte de los usuarios.
- Generación de Reportes preestablecidos.
- Búsqueda del material existente.

## **INTERFAZ DE HARDWARE**

**PC SERVIDOR:** Será el computador en que se ejecutarán las operaciones principales del sistema y el que contará con la base de datos.

- 1Gb Memoria.
- 180 GB tamaño disco Duro, 7200 rpm.
- Tarjeta de red inalámbrica, estándar IEEE 802.11g.

**PC CLIENTES:** Son los computadores asociados a búsqueda de material, éstos deben contar con una conexión a la red local.

- 512 MB Memoria.
- 80 GB Tamaño disco duro 7200 rpm.
- Tarjeta de red inalámbrica, estándar IEEE 802.11g.

**ROUTER:** Es el que establece conexión inalámbrica desde un Punto de Acceso, estándar IEEE 802.11g.

**IMPRESORA CÓDIGOS DE BARRA:** Es la que permitirá la creación de los códigos de barras para el registro de los libros y revistas de la biblioteca.

**LECTOR DE CÓDIGOS DE BARRA:** Permite la lectura de los códigos para los préstamos y devoluciones del material de la biblioteca.

**IMPRESORA MULTIFUNCIONAL:** Permitirá la impresión de los reportes requeridos del software.

## 5.2 PRINCIPALES ACTIVIDADES

### REGISTRO Y CLASIFICACIÓN DE LOS LIBROS Y REVISTAS.

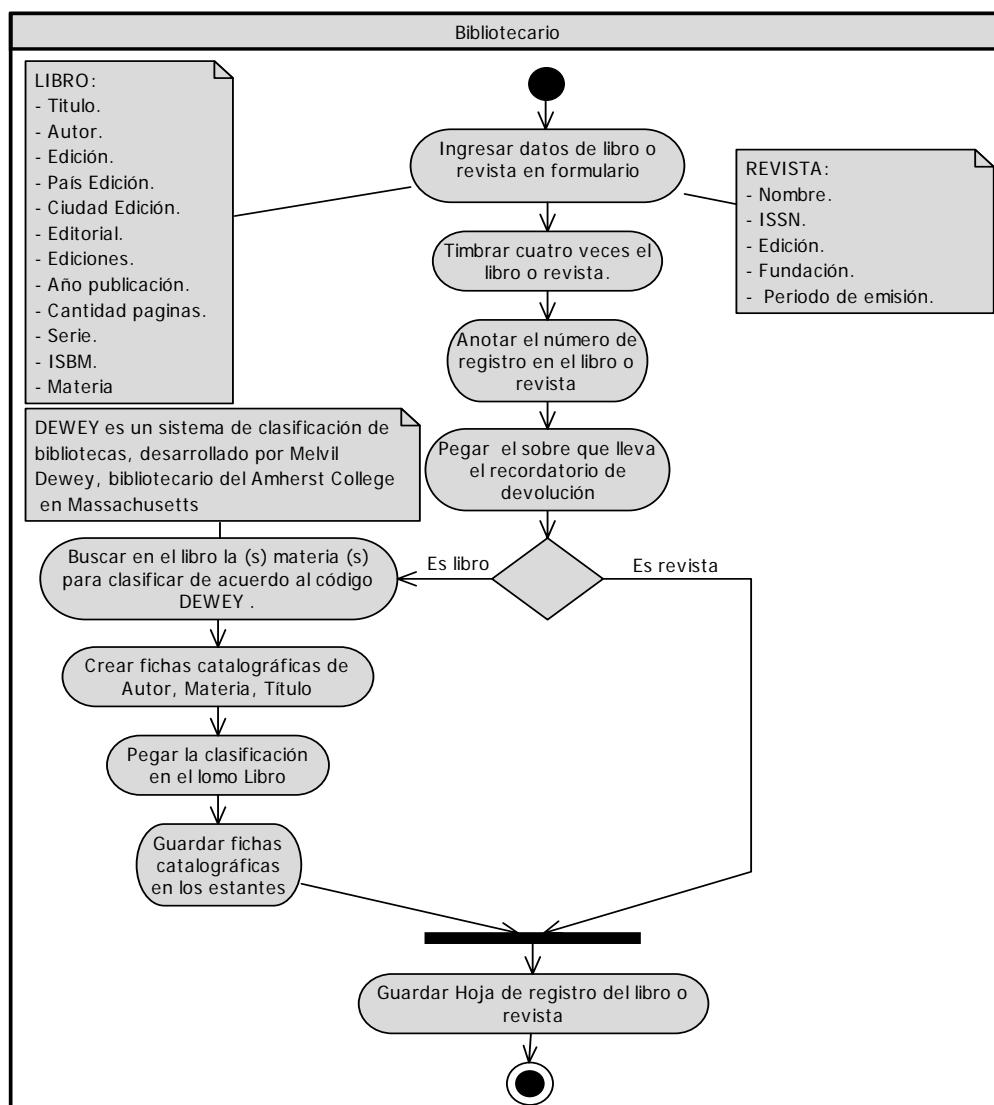


Figura 11: Registro y calificación de libros y revistas.

La Figura 11 muestra en detalle las actividades que se realizan para llevar a cabo el registro y clasificación de los libros y revistas. El registro y clasificación se realiza para que el material pueda quedar al servicio de todos los usuarios de la biblioteca del Seminario Metropolitano Concepción. Un libro necesita de más cuidado al momento de la clasificación, ya que es

necesario clasificarlo con el Sistema de Clasificación Decimal Dewey, para luego crear las fichas catalográficas que posteriormente se deben dejar en los estantes de búsqueda.

## PRÉSTAMOS DE LIBROS Y REVISTAS.

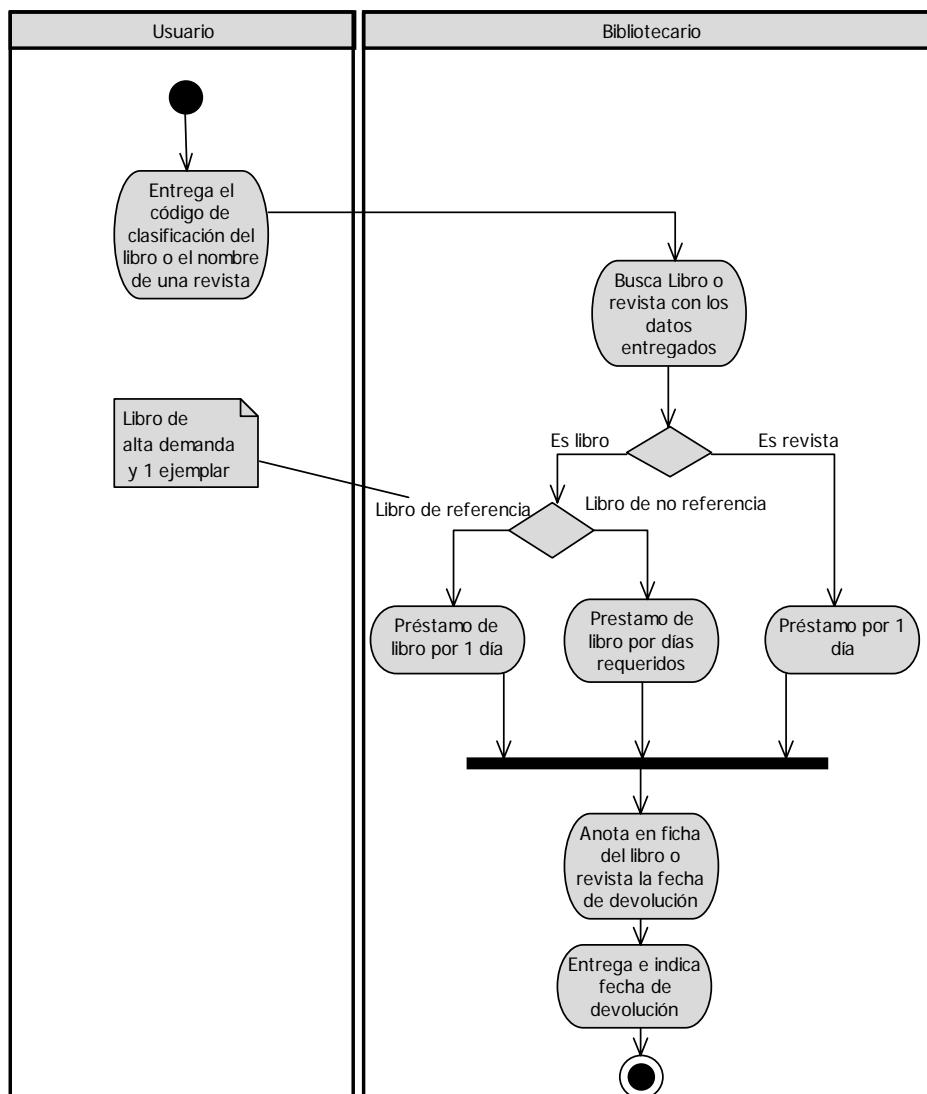


Figura 12: Préstamos de libros y revistas.

En la Figura 12 se muestran las actividades necesarias para poder llevar a cabo un préstamo de libro o revista. Lo más importante que se muestra, es la decisión que se toma para un préstamo ligado a un libro de referencia, ya existe un ejemplar y además son de alta demanda, por lo que el

préstamo debe ser solo por el día, en cambio los libros de no referencia son prestados de acuerdo a los requerimientos del usuario y el criterio del bibliotecario.

## DEVOLUCIÓN DE LIBROS Y REVISTAS.

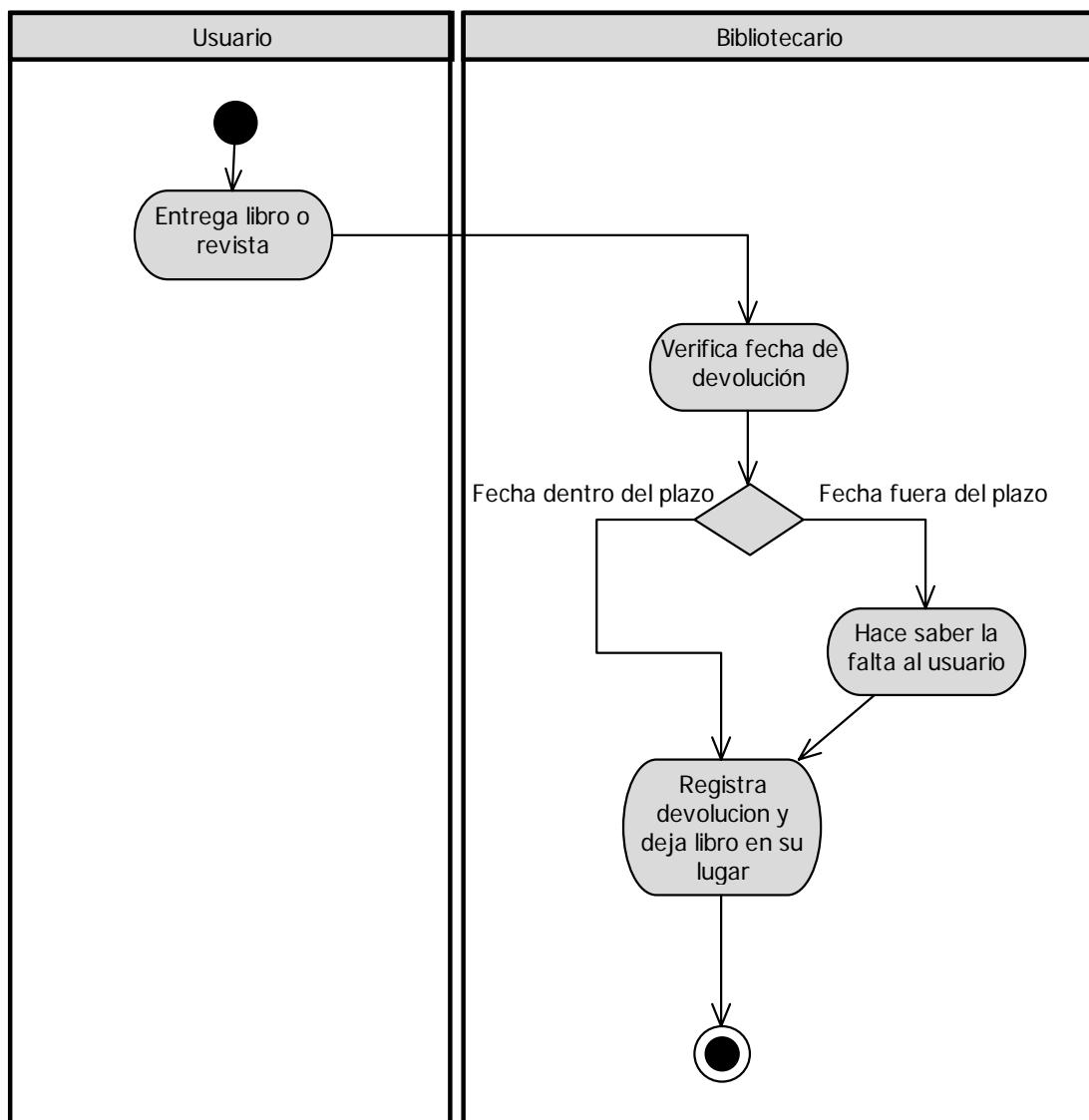


Figura 13: Devolución de libros y revistas.

En la Figura 13 se muestra el proceso de devolución, en el cual es necesario devolver el material en la fecha estipulada, el seminarista debe llevar el material a la biblioteca para entregárselo al

bibliotecario quien registra la devolución en la ficha que pertenece al libro o revista, si este es devuelto fuera del plazo, el bibliotecario debe informar la falta al usuario, para que éste no vuelva a cometerla, pero no existe forma de registrar o sancionar la falta cometida al seminarista, debido a la forma en que hoy opera la biblioteca.

## BÚSQUEDA DE MATERIAL.

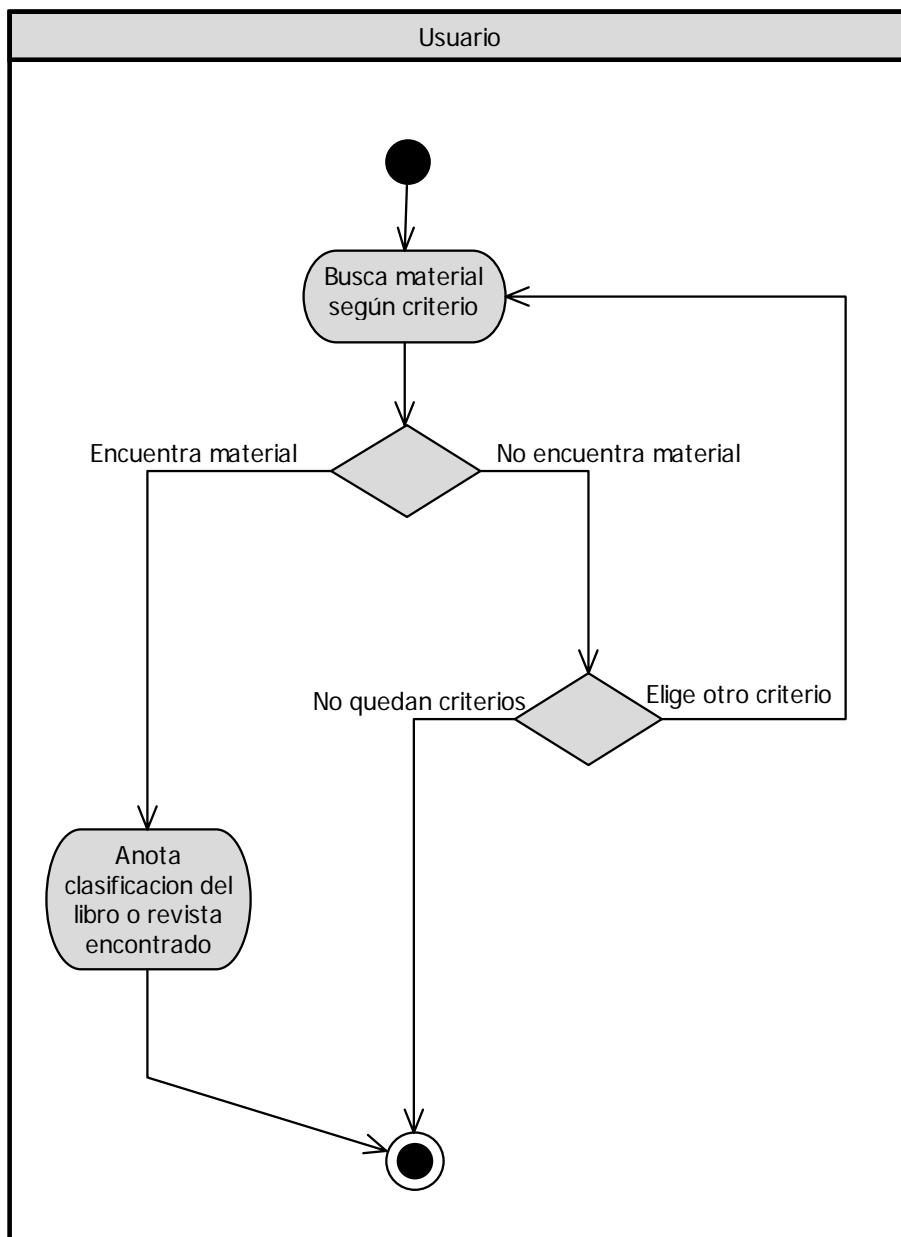


Figura 14: Búsqueda de Material.

La Figura 14 muestra la actividad de búsqueda del material, una tarea importante para realizar un préstamo de un libro, ya que permite asociar el código de clasificación a un libro, ésta actividad se realiza mediante tarjetas las que se revisan hasta encontrar lo que uno desea. Los criterios que existen son, por título, autor y materia.

## **5.3 REQUERIMIENTOS ESPECÍFICOS**

### **REQUERIMIENTOS FUNCIONALES DEL SISTEMA**

A continuación se listan los requerimientos acordados con el bibliotecario y usuarios de la biblioteca del Seminario Metropolitano Concepción. En los requerimientos R2, R5, R9 y R12 se menciona “baja” en vez de eliminación, ya que en realidad no se está eliminando información ya que ésta permanecerá en el sistema pero si se está deshabilitando, es decir existe una eliminación lógica.

- **R1:** Ingresar libros con su clasificación (Incluyendo copias).
- **R2:** Dar de baja a libros.
- **R3:** Modificación de propiedades de los libros.
- **R4:** Ingreso de Revistas.
- **R5:** Dar de baja a las revistas.
- **R6:** Modificación de propiedades de las revistas.
- **R7:** Creación de cuentas de usuarios de la biblioteca (Seminaristas).
- **R8:** Modificación de cuentas de usuarios (Seminaristas).
- **R9:** Dar de baja a usuarios del sistema (Seminaristas).
- **R10:** Crear cuentas de bibliotecario.
- **R11:** Modificación de cuenta de bibliotecario.
- **R12:** Dar de baja cuentas de bibliotecario.
- **R13:** Préstamos de libros y revistas a los usuarios de la biblioteca
- **R14:** Devoluciones de libros y revistas a los usuarios de la biblioteca.
- **R15:** Sanciones por atraso en las devoluciones.
- **R16:** Permitir préstamos a usuarios esporádicos que se acerquen a la biblioteca.
- **R17:** Generación de reportes con la información que genere el sistema.

- **R18:** Búsqueda de libros y/o revistas.
- **R19:** Las devoluciones pendientes deben aparecer en la cuenta de cada usuario junto con los préstamos históricos.
- **R20:** Permitir envío de mensajes tanto generales como individuales a través de la cuenta.
- **R21:** Permitir a los usuarios del sistema actualizar sus datos personales.
- **R22:** Permitir a los usuarios almacenar documentos digitales en su cuenta.
- **R23:** Permitir a los usuarios visualizar los documentos digitales de otros usuarios.
- **R24:** Permitir ingreso al sistema mediante usuario y contraseña.

## **DESCRIPCIÓN MEDIANTE CASOS DE USO.**

### **ACTORES :**

**Bibliotecario:** Es la persona que tiene a cargo el cuidado, ordenación y servicio de la biblioteca, es tarea de él buscar las alternativas de clasificación para el material existente, tomar las decisiones que se presenten en un momento determinado, acerca de los préstamos o no del material. Es quien tendrá a cargo las funciones administrativas del sistema.

**Seminarista:** Usuario del sistema que utilizará la biblioteca y solicitará los préstamos del material existente, además de realizar búsquedas y manejar la información de su cuenta.

**Usuario Especial:** Usuario ajeno a la biblioteca que necesita esporádicamente la utilización del material que existe en la biblioteca del Seminario.

## DIAGRAMA DE PAQUETES

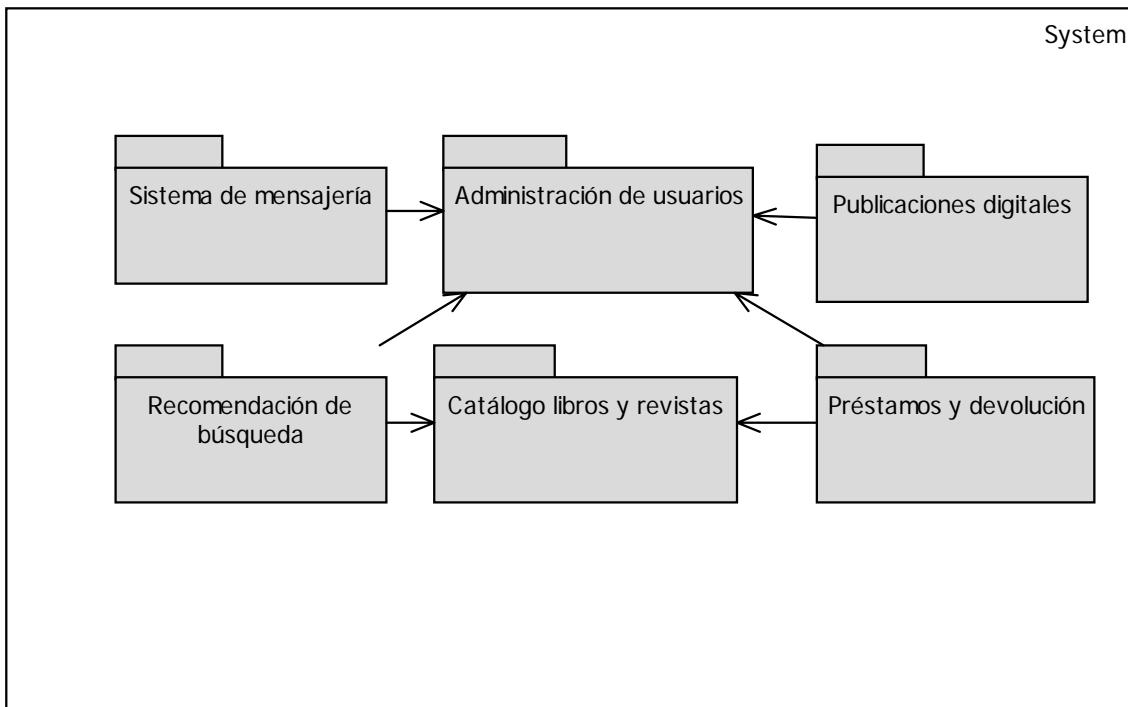


Figura 15: Paquetes para casos de uso.

En la Figura 15 se muestra como el sistema está dividido en agrupaciones lógicas denominadas paquetes, estos son dependientes unos de otros, exceptuando “Catalogo libros y revistas” y “Administración de usuarios” que se pueden trabajar por separado al ser totalmente independientes de los demás paquetes.

- **Paquete catálogo libros y revistas:** Es el paquete encargado de controlar toda la información referente a los libros y revistas existentes en la biblioteca, además de gestionar el ingreso, modificación y búsqueda del material. La búsqueda del material es propia de este paquete, pero la ordenación de los resultados la realiza el paquete “recomendación de búsqueda”.
- **Paquete administración de usuarios:** Es el paquete encargado de controlar la información que pertenece a los usuarios del sistema, así como el ingreso, modificación, y búsqueda de usuarios.

- **Paquete préstamos y devolución:** Es el encargado de registrar los préstamos que realizan los usuarios en la biblioteca, verificar las devoluciones, generar las sanciones necesarias entre otras. Este paquete es totalmente dependiente de los paquetes “catálogos libros y revistas” y “administración de usuarios”.
- **Paquete recomendación de búsqueda:** Es el encargado de ordenar la búsqueda de libros por medio de la recomendación que los seminaristas han hecho del material que han utilizado. Este paquete es totalmente dependiente de los paquetes “catálogos libros y revistas” y “administración de usuarios”.
- **Paquete sistema de mensajería:** Encargado de comunicar usuarios entre sí, es decir, permitirá a los usuarios del sistema enviar mensajes a otros usuarios. Este paquete es totalmente dependiente del paquete “Administración de usuarios”.
- **Paquete publicaciones digitales:** Encargado de almacenar documentos digitales que los usuarios estimen convenientes para luego ser descargados por otros usuarios. Este paquete es totalmente dependiente del paquete “Administración de usuarios”.

## PAQUETE CATÁLOGO LIBROS Y REVISTAS.

### DIAGRAMA DE CASOS DE USO

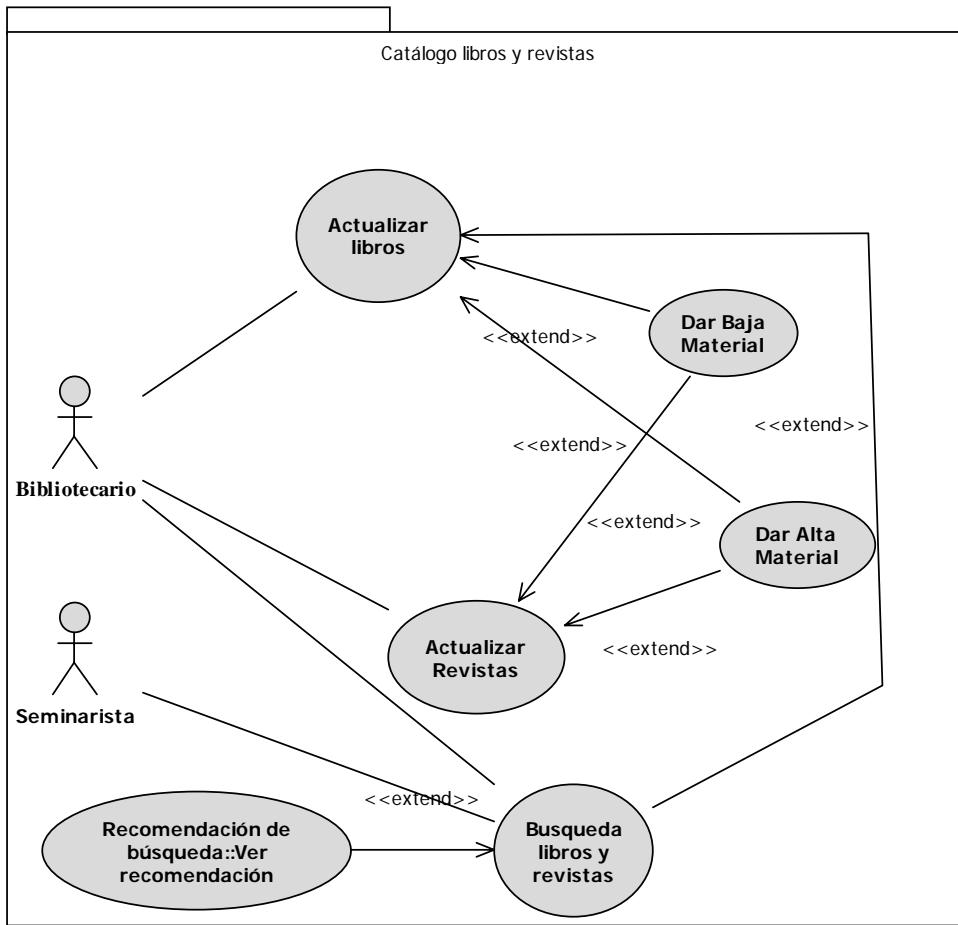


Figura 16: Paquete catálogo libros y revistas.

En la Figura 16 se muestra el diagrama de casos de uso perteneciente al paquete de catálogo de libros y revistas interactuado por el bibliotecario y seminaristas. En el diagrama se puede observar el caso de uso ACTUALIZAR LIBROS en el que se tiene la opción de ingresar un libro, modificar un libro y además de extender a los casos de uso DAR BAJA MATERIAL y DAR ALTA MATERIAL que a su vez son extendidos por ACTUALIZAR REVISTAS el que tiene la opción de Ingresar y Modificar revista. Para realizar la búsqueda del material existe BÚSQUEDA LIBROS Y REVISTAS, el que extiende el caso de uso VER RECOMENDACIÓN del paquete recomendación de búsqueda.

### **CASO DE USO: ACTUALIZAR LIBROS.**

**Actores**

: Bibliotecario.

**Propósito**

: Mantener actualizados los libros de la biblioteca.

**Tipo**

: Primario y esencial.

**Referencias cruzadas**

: Requerimientos R1, R2, R3.

**Descripción**

:

El bibliotecario, cuando desea actualizar un libro debe poseer una cierta cantidad de información que el sistema le solicitará, ésta será aún mayor al momento de hacer ingreso de material, el bibliotecario contará con las siguientes opciones:

- Ingresar Libro.
- Modificar Libro.
- Dar Baja Libro.
- Dar Alta Libro.

### **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario selecciona la opción que desea para mantener actualizada la información de un libro o revista.	
	Valida la opción elegida por el bibliotecario y despliega opción correspondiente.
	Deriva a la sección o caso de uso extendido según corresponda a la elección del

	<p>bibliotecario.</p> <ul style="list-style-type: none"><li>- <i>Ingresar libro:</i> Deriva a la sección ingresar libro.</li><li>- <i>Modificar libro:</i> Deriva a la sección modificar libro.</li><li>- <i>Dar Baja material:</i> Deriva al caso de uso extendido dar baja material.</li><li>- <i>Dar Alta material:</i> Deriva al caso de uso extendido dar alta material.</li></ul>
--	---

## CURSOS ALTERNOS

## **SECCIÓN: INGRESAR LIBRO.**

Descripción :

Si existe un libro que no está ingresado en el sistema, el bibliotecario debe tomar la información necesaria para poder ingresarla y dejar disponible el material a los usuarios de la biblioteca del Seminario. A continuación, se nombran los datos que pertenecen a un Libro.

**DATOS:** Título / ISBN / Cantidad de páginas / Volumen / Tomo / Tipo / Registro anterior / Autor principal / Clasificación / Edición / año publicación / Editorial / Ediciones / Serie / País publicación / Ciudad publicación / Largo / Ancho / Notas / Materias /Autores secundarios.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
	Muestra en pantalla el formulario para ingresar los demás datos.
Digita en el formulario: Título / ISBN / Cantidad de páginas / Volumen / Tomo / Registro anterior / Autor principal / Clasificación / Edición / año publicación / País publicación / Ciudad publicación / Largo / Ancho / Notas. Y selecciona desde las listas correspondientes: Tipo / Serie / Editorial / Ediciones.	
	Verifica que la clasificación ingresada no exista y muestra lista de materias y autores.
Selecciona materias y autores secundarios	
	Ingrera el libro, muestra en pantalla número

	de registro y da la opción de imprimir etiqueta de códigos de barra.
Selecciona imprimir código de barra.	
	Ordena a la impresora que imprima la etiqueta de códigos de barra.

## CURSOS ALTERNOS.

**Ítem 2:** Ingresa un título de libro que ya existe en el sistema. Da a conocer al bibliotecario la existencia de otro libro con el mismo título, él decide si ingresar una copia, ocupar la información del libro que existe para el nuevo libro o ingresar el libro con el mismo título.

**Ítem 2:** En las listas no existe el valor que se desea ingresar para: Tipo / Serie / Editorial / Ediciones. El usuario debe ingresar al sistema el valor para posteriormente escogerlo.

**Ítem 4:** La clasificación que el usuario ingresó ya existe en el sistema. Informa al usuario y no permite el ingreso del libro.

### **CASO DE USO: DAR BAJA MATERIAL.**

**Actores** : Bibliotecario.

**Propósito** : Inhabilitar un libro o revista del sistema.

**Tipo** : Primario y esencial.

**Referencias cruzadas** : Requerimientos R2, R5.

**Descripción** :

Si un libro o revista se estropea, extravía o simplemente es necesario eliminarlo del sistema, podrá ser dado de baja. El bibliotecario deberá indicar el código de registro del material para luego darlo de baja.

### **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario ingresa el código de registro (Manual o pistoleando el código de barras) del libro o revista.	
	Busca el material en la base de datos y muestra formulario en pantalla para indicar razones de la baja.
Indica razones de baja y envía el formulario al sistema.	
	Registra la baja.

## **CURSOS ALTERNOS**

**Ítem 1:** El bibliotecario ingresa un código inválido. Se indica error y muestra nuevamente opción de ingreso del código.

**Ítem 2:** El código ingresado no existe en el sistema. Se indica error y muestra nuevamente opción de ingreso del código.

### **CASO DE USO: DAR ALTA MATERIAL.**

<b>Actores</b>	: Bibliotecario.
<b>Propósito</b>	: Dejar nuevamente activo un libro o revista en el sistema.
<b>Tipo</b>	: Primario y esencial.
<b>Referencias cruzadas</b>	: Requerimientos R2, R5.

**Descripción** :

Un libro o revista después de ser dado de baja, puede nuevamente quedar activo en el sistema, para ello el bibliotecario deberá ingresar el código de registro del material, con el cual el libro o revista automáticamente volverá a estar a disposición de los usuarios.

### **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario ingresa el código de registro (Manual o pistoleando el código de barras) del libro o revista que se desea dar alta.	
	Busca el material en la base de datos y vuelve a preguntar si desea dar alta.
Ingresa respuesta positiva a dar el alta al libro.	
	Registra el alta y deja el libro activo.

## **CURSOS ALTERNOS**

**Ítem 1:** El bibliotecario ingresa un código inválido. Se indica error y muestra nuevamente opción de ingreso del código.

**Ítem 2:** El código ingresado no existe dentro del material que se encuentra de baja. Se indica error y muestra nuevamente opción de ingreso del código.

**Ítem 3:** Ingresa respuesta negativa. El sistema cancela el proceso.

### **CASO DE USO: ACTUALIZAR REVISTAS.**

<b>Actores</b>	: Bibliotecario.
<b>Propósito</b>	: Mantener actualizado el material de la Biblioteca.
<b>Tipo</b>	: Primario y esencial.
<b>Referencias cruzadas</b>	: Requerimientos R4, R5, R6.
<b>Descripción</b>	:
El bibliotecario, cuando desea actualizar una revista debe manejar una cierta cantidad de información que el sistema le solicitará, dependiendo de la acción que el bibliotecario seleccione. Las opciones se listan a continuación.	
<ul style="list-style-type: none"><li>• Ingresar Revista.</li><li>• Modificar Revista.</li><li>• Dar Baja Revista.</li><li>• Dar Alta Revista.</li></ul>	

### **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario selecciona la opción que desea para mantener actualizada la información de una revista.	
	Valida la opción elegida por el bibliotecario y despliega opción correspondiente.
	Deriva a la sección o caso de uso extendido según corresponda a la elección del

	<p>bibliotecario.</p> <ul style="list-style-type: none"><li>- <i>Ingresar revista:</i> Deriva a la sección ingresar revista.</li><li>- <i>Modificar revista:</i> Deriva a la sección modificar revista.</li><li>- <i>Dar Baja material:</i> Deriva al caso de uso extendido dar baja material.</li><li>- <i>Dar Alta material:</i> Deriva al caso de uso extendido dar alta material.</li></ul>
--	---

## CURSOS ALTERNOS

## **SECCIÓN: INGRESAR REVISTA.**

**Descripción** :

Si existe una revista que no está ingresada en el sistema, el bibliotecario debe tomar la información necesaria para ingresarla y dejarla disponible a los usuarios de la biblioteca del Seminario. El bibliotecario deberá seleccionar el título de la revista a la cual desea ingresar un nuevo número, en caso que la revista que desea seleccionar no exista en el sistema, deberá ingresar algunos datos adicionales.

**DATOS REVISTA:** Periodicidad (Fecha de edición y caducación) / número / volumen / año / contenido.

**DATOS ADICIONALES NUEVA REVISTA:** Nombre / fecha fundación / Periodicidad (cantidad de días) / ISSN.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
Selecciona nombre de la revista a la cual desea ingresar un nuevo número.	
	Muestra en pantalla el formulario para ingresar los demás datos.
Digita en el formulario: Periodicidad / número / volumen / año.	
	Verifica que los datos correspondan al formato que se pide e ingresa los datos al sistema con un número de registro.
Selecciona el contenido de la revista	

	Vincula el contenido a la revista ingresada.
	Ingresa la revista al sistema y muestra en pantalla el número de registro dando la opción de imprimir el código de barras.
Selecciona imprimir código de barras.	
	Ordena a la impresora que imprima la etiqueta de códigos de barra.

## CURSOS ALTERNOS.

**Ítem 1:** La revista que desea seleccionar no existe. En ese instante debe ingresar los “datos adicionales nueva revista”, para posteriormente seleccionar la revista en el ítem 1.

**Ítem 4:** Ingreso incorrecto según formato. Se indica error y no permite crear la revista hasta que ingrese los valores correctos.

**Ítem 5:** El contenido de la revista no existe en la lista. Debe ingresarla para posteriormente ser elegido.

## ***CASO DE USO: BÚSQUEDA LIBROS Y REVISTAS.***

<b>Actores</b>	: Bibliotecario, Seminarista.
<b>Propósito</b>	: Encontrar material por medio de criterios de búsqueda.
<b>Tipo</b>	: Primario y esencial.
<b>Referencias cruzadas</b>	: Requerimiento R18.

**Descripción** :

Si un seminarista o bibliotecario necesita realizar la búsqueda de material, primero debe especificar qué tipo de material es, ya sea libro o revista, luego debe seleccionar el criterio de búsqueda que estime conveniente. Los criterios de búsqueda se mencionan a continuación:

**PARA LIBRO:** Título / Autor / Materia / Editorial / Serie / Ediciones / ISBN / Registro material.

**PARA REVISTA:** Nombre / Contenido / ISSN / Fechas / Número de revista.

Luego de escoger el criterio de búsqueda, se debe digitar la palabra que se desea buscar, para que el sistema arroje los resultados encontrados con la información más importante del libro.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario o seminarista indica al sistema si lo que se desea buscar es un libro o revista.	
	Muestra los criterios de búsqueda asociados al material (Libro o revista)
Selecciona el criterio de búsqueda, digita	

palabras a buscar y envía al sistema.	
	Según el criterio escogido y la palabra digitada, realiza la búsqueda.
	Muestra en pantalla las coincidencias de la búsqueda.

## CURSOS ALTERNOS

**Ítem 3:** Envía formulario sin escribir nada. Indica error.

**Ítem 5:** No se encuentran coincidencias, se indica que no hubo éxito en la búsqueda.

## *MODELO CONCEPTUAL*

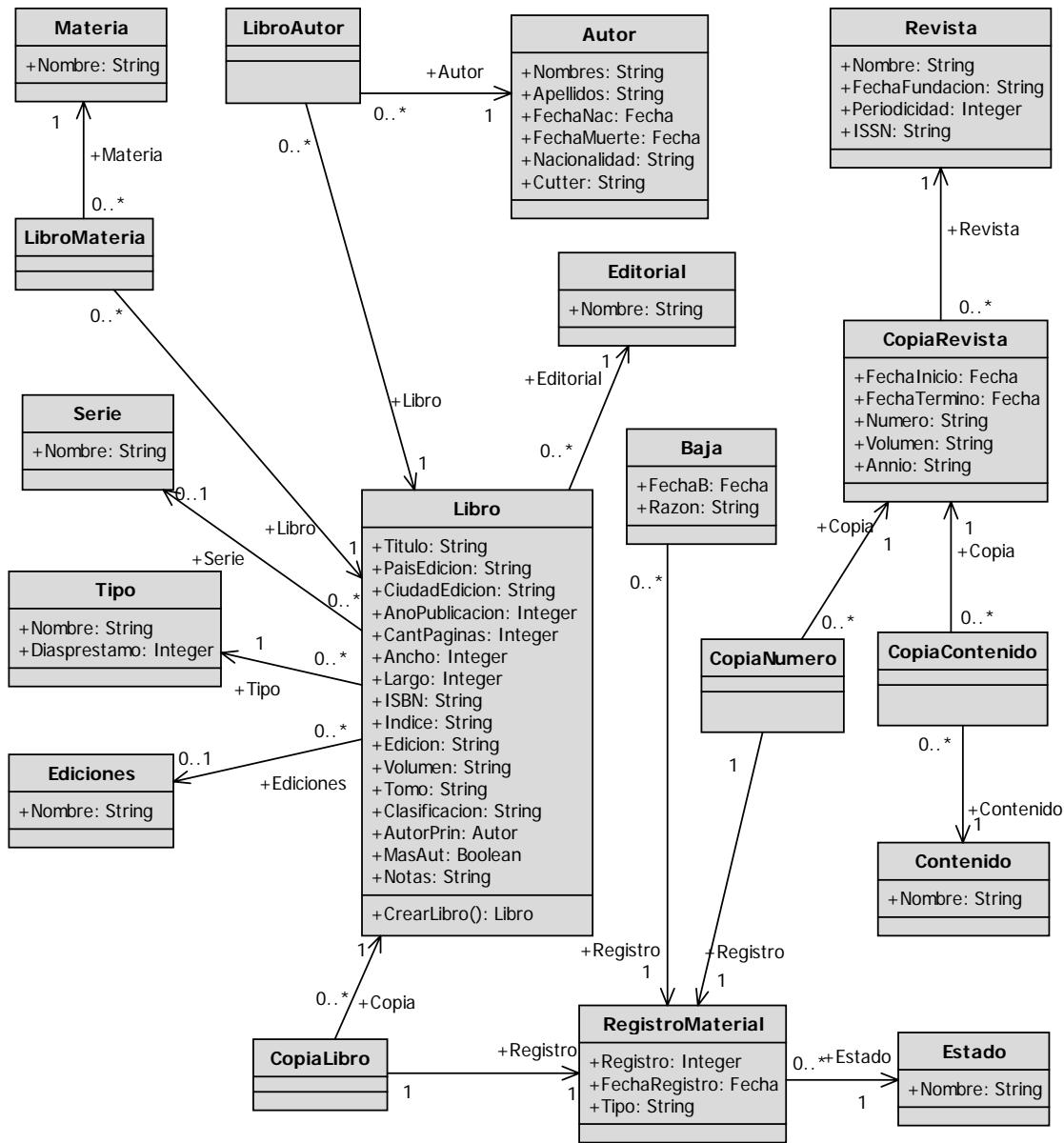


Figura 17: Modelo conceptual paquete Catálogo libros y revistas.

## PAQUETE ADMINISTRACIÓN DE USUARIOS.

### DIAGRAMA DE CASOS DE USO

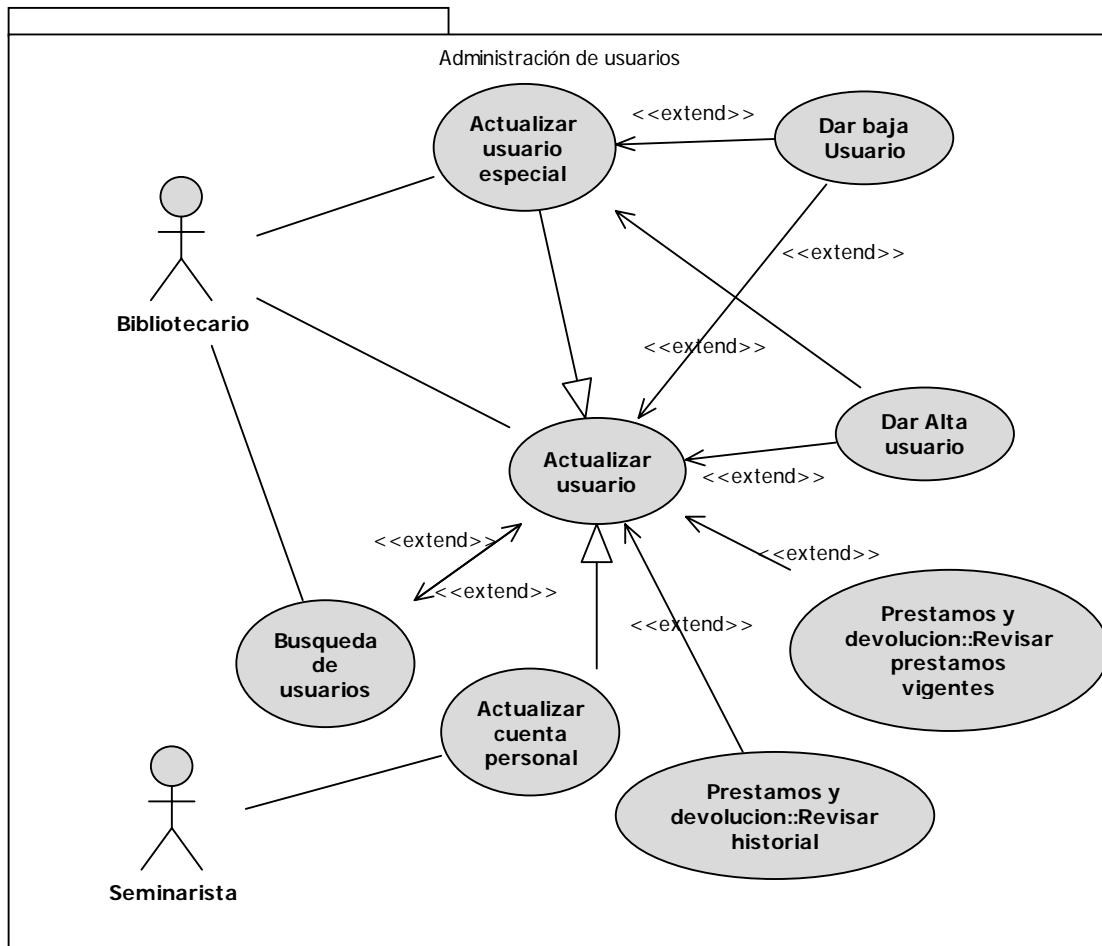


Figura 18: Paquete Administración de usuarios.

En la Figura 18 se muestra el diagrama de casos de uso perteneciente al paquete de administración de usuarios interactuado por el bibliotecario y seminaristas. En el diagrama se puede observar el caso de uso ACTUALIZAR USUARIO en el que se tiene la opción de ingresar un usuario (bibliotecario o seminarista), modificar un usuario (bibliotecario o seminarista) y extiende DAR BAJA USUARIO y DAR ALTA USUARIO a estos también los extiende ACTUALIZAR USUARIO ESPECIAL el que tiene la opción de ingresar usuario externo y modificar usuario externo. También se observa ACTUALIZAR CUENTA PERSONAL en a que un seminarista puede modificar sus datos personales. Para realizar la búsqueda de un usuario existe BÚSQUEDA DE USUARIOS. En el diagrama se puede apreciar que el caso de uso

Actualizar usuario extiende a los casos de uso REVISAR HISTORIAL Y REVISAR PRÉSTAMOS VIGENTES pertenecientes al paquete de préstamos y devolución.

## **CASO DE USO: ACTUALIZAR USUARIO.**

<b>Actores</b>	: Bibliotecario.
<b>Propósito</b>	: Mantener actualizada la información de los usuarios.
<b>Tipo</b>	: Primario y esencial.
<b>Referencias cruzadas</b>	: Requerimientos R7, R8, R9, R10, R11, R12.
<b>Descripción</b>	:  Para poder actualizar una cuenta de usuario, el bibliotecario deberá en primera instancia elegir qué tipo de usuario desea actualizar, ya sea seminarista o bibliotecario para luego tener la opción de crear, modificar, dar de baja y dar de alta al usuario que desee.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario selecciona la opción que desea para actualizar la información relacionada a un usuario.	
	Valida la opción elegida por el bibliotecario y despliega opciones existentes.
	Deriva a la sección o caso de uso extendido según corresponda a la elección del bibliotecario.  - <i>Ingresar usuario:</i> Deriva a la sección ingresar usuario.  - <i>Modificar Usuario:</i> Deriva a la

	<p>sección modificar usuario.</p> <ul style="list-style-type: none"><li>- <i>Dar Baja Usuario:</i> Deriva al caso de uso extendido dar baja usuario.</li><li>- <i>Dar Alta Usuario:</i> Deriva al caso de uso extendido dar alta usuario.</li></ul>
--	---

## CURSOS ALTERNOS.

## **SECCIÓN: INGRESAR USUARIO.**

**Descripción** :

Para crear un usuario, el bibliotecario debe en primera instancia indicar el tipo de usuario que desea ingresar, estos pueden ser seminarista y bibliotecario, además debe tener en su poder una serie de datos que deben ser ingresados en el sistema para crear la cuenta, la información asociada a cada usuario se lista a continuación.

**PARA SEMINARISTA:** Nombres / Apellidos / RUT / Fecha de Nacimiento / Teléfono / Celular / Parroquia / Diócesis / Año de ingreso / Curso

**PARA BIBLIOTECARIO:** Nombres / Apellidos / RUT / Fecha de Nacimiento / Teléfono

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
Indica al sistema si lo que se desea ingresar es un seminarista o bibliotecario.	
	Muestra en pantalla el formulario correspondiente al tipo de usuario que se desea ingresar.
Ingresa los datos correspondientes al usuario a ingresar.	
	Valida que el RUT ingresado no exista en el sistema.
	Crea la nueva cuenta y la deja activa para su uso.

## **CURSOS ALTERNOS.**

**Ítem 3:** El RUT ya existe en el sistema. Indica que el usuario que desea ingresar ya está en el sistema.

### **CASO DE USO: DAR BAJA USUARIO.**

<b>Actores</b>	: Bibliotecario.
<b>Propósito</b>	: Dejar inactivo a un usuario del sistema de todo acto.
<b>Tipo</b>	: Primario y esencial.
<b>Referencias cruzadas</b>	: Requerimientos R9, R12, R16.
<b>Descripción</b>	:
	<p>Si por alguna razón un usuario del sistema, ya sea seminarista, bibliotecario o un usuario especial, debe perder su condición de usuario, el bibliotecario deberá ingresar que tipo de usuario es el que se desea dar de baja, luego ingresar el RUT, y por último indicar cuál es la razón de la baja.</p>

### **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario ingresa que tipo de usuario desea dar baja (seminarista, bibliotecario, usuario especial) y luego el RUT correspondiente al usuario.	
	Busca al usuario y muestra formulario en pantalla para ingresar las razones de baja.
Indica razones de baja y envía el formulario.	
	Registra la baja.

### **CURSOS ALTERNOS**

**Ítem 1:** El bibliotecario ingresa un RUT inválido. Se indica error y muestra nuevamente opción de ingreso de RUT.

**Ítem 1:** El bibliotecario ingresa el RUT asociado a su cuenta. Indica error y no permite la baja.

**Ítem 2:** El RUT ingresado no existe en el sistema. Se indica error y muestra nuevamente opción de ingreso de RUT.

Ítem 2: El RUT ingresado ya se encuentra de baja. Se indica error y cancela el proceso.

*MODELO CONCEPTUAL*

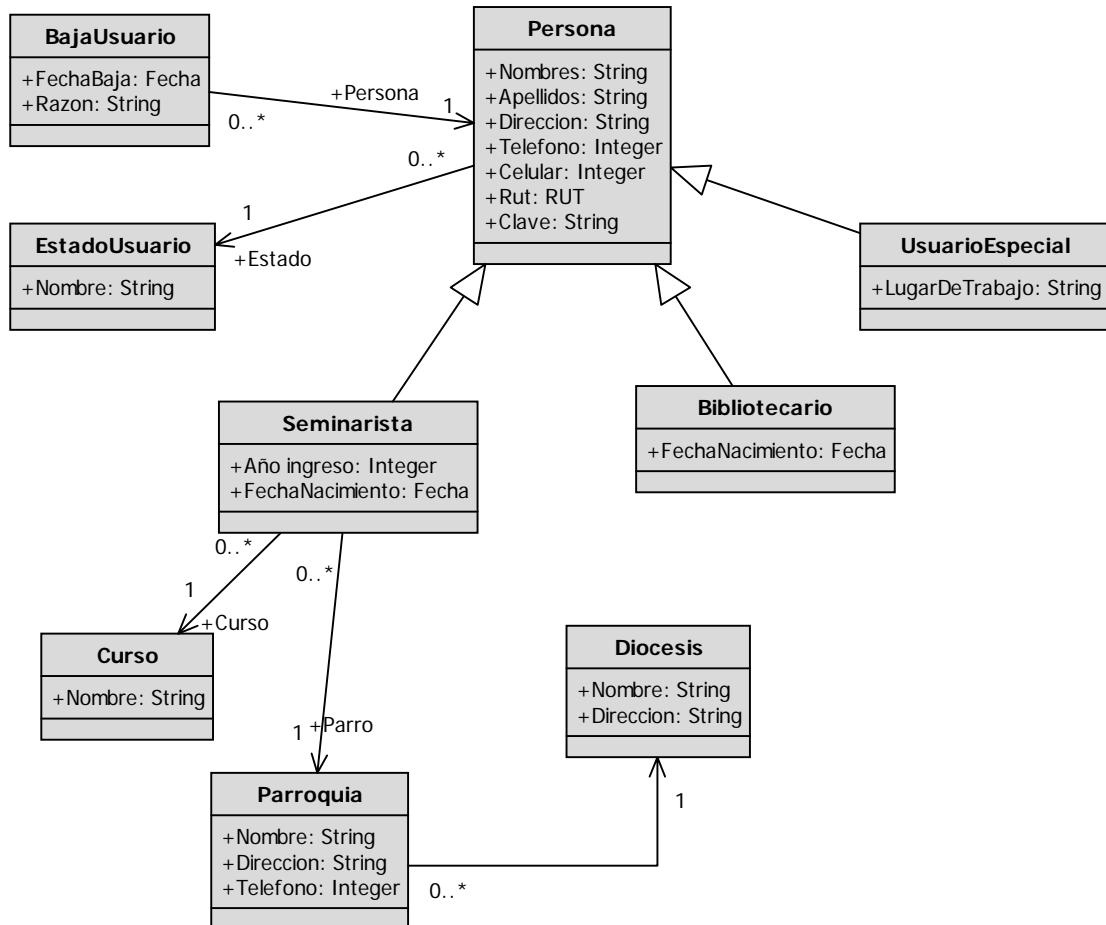


Figura 19: Modelo conceptual del paquete Administración de Usuarios.

## PAQUETE PRÉSTAMOS Y DEVOLUCIONES.

### DIAGRAMA DE CASOS DE USO

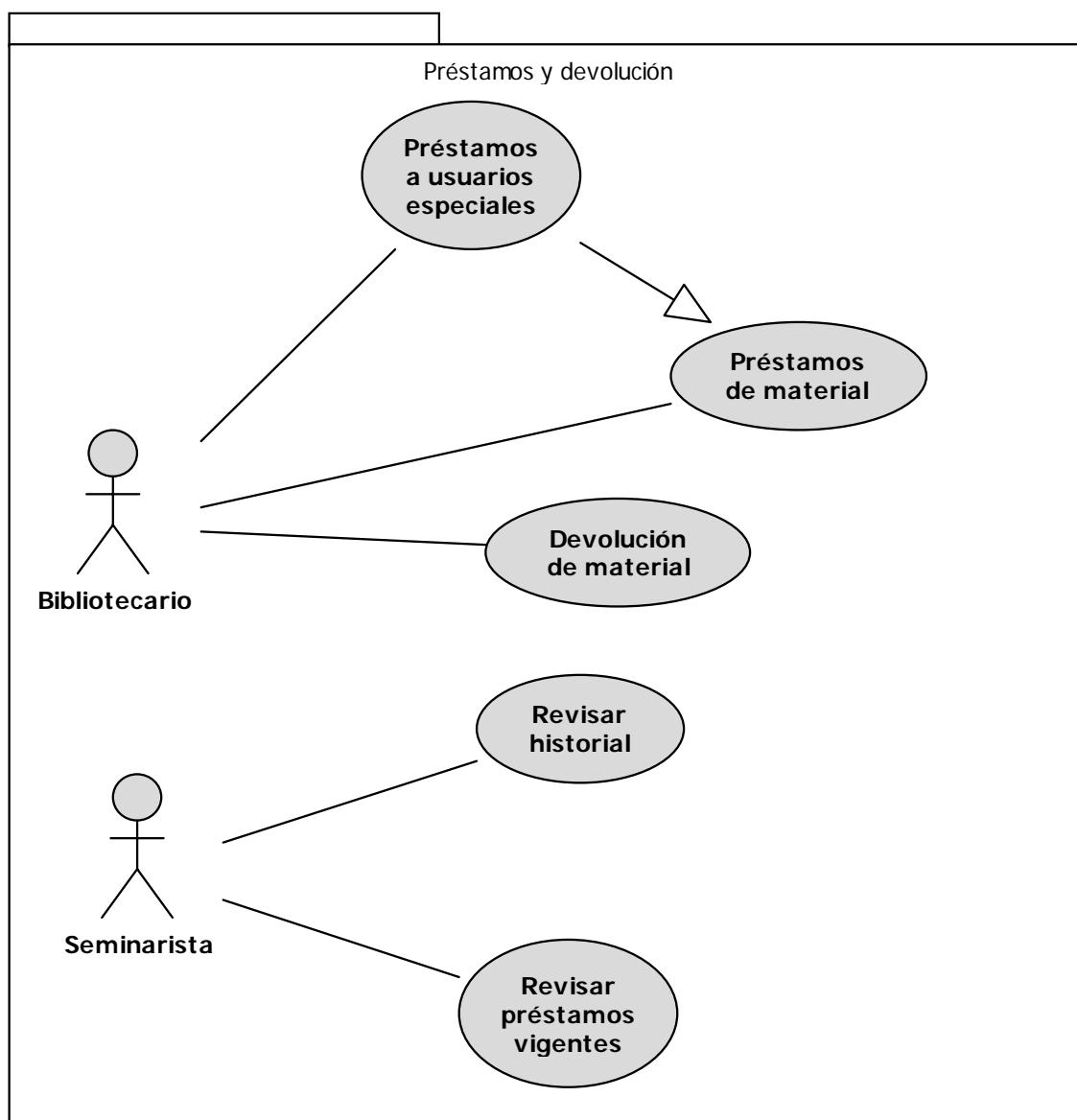


Figura 20: Paquete préstamos y devoluciones.

En la Figura 20 se muestra el diagrama de casos de uso perteneciente al paquete de préstamos y devoluciones interactuado por el bibliotecario, seminaristas y usuarios especiales. En el diagrama se puede observar el caso de uso PRÉSTAMOS DE MATERIAL que a su vez posee un caso

especial como PRÉSTAMOS A USUARIOS ESPECIALES. Después que se produce un préstamo de material, debe ser devuelto mediante el caso de uso DEVOLUCIÓN DE MATERIAL. Un seminarista como usuarios especiales tiene acceso a REVISAR HISTORIAL y a REVISAR PRÉSTAMOS VIGENTES.

### **CASO DE USO: PRÉSTAMOS DE MATERIAL.**

<b>Actores</b>	: Bibliotecario.
<b>Propósito</b>	: Realizar préstamo del material de la biblioteca (libros y revistas).
<b>Tipo</b>	: Primario y esencial.
<b>Referencias cruzadas</b>	: Requerimiento R13

**Descripción** :

El bibliotecario, una vez que recibe una petición de préstamo por parte del seminarista, lo ingresa al sistema identificándolo con su RUT. Luego, el seminarista le indica el código de clasificación del libro o revista, para que este lo busque en los estantes y lo registre con el código de barras, y así registrar el préstamo. Los días de préstamo corresponden a los siguientes criterios.

Libro de **referencia** (Libros más solicitados): 1 día de préstamo.

Libro de **No referencia**: 1 mes de préstamo.

Revista: 1 día de préstamo.

Es importante mencionar que todo préstamo debe quedar con la opción de cambiar la fecha de devolución, según el criterio del bibliotecario.

### **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario ingresa el RUT del usuario que desea realizar el préstamo.	
	Verifica que el usuario se encuentre en

	condición regular (sin atraso de devolución) para realizar el préstamo.
Ingresa el código de registro del libro o revista (ya sea de forma manual o pistoleando el código de barras).	
	Genera el préstamo e indica la fecha de devolución del libro o revista.

## CURSOS ALTERNOS

**Ítem 1:** Ingreso del RUT erróneo o inexistente. El sistema indica error y muestra nuevamente la opción de ingresar el RUT del seminarista.

**Ítem 2:** La verificación indica que el usuario no puede realizar préstamo. Se cancela el préstamo en curso y se indica la fecha en que el usuario estará en condición regular.

**Ítem 3:** Se ingresa código inexistente, El sistema indica error, queda a la espera de otro código...

**Ítem 4:** Si el bibliotecario desea cambiar la fecha de devolución que el sistema genera, puede hacer el cambio.

## **CASO DE USO: DEVOLUCIÓN DE MATERIAL.**

<b>Actores</b>	: Bibliotecario.
<b>Propósito</b>	: Realizar la devolución de un libro o revista.
<b>Tipo</b>	: Primario y esencial.
<b>Referencias cruzadas</b>	: Requerimiento R14

### **Descripción :**

Para concretar una devolución, se debe ingresar el código del libro, y en caso de que el libro se entregue fuera de la fecha establecida, el usuario vinculado al libro quedará suspendido de realizar préstamos por los días equivalentes a los días de atraso, es decir, por cada día de atraso se suspende un día de la posibilidad de realizar préstamos. La suspensión del usuario se manifestará por el estado en que se mantendrá, INACTIVO.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario ingresa el código del libro al sistema para efectuar la devolución (ya sea de forma manual o pistoleando el código de barras).	
	Vincula el libro al préstamo al que pertenece, para registrar la devolución del seminarista al que está vinculado y verificar que se entregó dentro de la fecha estipulada.
	Informa la devolución satisfactoria.

## **CURSOS ALTERNOS**

**Ítem 1:** Se ingresa un código que no pertenece a un préstamo. Se indica error, y se cancela el proceso.

**Ítem 2:** El libro devuelto se entrega fuera del plazo de entrega. Se indica error y se genera sanción de acuerdo a los días de atraso indicando en pantalla.

## **CASO DE USO: REVISAR PRÉSTAMOS VIGENTES.**

<b>Actores</b>	: Seminarista, Bibliotecario.
<b>Propósito</b>	: Revisar los préstamos Vigentes que posee usuario.
<b>Tipo</b>	: Secundario y esencial.
<b>Referencias cruzadas</b>	: Requerimiento R13.
<b>Descripción</b>	:
	Un usuario puede revisar todos los préstamos vigentes que mantiene en la biblioteca. Estos aparecerán en una lista indicando la fecha de préstamo y la fecha tope en que debería ser devuelto el material.
	Un bibliotecario también puede revisar los préstamos vigentes de un usuario, cuando está actualizando sus datos.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el seminarista o bibliotecario desean ver los préstamos vigentes, en el caso del bibliotecario debe indicar el RUT del usuario al que desea ver los préstamos vigentes.	
	Busca los préstamos vigentes en la base de datos.
	Muestra en pantalla todos los préstamos vigentes que posee.

## **CURSOS ALTERNOS**

Ítem 3: El usuario no posee préstamos vigentes. Indica que no posee préstamos vigentes.

## MODELO CONCEPTUAL

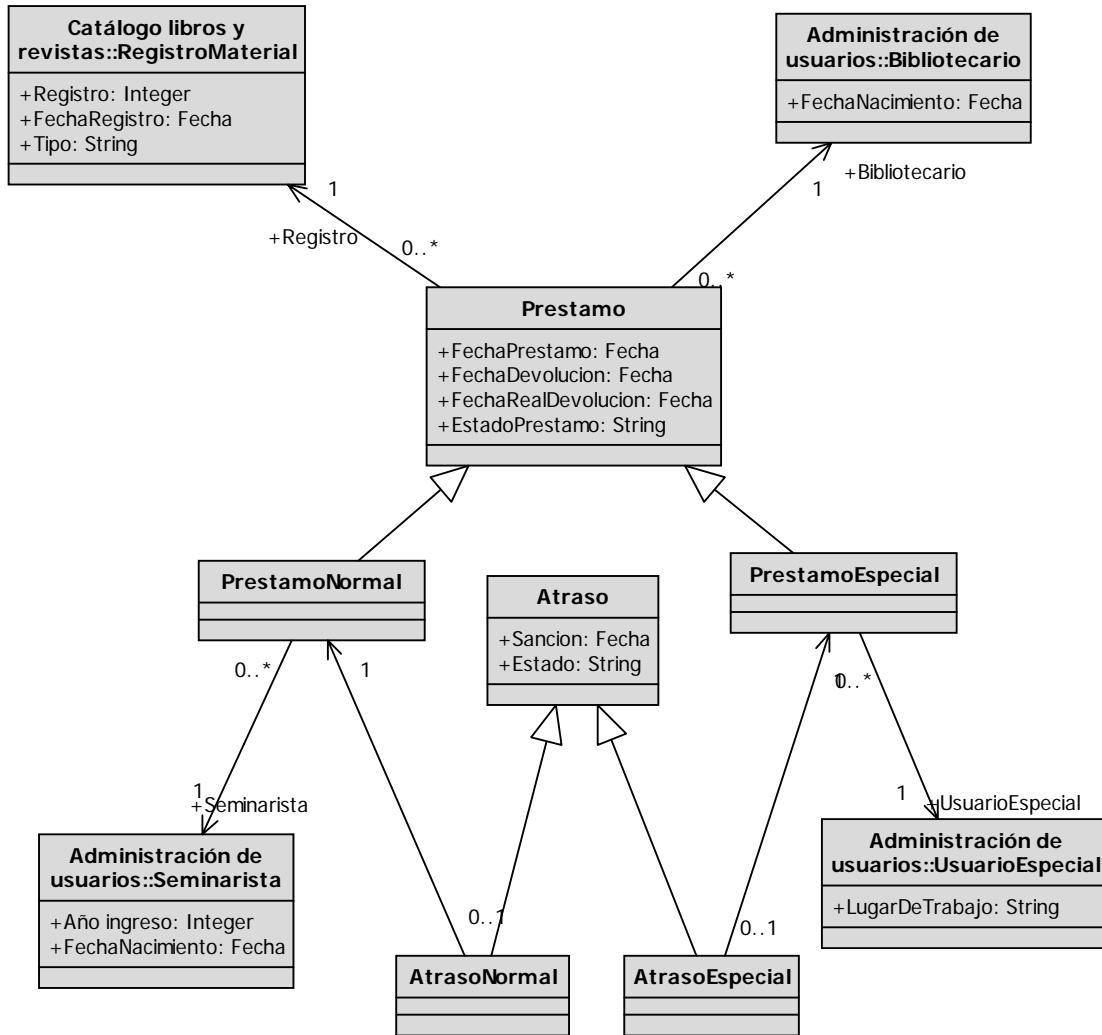


Figura 21: Modelo conceptual del paquete Préstamos y devoluciones

## **PAQUETE PUBLICACIONES DIGITALES.**

### *DIAGRAMA DE CASOS DE USO*

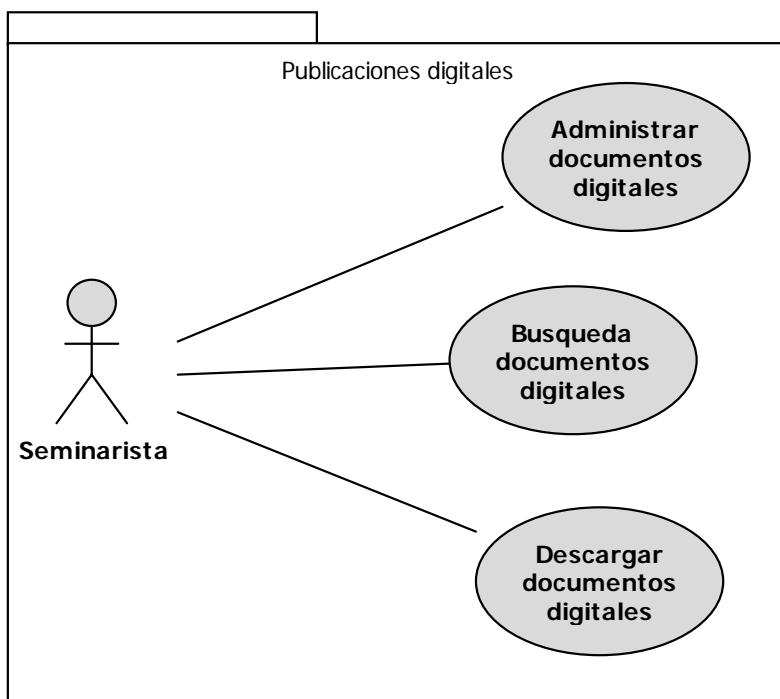


Figura 22: Paquete Publicaciones digitales.

En la Figura 22 se muestra el diagrama de casos de uso perteneciente al paquete de publicaciones digitales interactuado por seminaristas. En el diagrama se puede observar el caso de uso ADMINISTRAR DOCUMENTOS DIGITALES en el que se tiene la opción de ingresar archivo, eliminar archivo y modificar propiedades. Luego se aprecia BÚSQUEDA DOCUMENTOS DIGITALES para buscar con más exactitud algún archivo y finalmente se aprecia DESCARGAR DOCUMENTOS DIGITALES para poder observar el contenido del material.

## *MODELO CONCEPTUAL*

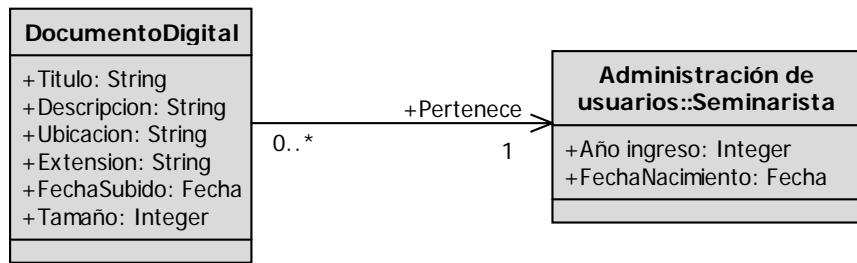


Figura 23: Modelo conceptual del paquete Publicaciones Digitales

## PAQUETE SISTEMA DE MENSAJERÍA.

### DIAGRAMA DE CASOS DE USO

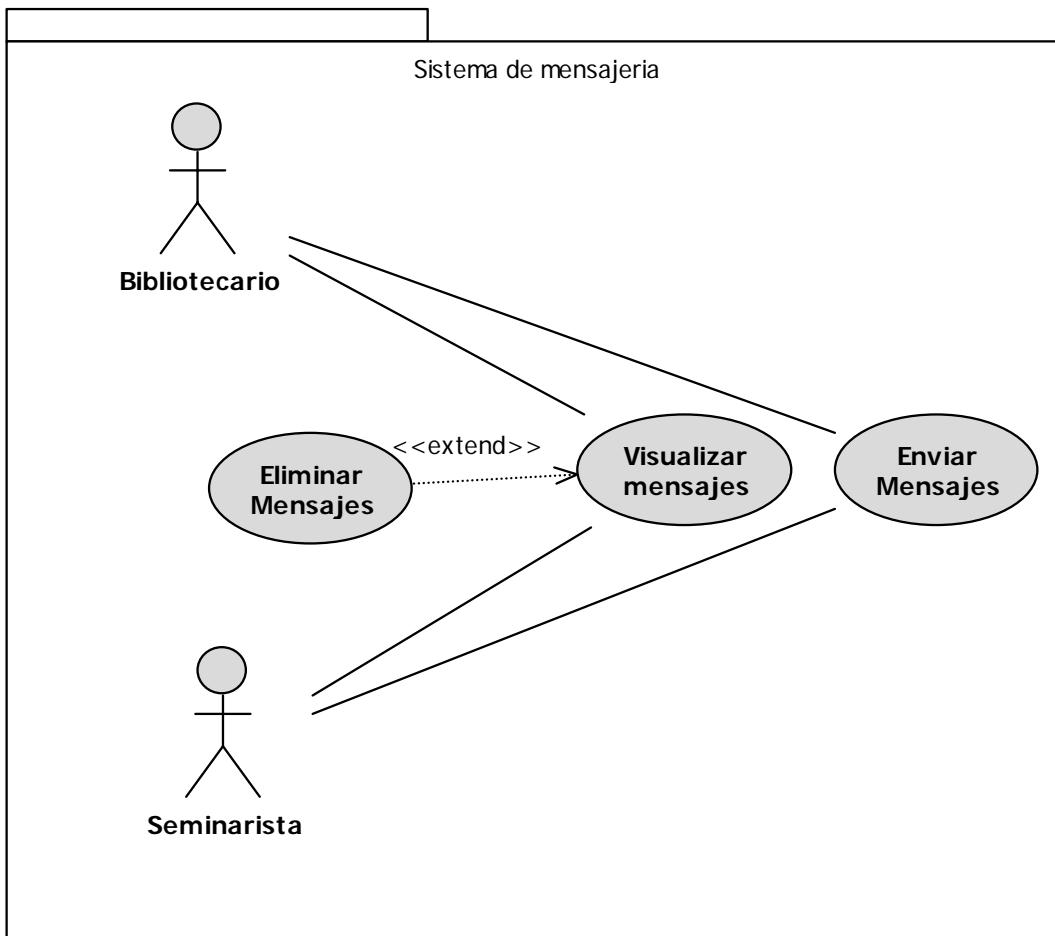


Figura 24: Paquete sistema de mensajería.

En la Figura 24 se muestra el diagrama de casos de uso perteneciente al paquete sistema de mensajería interactuado por el bibliotecario y seminaristas. En el diagrama se puede observar el caso de uso ENVIAR MENSAJES los cuales podrán ser vistos por los receptores mediante el caso de uso VISUALIZAR MENSAJES y si este lo desea podrá ELIMINAR MENSAJES.

*MODELO CONCEPTUAL*

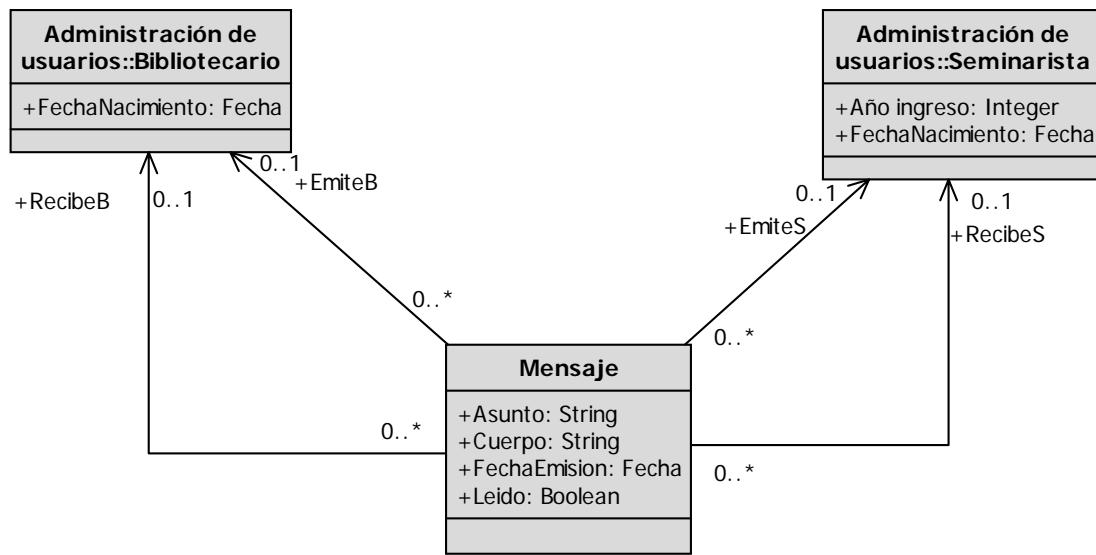


Figura 25: Modelo conceptual del paquete sistema de mensajería.

## **PAQUETE RECOMENDACIÓN DE BÚSQUEDA.**

### *DIAGRAMA DE CASOS DE USO*

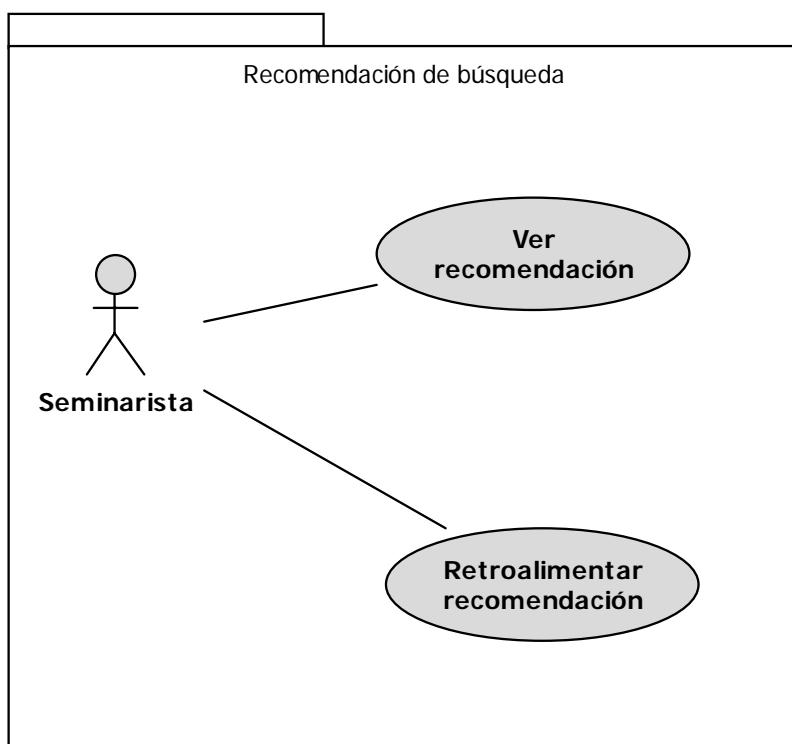


Figura 26: Paquete recomendación de búsqueda.

En la Figura 26 se muestra el diagrama de casos de uso perteneciente al paquete recomendación de búsqueda interactuado por seminaristas. En el diagrama se puede observar el caso de uso VER RECOMENDACIÓN que sirve para tomar la decisión de que libro elegir de la búsqueda que arroja el sistema, para ello cada seminarista puede RETROALIMENTAR RECOMENDACIÓN para que la experiencia de leer un material sea de utilidad para otros.

## *MODELO CONCEPTUAL*

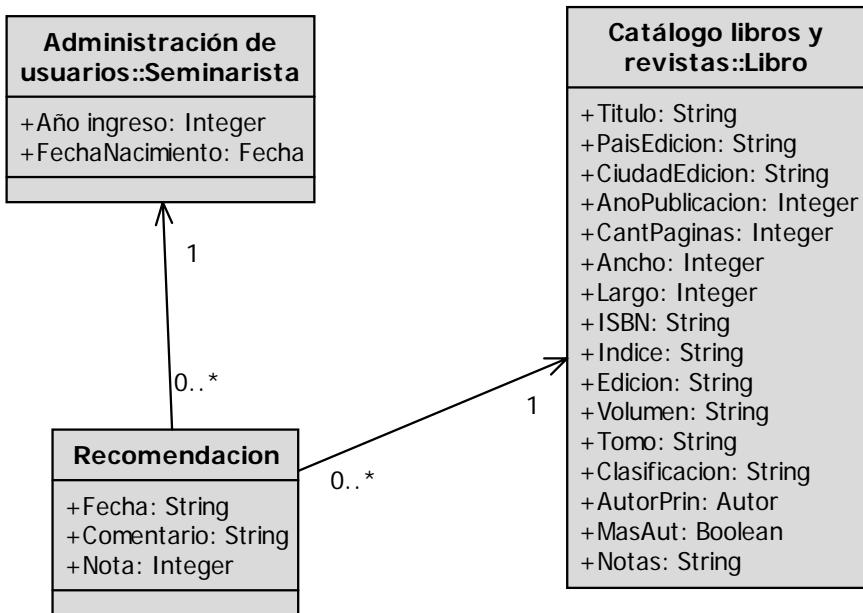


Figura 27: Modelo conceptual del paquete recomendación de búsqueda.

## CAPITULO VI: DISEÑO

El diseño orientado a objetos transforma el modelo de análisis, en un modelo de diseño que sirve como anteproyecto para la construcción de software. (Pressman, 2002).

En este capítulo se presenta el diseño arquitectónico, los principales diagramas de interacción, los diagramas de clases, la jerarquía de menús y finalmente se muestran las principales pantallas de la aplicación de escritorio y la aplicación Web.

## 6.1 DISEÑO ARQUITECTÓNICO

El diseño arquitectónico, define la relación entre los elementos estructurales del software. A continuación se presenta diseño arquitectónico del sistema.

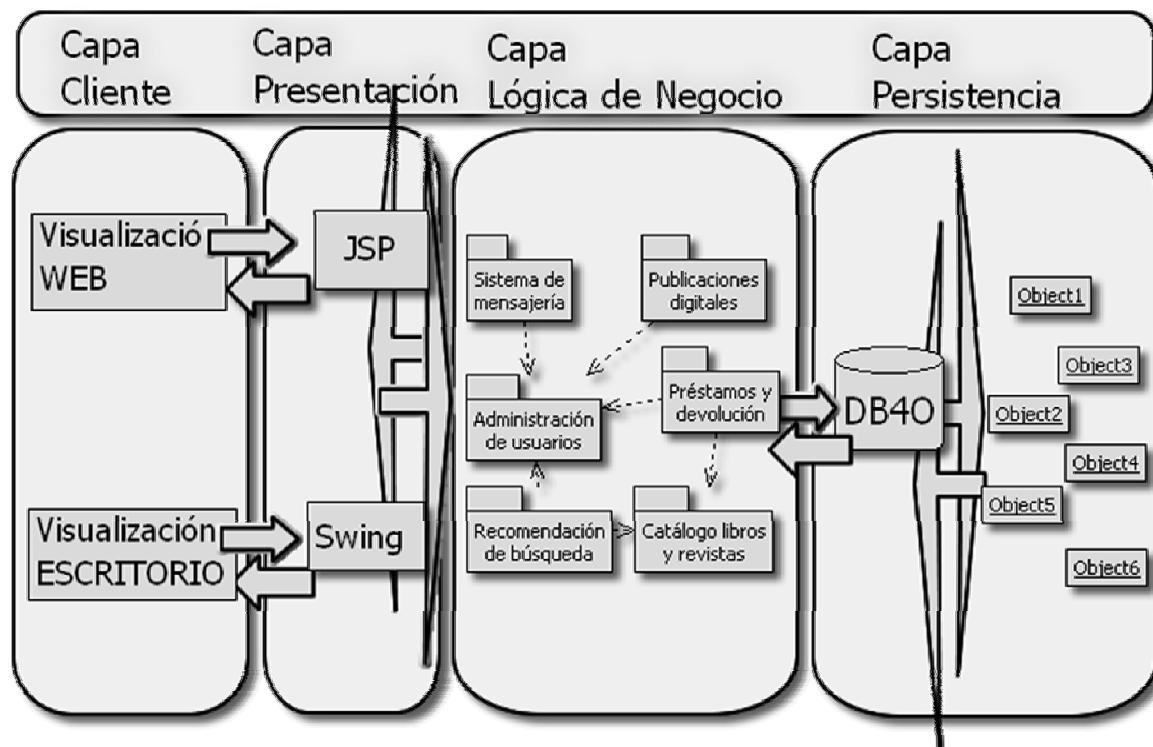


Figura 28: Diseño Arquitectónico

En la Figura 28 se aprecian todos los elementos que componen el sistema bibliotecario organizado en una arquitectura cliente-servidor de cuatro capas.

- La capa cliente corresponde a las interfaces con las que interactúa el cliente, en este caso una interfaz Web y otra de escritorio.
- La capa de presentación prepara datos para su envío a la capa de cliente y procesa solicitudes desde la capa de cliente para su envío a la capa lógica de negocios, como lo hace JSP y los componentes Swing.

- En capa lógica de negocio se realizan las funciones principales de la aplicación: procesamiento de objetos y la implementación de los métodos. Es aquí donde se sitúan los paquetes asociados al sistema.
- La capa de persistencia permite que los objetos perduren en el tiempo, a cargo de DB4o.

## 6.2 DIAGRAMAS DE COLABORACIÓN

Los diagramas de interacción que serán utilizados en el proyecto en cuestión, son los “diagramas de colaboración”, y es necesario mencionar algunos detalles importantes para una mejor comprensión de los diagramas.

### OBSERVACIONES

- Sólo se presentará el diseño de los principales eventos que pertenecen a los paquetes esenciales del sistema, como lo son: “catálogo de libros y revistas”, “administración de usuarios” y “préstamos y devolución”.
- Los métodos que realizan la modificación de datos no serán presentados ya que estos sólo modifican los atributos internos de los objetos. Los métodos que se encargan de realizar las consultas a la base de datos tampoco serán presentadas, debido a que ésta tarea es realizada por la capa de persistencia y no por la aplicación como tal.
- En todos los diagramas se presenta el sentido lógico de la implementación, dejando de lado todas las validaciones que se realizan para lograr un mejor entendimiento de éstos.

### CONSIDERACIONES GENERALES

Para mostrar una instancia de una clase en un diagrama de interacción, se utiliza el símbolo gráfico de una clase, esto es el rectángulo, pero con el nombre subrayado. (Opción A, Figura 29)

Se puede utilizar un nombre para identificar de manera única la instancia. Obsérvese que los “:” preceden el nombre de la clase. (Opción B, Figura 29)

Si los mensajes son enviados a las propias clases, en lugar de una instancia, para invocar a **métodos estáticos** se utiliza un rectángulo de clase cuyo nombre no está subrayado, pero para efectos del proyecto, debido a que no se cuenta con esa opción en la herramienta CASE, el nombre de la clase estará subrayada pero con StereoType <>Clase>>. (Opción C, Figura 29)

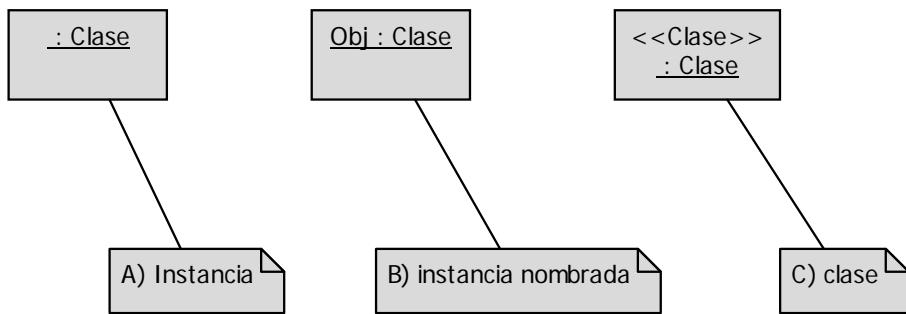


Figura 29: Instancia en un diagrama de Colaboración.

En la mayoría de las clases, existen métodos comunes, por ejemplo por cada propiedad que existe se aprecian los métodos get (extraer) y set (cambiar) seguido del nombre de la propiedad o atributo (ver Figura 30).

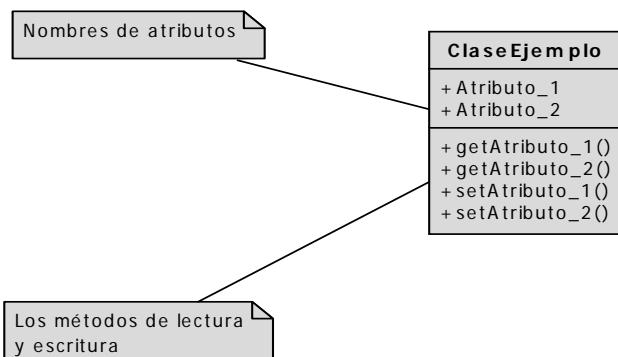


Figura 30: Ejemplo métodos get y set.

También, en las clases existe el método setBD() que es encargado de persistir (guardar) en la base de datos el objeto que lo está invocando, es decir, este método también llama a los métodos que proporciona la base de datos para que pueda ingresar el objeto.

En DB4o (motor de base de datos orientada a objetos utilizada en el proyecto) la forma de guardar un objeto es el que aparece en la Figura 31 (método setBD()):

```

28 | try{
29 |     ObjectContainer cliente =
30 |         Db4o.openClient(Util.HOST,Util.PUERTO,Util.USER,Util.PASS);
31 |     Object obj1 = new Object();
32 |     cliente.set(obj1);
33 |     cliente.commit();
34 | }catch(IOException msg){
35 |     System.out.println(msg.getMessage());
36 |

```

Figura 31: Método setBD ocupado para guardar un objeto.

Entonces lo que hace el método setBD es guardar en la base de datos el Objeto que lo está invocando, es decir realiza un cliente.set(this).

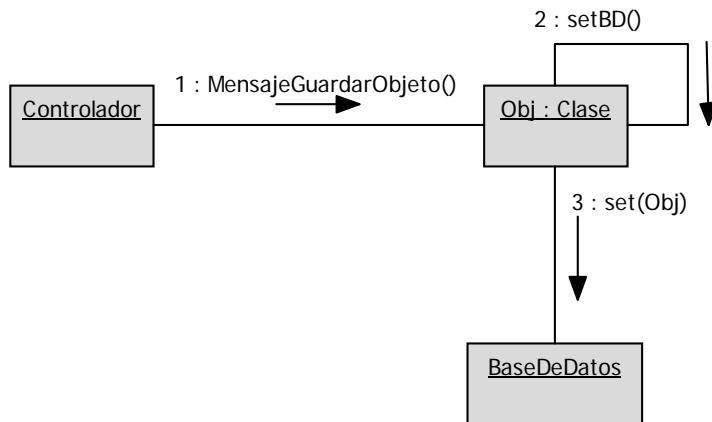


Figura 32: Ejemplo setBD().

En los diagramas, la base de datos será representada por el objeto BaseDeDatos, sobre el cual se realizarán las consultas necesarias mediante el método Consulta(atributo), donde “atributo” es la propiedad por la cual está consultando como se muestra en la Figura 33.

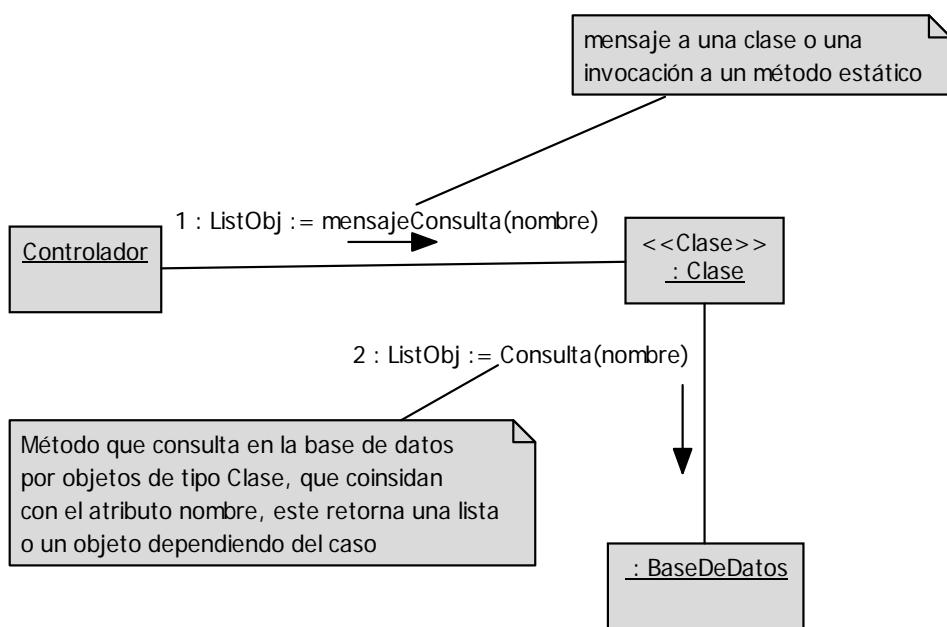


Figura 33: Ejemplo consulta a la base de datos.

Es necesario, dejar en claro que este objeto (BaseDeDatos) y el método Consulta no están implementados (No existe), ya que esto es parte del funcionamiento interno de la base de datos y es sólo una manera de dar a entender el funcionamiento del código como se muestra en la Figura 33. La forma en que se realiza una consulta en la base de datos es mediante una serie de métodos que se dan a entender en la Figura 34.

```

291     public static Libro BuscarPorTitulo(String tit, ObjectContainer cliente) {
292         Libro aux;
293         Query query = cliente.query();
294         query.constrain(Libro.class);
295         query.descend("Titulo").constrain(tit);
296         ObjectSet result = query.execute();
297         while(result.hasNext()){
298             aux = (Libro)result.next();
299             return aux;
300         }
301         return null;
302     }
  
```

Figura 34: Método BuscarPorTitulo de la clase Libro.

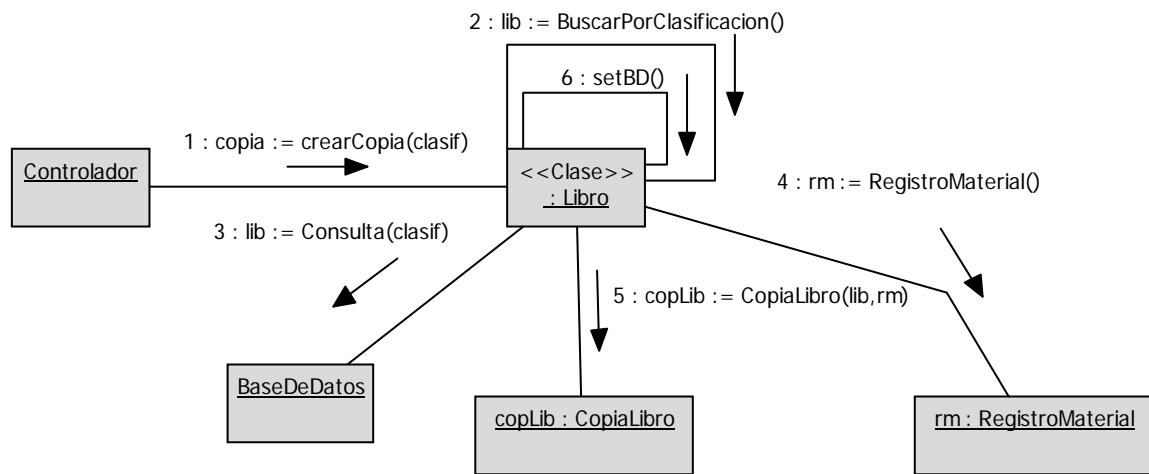
En la Figura 34 se aprecia cómo se consulta a la base de datos por todos los libros que tengan un determinado título, a este método se le envía el parámetro de búsqueda (tit) y la conexión realizada a la base de datos (cliente). En la línea 293 se crea un objeto query asociado a la conexión cliente, el cual luego con el método constrain se indica que la búsqueda se realizará a los objetos pertenecientes a la clase Libro, posteriormente se desciende mediante el método descend a través del atributo Titulo, el que debe coincidir con “tit” mediante el método constrain. Finalmente se realiza la consulta con execute, el que retorna una lista tipo ObjectSet con todos los objetos tipo Libro que coincidieron con el título.

## PAQUETE CATÁLOGO LIBROS Y REVISTAS.

**Caso de uso:** Actualizar Libros.

**Sección:** ingresar libro.

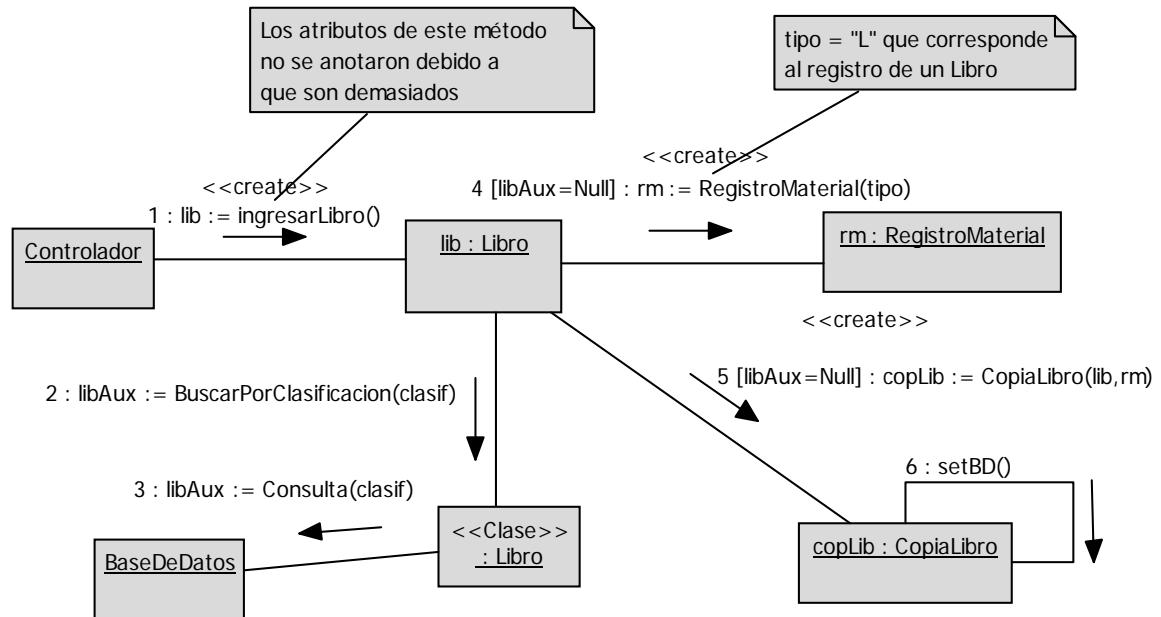
**Mensaje:** crearCopia()



**Caso de uso:** Actualizar Libros.

**Sección:** ingresar libro.

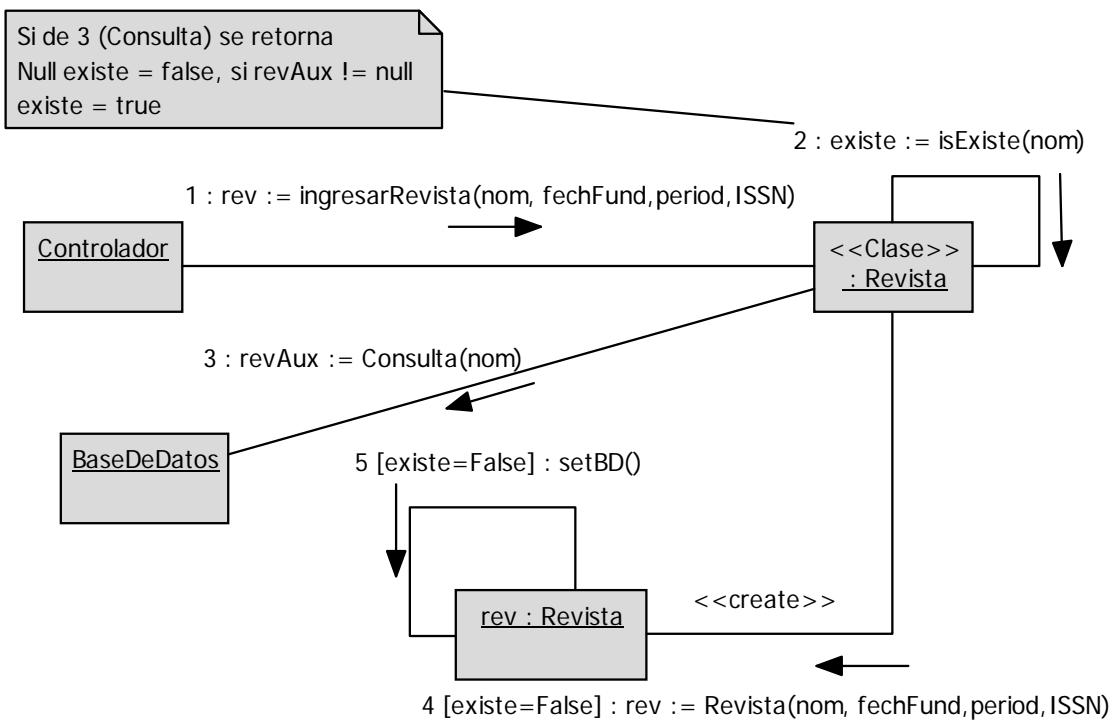
**Mensaje:** ingresarLibro()



**Caso de uso:** Actualizar Revistas.

**Sección:** ingresar revista.

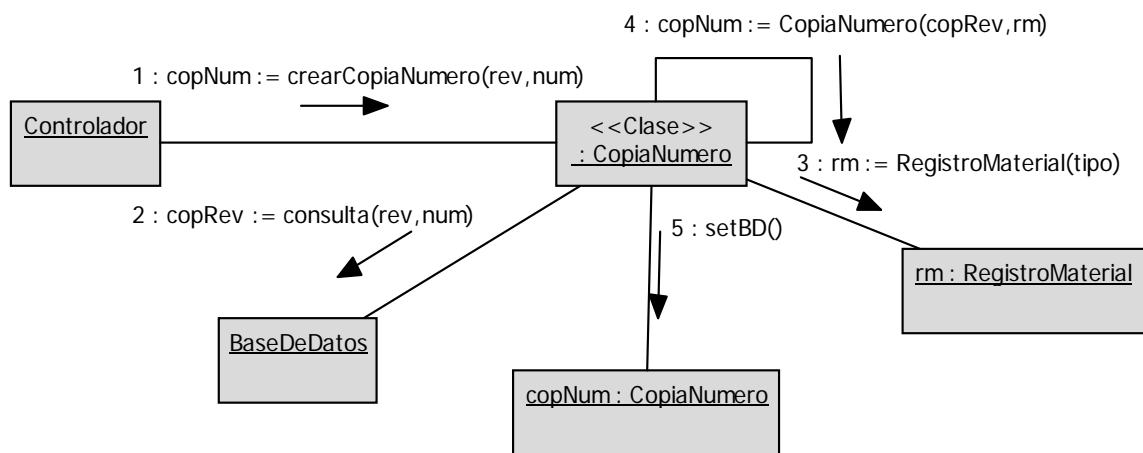
**Mensaje:** ingresarRevista()



**Caso de uso:** Actualizar Revistas.

**Sección:** ingresar revista.

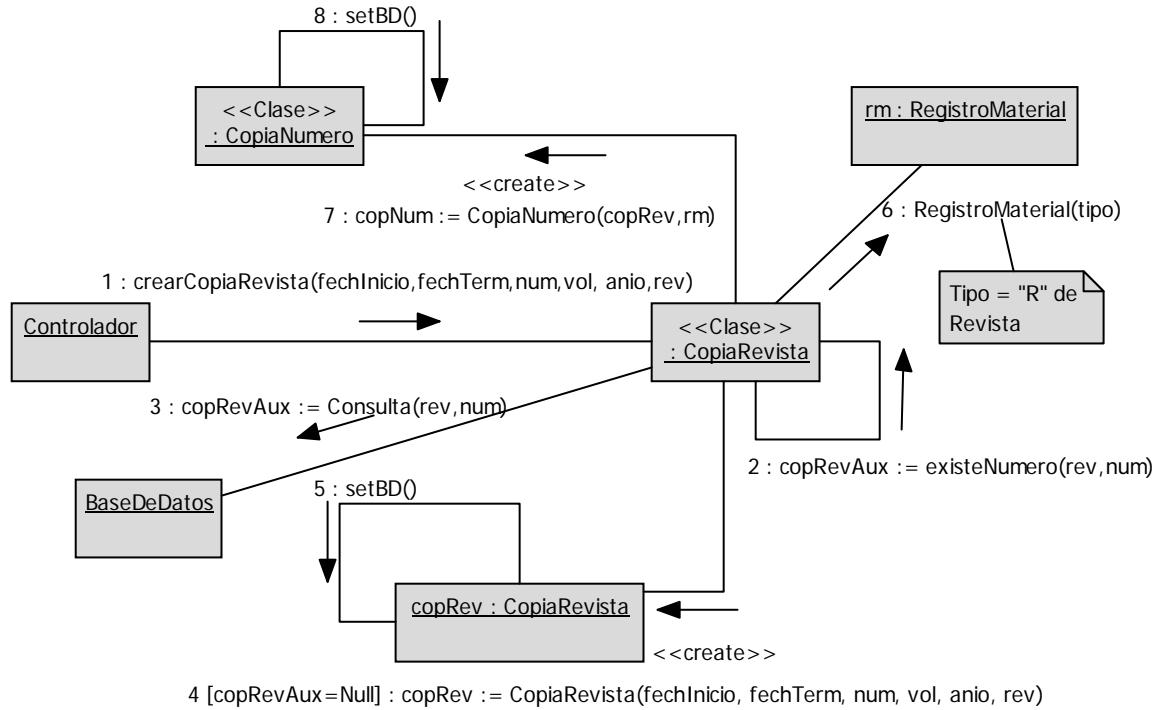
**Mensaje:** CrearCopiaNumero()



**Caso de uso:** Actualizar Revistas.

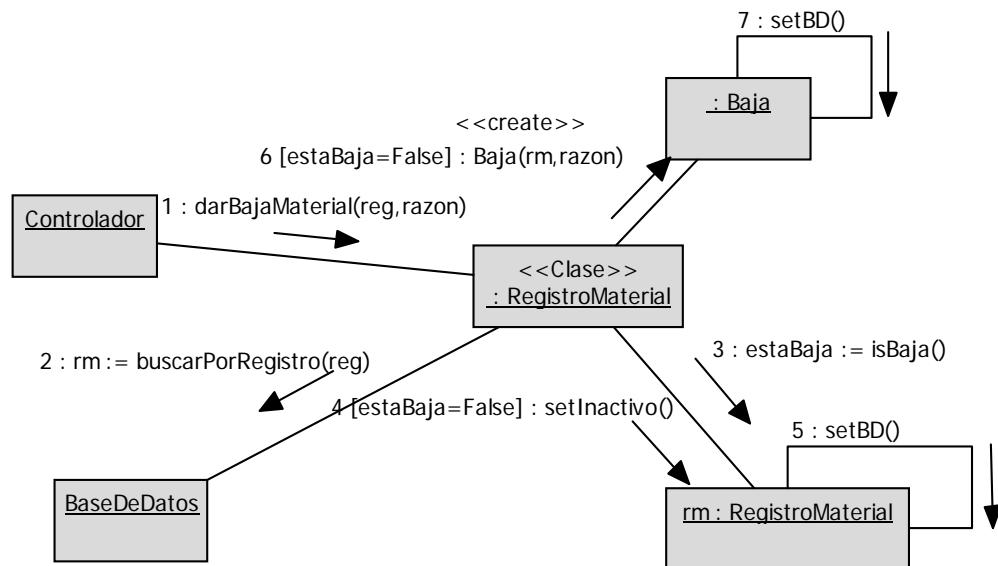
**Sección:** ingresar revista.

**Mensaje:** crearCopiaRevista()



**Caso de uso:** Dar baja material.

**Mensaje:** darBajaMaterial()

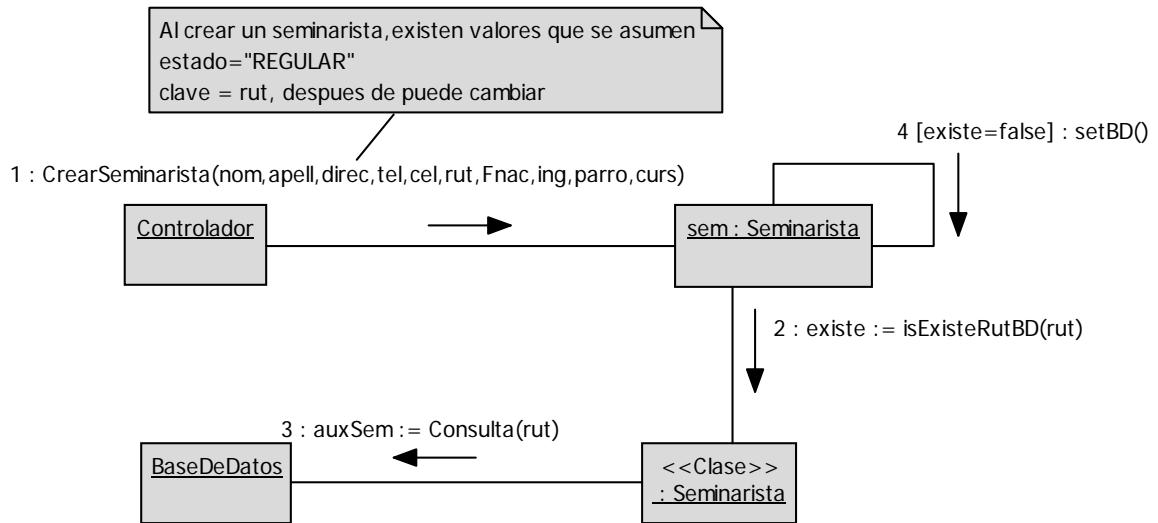


## PAQUETE ADMINISTRACIÓN DE USUARIOS.

**Caso de uso:** Actualizar Usuario.

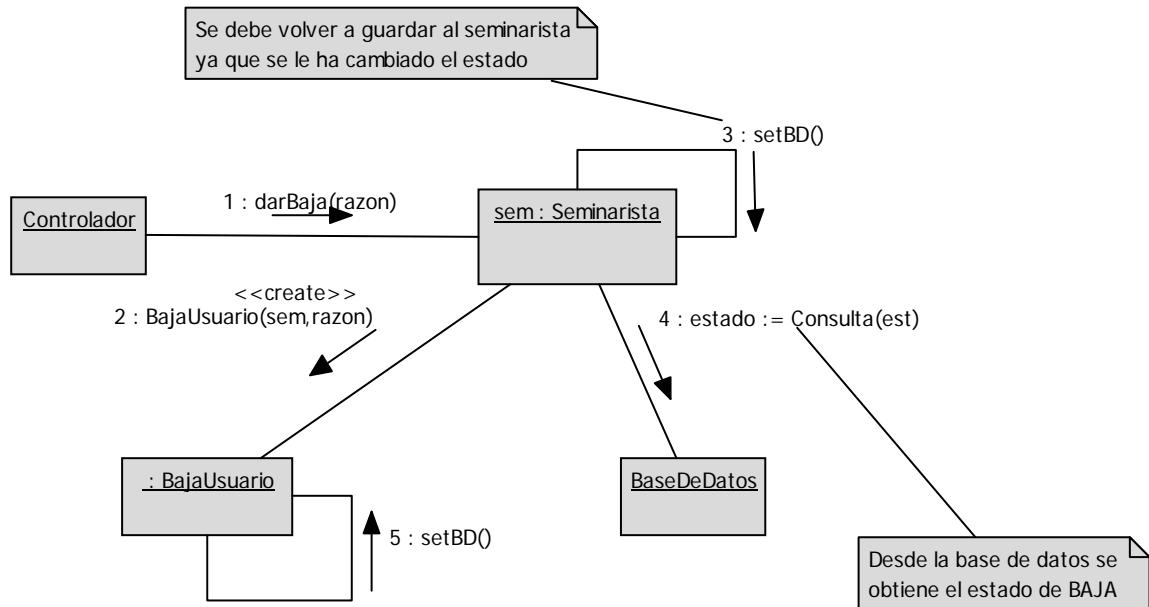
**Sección:** ingresar Usuario.

**Mensaje:** crearSeminarista()



**Caso de uso:** Dar baja usuario.

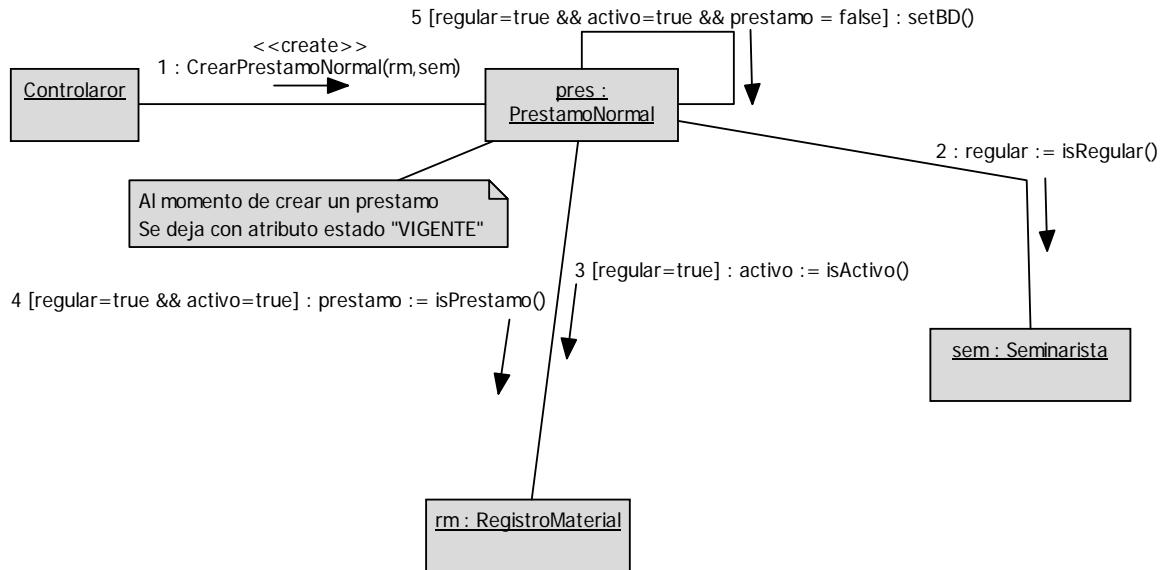
**Mensaje:** darBaja()



## PAQUETE PRÉSTAMOS Y DEVOLUCIONES.

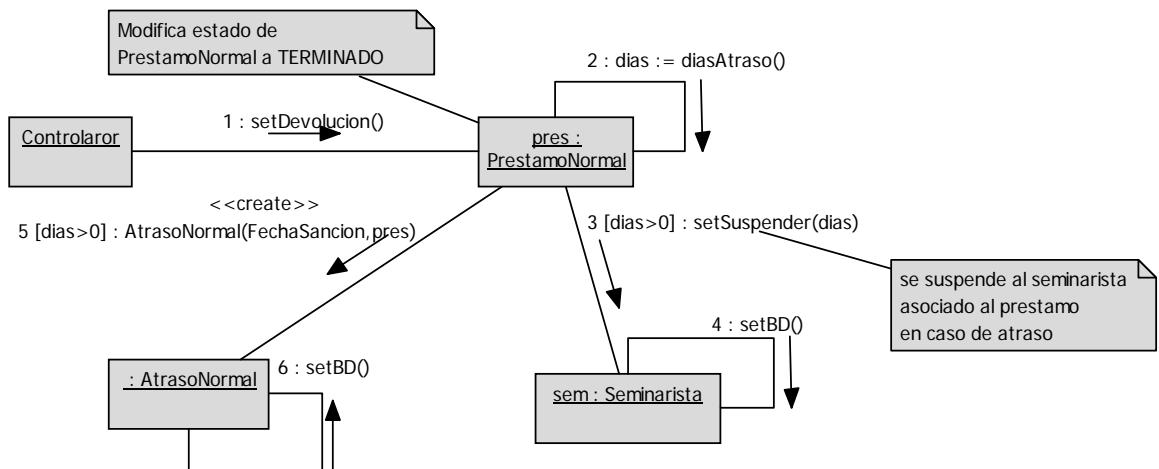
**Caso de uso:** Préstamos de material.

**Mensaje:** CrearPrestamoNormal().



**Caso de uso:** Devolución de material.

**Mensaje:** setDevolucion()



## 6.3 DIAGRAMAS DE CLASE

### PAQUETE CATÁLOGO LIBROS Y REVISTAS

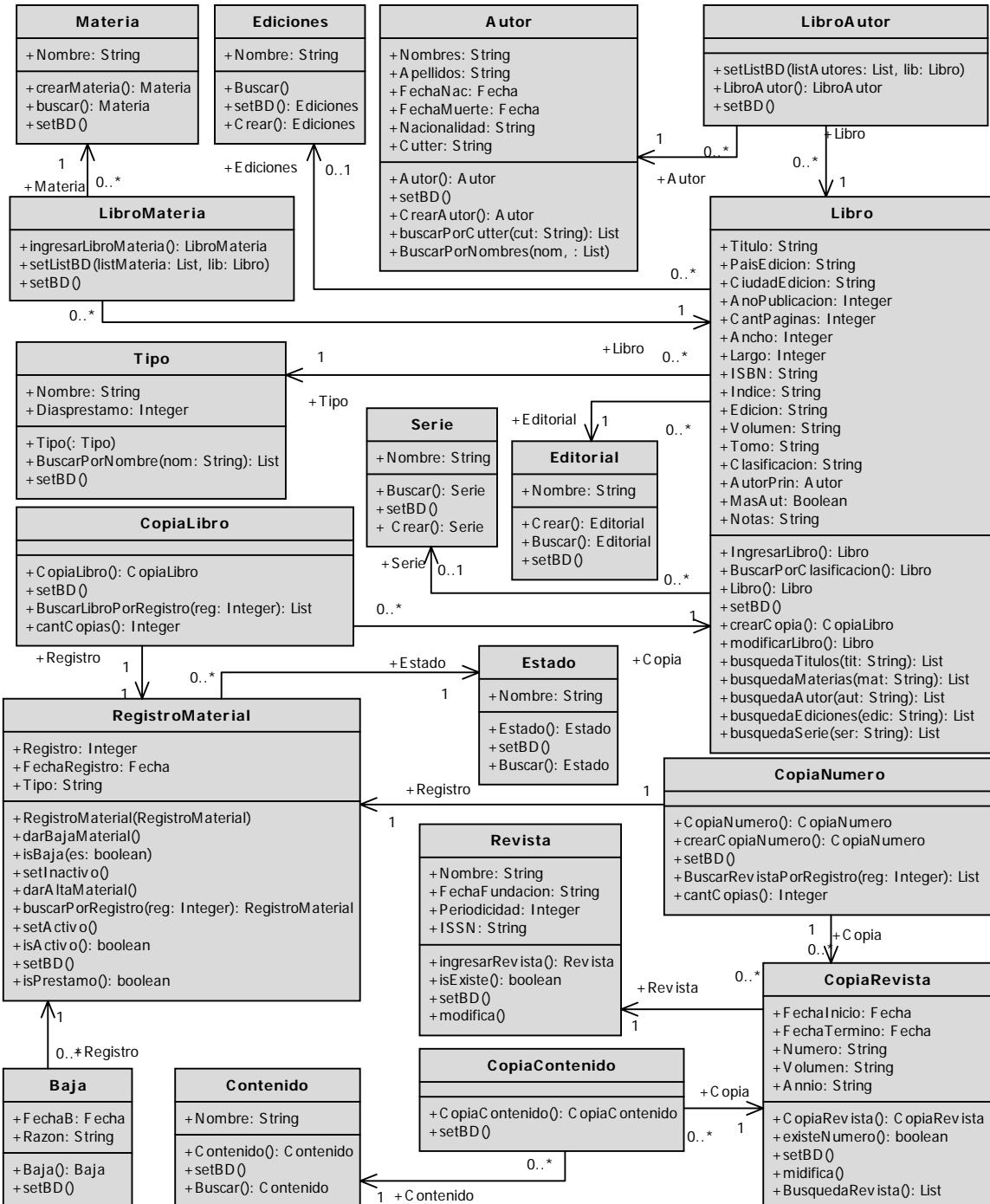


Figura 35: Diagrama de clases, paquete catálogo de libros y revistas.

## PAQUETE ADMINISTRACIÓN DE USUARIOS

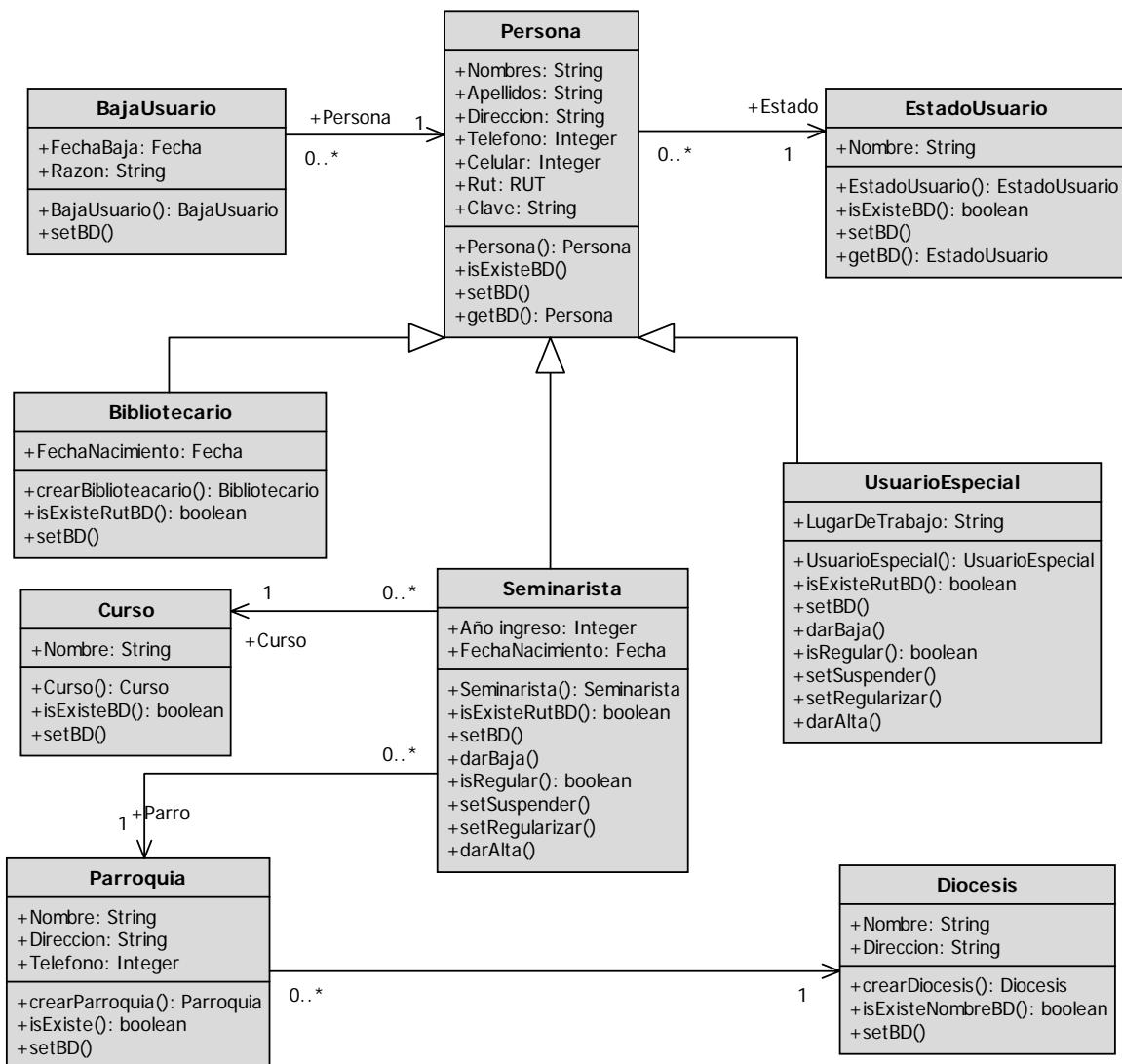


Figura 36: Diagrama de clases, paquete administración de usuarios.

## PAQUETE PRÉSTAMOS Y DEVOLUCIONES

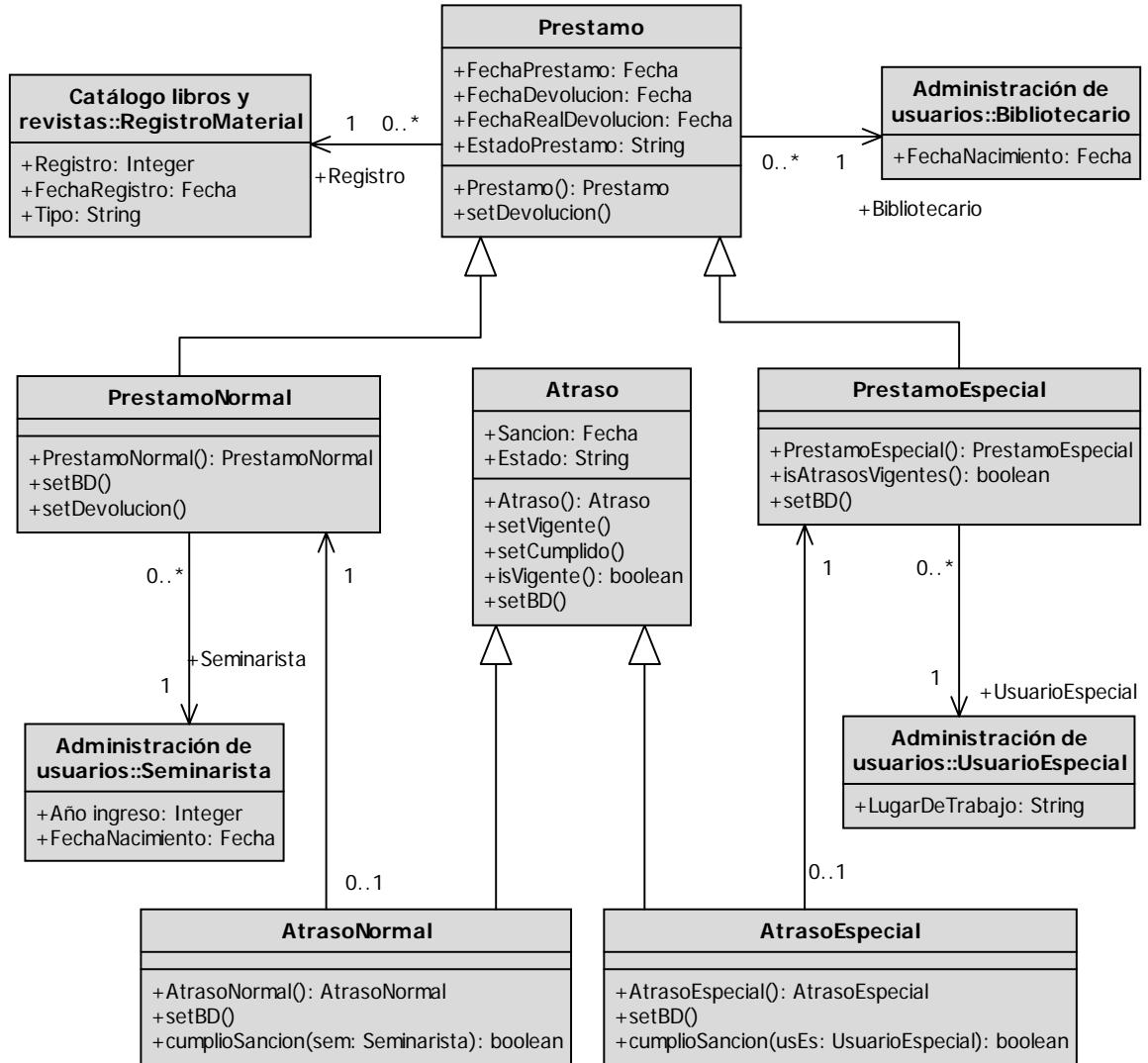


Figura 37: Diagrama de clases, paquete préstamos y devoluciones.

## PAQUETE PUBLICACIONES DIGITALES

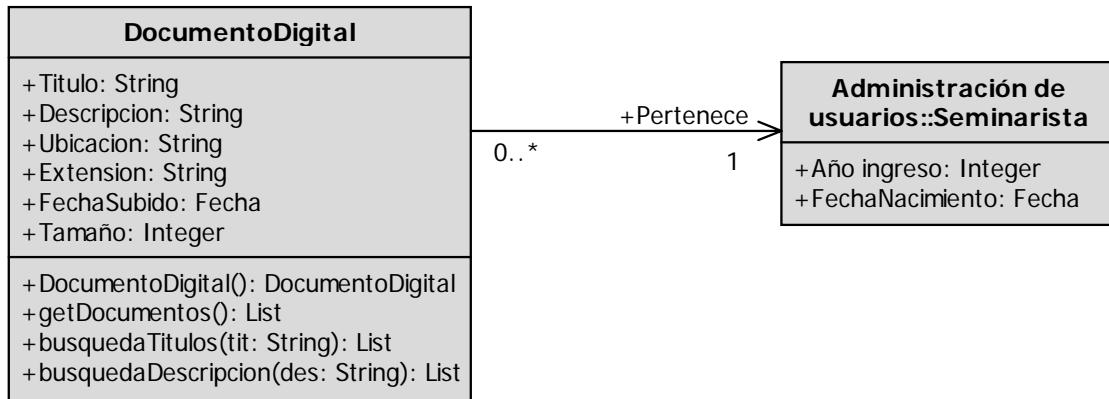


Figura 38: Diagrama de clases, paquete publicaciones digitales.

## PAQUETE SISTEMA DE MENSAJERÍA

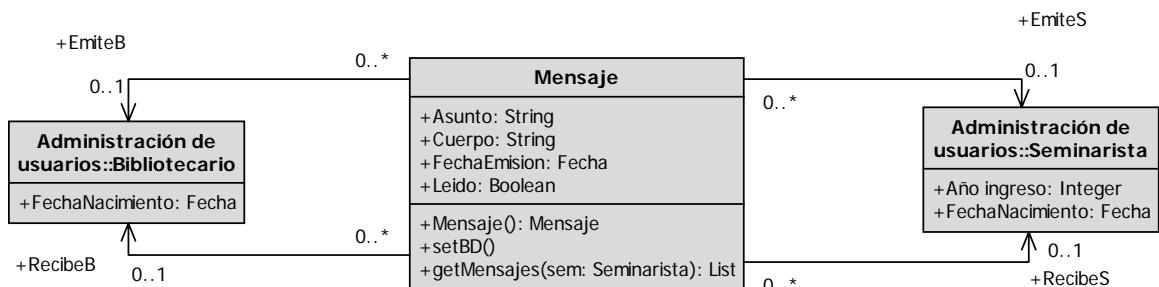


Figura 39: Diagrama de clases, paquete sistema de mensajería.

## PAQUETE RECOMENDACIÓN DE BÚSQUEDA

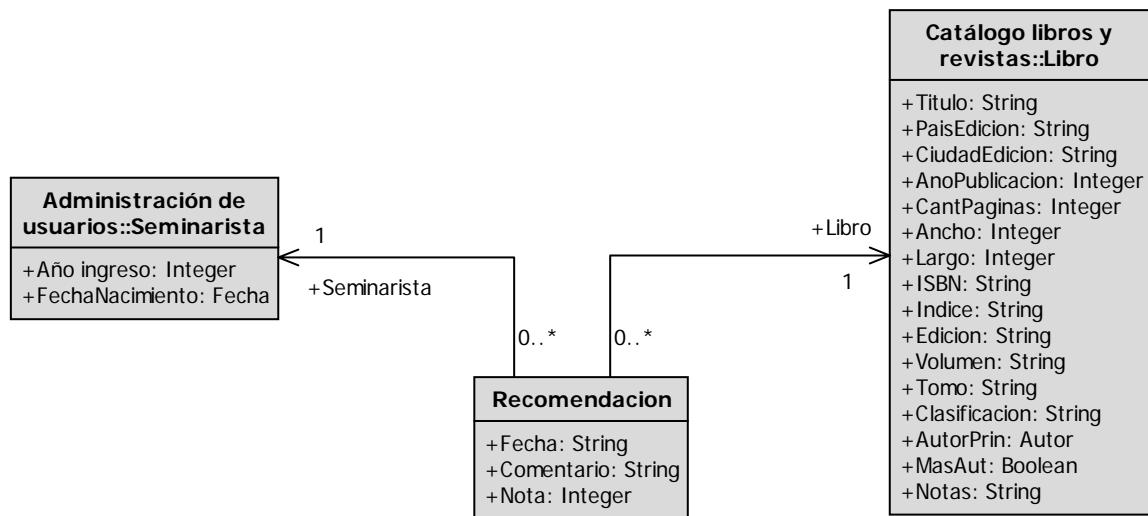


Figura 40: Diagrama de clases, paquete recomendación de búsqueda.

## **6.4 JERARQUÍA DE MENÚ**

### **APLICACIÓN DE ESCRITORIO**

- Libros
  - Crear libro
  - Modificar libro
  - Buscar libro
  - Dar baja material
  - Dar alta material
- Revistas
  - Crear revista
  - Modificar revista
  - Modificar número
  - Buscar revista
  - Dar baja material
  - Dar alta material
- Usuarios
  - Seminarista
    - Ingresar seminarista
    - Modificar seminarista
    - Como bibliotecario
    - Dar baja seminarista
    - Dar alta seminarista
  - Bibliotecario
    - Ingresar bibliotecario

- Modificar datos
  - Cambiar clave
  - Dar baja bibliotecario
  - Dar alta bibliotecario
  - Usuario especial
    - Ingresar usuario especial
    - Modificar usuario especial
    - Dar baja usuario especial
    - Dar alta usuario especial
  - Buscar Usuario
- Préstamos
    - Seminarista
      - Ingresar préstamo
      - Devolución
    - Usuarios especiales
      - Ingresar préstamo
      - Devolución

## **APLICACIÓN DE WEB**

- Búsqueda de libros

- Por título
- Por Autor
- Por Materia
- Por editorial
- Por ediciones
- Serie
- ISBN

- Revistas

- Por nombre
- Por ISSN
- Por Fecha
- Por Contenido
- Por Número

- Préstamos vigentes

- Préstamos históricos

- Configurar cuenta

- Mensajería

- Ver mensajes
- Enviar mensajes
  - Seminarista
  - Bibliotecario
  - Curso
  - Todos

- Documentos
  - Crear documento
  - Eliminar documento
  - Modificar documento
  - Buscar documento
    - Título
    - Descripción
- Recomendación
  - Recomendar

## 6.5 PANTALLAS DEL SISTEMA

En ésta sección sólo se mostrarán las principales pantallas de ambas aplicaciones (de escritorio y Web)

### PANTALLA PRINCIPAL APLICACIÓN DE ESCRITORIO

The screenshot shows the main window of the 'BIBLIOTECA SEMINARIO METROPOLITANO CONCEPCION' application. The menu bar includes 'Menu', 'Libros', 'Revistas', 'Usuarios', 'Prestamos', and 'Informes'. Below the menu are several icons: 'IngresarLibro', 'Ingresar Revista', 'Buscar Libro', 'Buscar Revista', and 'Buscar Seminarista'. The main area is divided into sections: 'Préstamos' (with fields for Rut, Nombre, Apellidos, Curso, Estado, and Parroquia) and 'Devolución' (with fields for Material, Tipo material, and Nombre). There is also a section for material information (Material, Tipo material, and Nombre). A large button at the bottom right says 'REALIZAR PRÉSTAMO' with a book icon. The background features illustrations of a priest reading a book and a bookshelf.

### PANTALLA INGRESAR SEMINARISTA

The screenshot shows the 'Ingresar seminarista' dialog box. It has a title bar 'Ingresar seminarista' with a small icon of a person. The main area is titled 'Ingreso de Seminarista' and contains the following fields:

Nombres	[Text Box]		
Apellidos	[Text Box]		
Rut	[Text Box]	Dirección	[Text Box]
Teléfono	[Text Box]	Celular	[Text Box]
Año Ingreso	[Text Box]	Fecha nacimiento	[Text Box] diciembre 20, 2007 [Calendar Icon]
<input type="checkbox"/> Sin fecha			
Curso	PROPE	Agregar	[Button]
Parroquia origen	SANTA CECILIA / CONCEPCION	Agregar	[Button]

At the bottom right is a 'Finalizar Ingreso' button.

## PANTALLA INGRESAR LIBRO (1/2)

Ingreso de Libros

**Ingrese los datos del libro**

Título	<input type="text"/>	<input type="button" value="Ingresar copia"/>	
ISBN	<input type="text"/>	Cantidad páginas	<input type="text"/>
Volumen	<input type="text"/>	Tomo	<input type="text"/>
Tipo	<input type="text" value="NO REFERENCIA"/> <input type="button" value="▼"/>	Registro anterior	<input type="text"/>
<i>Autor principal</i>			
Autor		<input type="text"/> <input type="button" value="▼"/>	<input type="button" value="Agregar"/>
<i>Clasificación</i>		<i>Edición</i>	
<input type="text"/>		<input type="text"/>	Año <input type="text"/>
<input type="text"/>		<input type="text"/> <input type="button" value="▼"/>	<input type="button" value="Agregar"/>
<input type="text"/>		<input type="text"/> <input type="button" value="▼"/>	<input type="button" value="Agregar"/>
<input type="text"/>		<input type="text"/> <input type="button" value="▼"/>	<input type="button" value="Agregar"/>
<i>Medidas</i>		<i>Notas</i>	
Largo	<input type="text"/>	<input type="text"/>	
Ancho	<input type="text"/>	<input type="text"/>	
<input type="button" value="Siguiente &gt;"/>			

## PANTALLA INGRESAR LIBRO (2/2)

Segunda etapa ingreso de libro

**Segunda etapa ingreso del libro**

Libro	ANTROPOLOGÍA TEOLOGÍA FUNDAMENTAL
Autor principal	MARTINEZ SIERRA, ALEJANDRO
Cutter	MA22

Seleccione Autor (es) secundario (s)

**Lista de autores**

ULLOA MERINO, CLAUDIO ENRIQUE (CU22)

**Autor(es) del libro**

>>  
<<

**Agregar nuevo autor**

Seleccione materia (s)

**Lista de materias**

PRUEBA  
RELIGION  
TEOLOGIA

**Materia (s) del libro**

ANTROPOLOGÍA  
TEOLOGÍA

>>  
<<

**Agregar nueva materia**

**Finalizar / ingresar Libro**

## PANTALLA INGRESAR REVISTA

 INGRESO DE REVISTA

**Ingrese los datos de la Revista**

Seleccione la revista

Datos número

Número  Año  Volumen

Periodicidad

Desde diciembre 20, 2007   Hasta diciembre 27, 2007

Contenido

CONTENIDO DE PRUEBA  
OTRO CONTENIDO

>>

## PANTALLA PRINCIPAL APLICACIÓN WEB



## PANTALLA BUSCAR LIBRO

Biblioteca Seminario Metropolitano Concepción

Criterio: TÍTULO  
Texto a buscar:  
Limpiar Enviar

TÍTULO	AUTOR (ES)	CLASIFICACIÓN	COPIAS	ESTADO	NOTA
Seminario Metropolitano Concepción O'higgins 1487 Chiguayante Concepción Bío-Bío Chile					

## PANTALLA BUSCAR REVISTA

Biblioteca Seminario Metropolitano Concepción

Nombre:  ISSN:   
Desde: 15/12/2007 ... Hasta: 15/12/2007 ...  
 Sin Fecha  Sin Fecha  
Contenido:  Número:   
Buscar revista

NOMBRE	NUMERO	COPIAS	ESTADO
Seminario Metropolitano Concepción O'higgins 1487 Chiguayante Concepción Bío-Bío Chile			

## PANTALLA GENERAL DE LA CUENTA

Biblioteca Seminario Metropolitano Concepción

Método: CREDITOS ALTIOS, TIPO: PENDIENTE  
Número de cuenta: 1021001  
Último: 10209  
Páginas visitadas: Páginas más visitadas: Configuraciones: Actividades: Comunicados: Recomendaciones:  
Historial de visitas:

Título	Tipo	Últ. Acceso	Últ. Página
METRO 3 105	TIPO3	14/12/2007	15/12/2007

Seminario Metropolitano Concepción  
O'higgins 1487 Chiguayante Concepción Bío-Bío Chile

## CAPITULO VII: DETALLES DE IMPLEMENTACIÓN

En este séptimo capítulo, se abarcarán los detalles más importantes de la implementación, vale decir, aquellos puntos que no son triviales y que sirven para dar una solución en el software del proyecto. Estos puntos son: el nivel de concurrencia en DB4O, la tecnología Ajax, algunos criterios al momento de trabajar con componentes Swing y la entrada y salida de datos mediante códigos de barra.

## 7.1 CONTROL DE CONCURRENCIA EN DB4O

Db4o es una base de datos orientada a objetos, por lo cual obtiene todas las características del paradigma, pero también limita al programador de algunas características que provee una base de datos relacional.

En primera instancia, en un objeto no se puede declarar un atributo como Primary Key, por lo que si existe algún atributo de un objeto que deba ser único, debe ser tratado mediante métodos, en otras palabras, se debe implementar la funcionalidad de impedir el ingreso de un atributo que ya está en la base de datos.

¿Cómo se hace con una base de datos en modo cliente-servidor?

La pregunta anterior hace mención a la posibilidad, que dos clientes ingresen al mismo tiempo el mismo valor para un atributo “identificador”, por ejemplo el RUT, o en el caso de los libros y revistas, el código de registro. Para aquello, DB4O provee semáforos, los que permiten bloquear las operaciones riesgosas, es decir, si dos usuario solicitan un semáforo con un nombre determinado, el segundo al momento de solicitarlo mediante el método para aquello, le retornará false, indicando que otro usuario pidió un semáforo antes y no lo ha devuelto.

A continuación se describe el código utilizado para aquello.

```
28 | try{
29 |     ObjectContainer cliente =
30 |         Db4o.openClient(Util.HOST,Util.PUERTO,Util.USER,Util.PASS);
31 |     while(!cliente.ext().setSemaphore("semáforo",2));
32 |     //Hacer verificación
33 |     //Guardar el dato si se desea
34 |     cliente.ext().releaseSemaphore("semáforo");
35 |     cliente.close();
36 | }catch(IOException msg){
37 |     System.out.println(msg.getMessage());
38 | }
```

Figura 41: Código con semáforos de DB4o.

En la Figura 41 se puede observar que si dos usuarios piden el semáforo de nombre “semáforo” (cliente.ext().setSemaphore(“semáforo”,2))), al segundo usuario le retornará falso y como en el ciclo while existe una negación, esperará 2 milisegundos antes de preguntar nuevamente por el semáforo, es decir, el usuario podrá pasar el ciclo while una vez se devuelva el semáforo (cliente.ext().releaseSemaphore(“semáforo”)).

De ésta forma, se controlan los ingresos de atributos que son identificadores dentro del objeto.

## 7.2 IMPLEMENTACIÓN CON AJAX

El sistema Web dedicado, principalmente, a la búsqueda de material, posee varias características importantes, ya que posee las siguientes opciones:

- Búsqueda de libros y revistas.
- Consultar por préstamos vigentes e históricos.
- Sistema de mensajería.
- Sistema de almacenamiento para archivos digitales.
- Recomendación de búsqueda.
- Configuración de cuenta.

Todas las opciones mencionadas anteriormente han sido implementadas mediante la tecnología Ajax, esto permite que la interfaz principal se mantenga intacta mientras se realiza la recarga solo del contenido que solicita. A continuación, se dará a conocer un ejemplo de cómo se realiza la recarga (funciones en JavaScript).

```
30 function cargarPagina(div,pagina){  
31     var ajax=objetoAjax();  
32     var divMensaje = document.getElementById(div);  
33     ajax.open("GET", pagina);  
34     ajax.onreadystatechange=function()  
35     {  
36         if (ajax.readyState==4)  
37         {  
38             divMensaje.innerHTML = ajax.responseText;  
39         }  
40     }  
41     ajax.send(null)  
42 }
```

Figura 42: Función que ocupa el sistema para recargar la Web.

En la Figura 42 se muestra que la función (de javascript) recargarPagina(div,pagina), se puede apreciar que se le entregan 2 parámetros, el primero (div) es el nombre o identificador del contenedor en la que realizará la recarga, y el segundo parámetro (pagina) es la dirección de la pagina que estamos solicitando para que sea recargada, ésta puede venir con datos incluidos, por ejemplo, pagina.jsp?idObjeto=123.

A continuación se especifica la función objetoAjax() que se nombra en la imagen anterior en la línea 31.

```
4 function objetoAjax(){
5     var xmlhttp=false;
6     try {
7         xmlhttp = new ActiveXObject("Msxml2.XMLHTTP");
8     } catch (e) {
9         try {
10             xmlhttp = new ActiveXObject("Microsoft.XMLHTTP");
11         } catch (E) {
12             xmlhttp = false;
13         }
14     }
15
16     if (!xmlhttp && typeof XMLHttpRequest!=='undefined') {
17         xmlhttp = new XMLHttpRequest();
18     }
19     return xmlhttp;
20 }
```

Figura 43: función objetoAjax().

En la imagen anterior se aprecia la función encargada de crear el objeto Ajax, que nos permitirá mantener la comunicación asíncrona con el servidor.

## 7.3 DETALLES DE DISEÑO CON SWING EN JAVA

La aplicación de escritorio que utilizará el bibliotecario está realizada en java con componentes Swing (Biblioteca gráfica), los componentes que más se utilizaron son:

- JLabel** : Utilizado para texto en la aplicación.
- JButton** : Utilizado para los botones.
- JList** : Utilizado para las listas.
- JTextArea** : Utilizado para las cajas de texto de varias líneas.
- JTextField** : Utilizado para cajas de texto.
- JTable** : Utilizado para la creación de tablas.
- JOptionPane** : Utilizado para mensajes al usuario.
- JDialog** : Utilizado para contener otros componentes swing, (ventana flotante).
- JFrame** : Utilizado para la contener otros componentes swing (ventana).

### JTABLE

Es relevante hacer mención que el componente **JTable** no fue utilizado por defecto, es decir no tiene las propiedades definidas que la biblioteca proporciona, ya que la tabla por defecto puede ser editada (modificar el texto que aparece en las celdas), y en el sistema aquello no tiene utilidad, por lo cual fue reescrito un método. El método que fue reescrito fue:

```
23     Borrar metodos isCellEditable de la clase javax.swing.table.DefaultTableModel  
24  
25     ↑  
26     public boolean isCellEditable (int row, int column)  
27     {  
28         return false;  
29     }
```

Figura 44: Método reescrito en el modelo de JTable.

En la Figura 44 se aprecia el método que fue reescrito en una clase que hereda las propiedades de DefaultTableModel (Clase que se asocia al modelo de tabla). Este método se encarga de retornar si las celdas son editables o no. Para que el componente JTable se encargue de aquello, esa es la razón de los parámetros que se le entregan, row y column, pero en este caso puntual ninguna fila y columna se podrá editar, es decir, no importa la celda que se le entregue por parámetro, el método siempre retornará false.

Al ser reescrita ésta propiedad, la herramienta CASE también nos hace saber que el método está sobrescribiendo al método del padre. Luego, ésta nueva clase será asociada como modelo al componente JTable.

## **JDIALOG y JFRAME**

Al comenzar la programación de la interfaz gráfica, lo primero que se hace es la creación de ventanas, para este caso JFrame, la herramienta CASE proporciona una serie de utilidades para que la creación de ventanas se pueda realizar con un entorno gráfico.

¿Hay alguna restricción al momento de crear ventanas?

En el transcurso del proyecto, esta pregunta fue tomada el peso que en un comienzo se debió tener, el sistema estaba repleto de ventanas tipo JFrame, además cada vez que se llamaba a una ventana se construía nuevamente, es decir, si una ventana se abre 20 veces, 20 veces el recolector de basura de java tendrá la misión de liberar memoria (ya que se pierde referencia al objeto). Entonces, antes de implementar cualquier parte del software es necesario recurrir a la literatura y a la experiencia que han tenido otros programadores que ya han pasado por aquello. Luego de esto, se debió reconstruir prácticamente todo, pero ésta vez con nuevos criterios que han hecho que la aplicación pueda ser utilizada con mayor calidad.

### **Las conclusiones que se tomaron son las siguientes:**

- En toda la aplicación debe existir solo un contenedor JFrame como ventana principal. Como ventanas secundarias se utiliza el componente JDDialog.

- Las ventanas secundarias al momento de cerrarlas solo quedarán “ocultas” <setVisible(false)>, es decir, no destruirlas, y al momento de llamarlas nuevamente hacerlas visible <setVisible(true)>, esto hace que las ventanas aparezcan mucho más rápida que estarlas construyendo nuevamente. También en cada JDialog del sistema se implementó un método que es fundamental para esto, es el método reset(). Este método, deja la ventana nuevamente por defecto, es decir elimina todos los contenidos en las cajas de texto, deja la ventana en la posición inicial, etc.
- Al iniciar el programa, cargar en memoria todas las ventanas que el sistema necesita, lo que proporciona mayor rapidez al momento de su ejecución (basta con crear las ventanas al inicio del programa).

## **7.4 CÓDIGOS DE BARRA**

Una de las funciones principales del proyecto, es que los libros son identificados mediante un número de registro que se representa mediante códigos de barra, el tratamiento para la generación de códigos de barra y para su lectura se dan a continuación.

### **IMPRESORA DE CÓDIGOS DE BARRA.**

Para la generación de códigos de barra, existen variadas posibilidades, pero no todas ofrecen soluciones eficientes. La solución más óptima, es la adquisición de una impresora de etiquetas. Una impresora de este tipo no posee el mismo funcionamiento de una impresora tradicional, una de las propiedades más importantes es la impresión térmica sobre etiquetas autoadhesivas, la que ofrece una buena calidad de impresión.

Una vez obtenida la impresora, existe la interrogante, ¿cómo imprimir los códigos de barra?, nuevamente existen varias opciones, entre ellas están:

- obtener un software para el diseño e impresión de códigos de barra.
- utilizar bibliotecas para JAVA como “barbecue-1.5-beta1”, las que producen imágenes de códigos de barra.
- Impresión directa mediante el lenguaje de programación que soporte la impresora, en el caso puntual de Zebra LP2824, el lenguaje es EPL2 (Eltron Programming Language).

La última opción es la más eficiente y compleja, pero para este proyecto es indispensable que el mismo software genere las etiquetas de clasificación y las etiquetas de códigos de barra, por lo cual, adquirir otra aplicación es descartada en primer instancia, luego se ha decidido la impresión a través del lenguaje que provee la impresora.

Para imprimir una etiqueta, se crea un archivo temporal, por ejemplo archivo.txt en él se encuentran los comandos que son enviados al puerto de la impresora (LPT1) con los comandos de sistema, por ejemplo, copy código.txt LPT1

**Ejemplo de un código de barras sencillo:**

N

A50,5,0,1,2,2,N,"SEMINARIO METROPOLITANO CONCEPCIÓN"

A50,30,0,4,1,2,N,"NOMBRE LIBRO"

B250,100,0,E30,3,20,160,B,"8424070022915"

P1

**Comando N:**

*Descripción:* Indica el comienzo de una nueva etiqueta.

**Comando A:**

*Descripción:* Indica texto, formato = Ap1,p2,p3,p4,p5,p6,p7,“TEXTO”

Donde: p1= posición del texto horizontal.

p2= posición vertical.

p3= rotación. 0=sin rotación; 1=90 grados; 2= 180 grados; 3= 270 grados.

p4= fuente. No se muestra la tabla debido a lo extenso de ésta.

p5= amplía el texto horizontalmente (lo multiplica por el valor) de 1 a 9.

p6= amplía el texto verticalmente (lo multiplica por el valor) de 1 a 9.

p7= indica si es normal o al revés. Valores N y R.

**Comando B:**

*Descripción:* similar al comando “A” pero en TEXTO se indica el código de barras.

**Comando P:**

*Descripción:* Indica la cantidad de copias que se deben imprimir.

## **LECTOR DE CÓDIGOS DE BARRA.**

Un lector de códigos de barra decodifica la información a través de la digitalización proveniente de una fuente de luz reflejada en el código y luego se envía la información a una computadora como si la información hubiese sido ingresada por teclado. Lo bueno de los lectores de códigos de barra es que finalizan el ingreso de la lectura con [Enter], lo que permite manejar los eventos.

## CAPITULO VIII: PLAN DE PRUEBA, PUESTA EN MARCHA Y CAPACITACIÓN

En este octavo capítulo, se desarrollará un plan de pruebas para detectar posibles errores que el software pueda presentar en su funcionamiento. También se nombra cual es la puesta en marcha que tendrá el sistema y una pequeña descripción de cómo capacitar a los usuarios del software.

## 8.1 PLAN DE PRUEBAS

Al trabajar con el modelo incremental, existe una gran ventaja al momento de las pruebas, ya que éstas se realizan en cada incremento. Las pruebas realizadas en cada incremento han sido con datos reales, esto quiere decir, que los primeros incrementos han sido probados fuertemente. Además, en cada incremento realizado para este proyecto, se realizaron constantemente pruebas de “caja negra”, estas pruebas fueron realizadas por el programador sin mayor formalidad.



Figura 45: Pruebas de los incrementos a las de integración.

Una vez que se han implementados todos los incrementos, se realizan las pruebas de integración, las que deben asegurar el funcionamiento de los casos de usos que establecieron en el capítulo de análisis.

En este plan, solo se contemplan pruebas de validación, es decir, se verificará que el software cumple con los requisitos del análisis.

También es de importancia mencionar, que el software deberá operar para varios clientes al mismo tiempo y podrían realizar las mismas o distintas operaciones, por lo que el control de la concurrencia también está presente en el control de errores.

A continuación se listarán los casos de pruebas, para los posibles errores que pudiesen perjudicar severamente el funcionamiento del sistema, y solo se abarcan los casos de usos *Primarios*. Además en las entradas solo se reflejan los atributos que pueden generar daño y en el mismo orden en que se muestran los casos de prueba deben ser comprobados.

## CASO DE PRUEBA 1:

**Paquete:** Catálogo libros y revistas.

**Caso de uso:** Actualizar Libro, sección: Ingresar Libro.

**Atributos de entrada:** (título, autor, clasificación, materia):

Entradas: “LIBRO NUEVO”, “NUEVO AUTOR”, “123.4/C22AUTOR/2008”, “INFORMÁTICA”.

**Resultado esperado:**

El libro se ingresa sin problemas.

Entradas: “NUEVO LIBRO”, “AUTOR”, “123.4/C22AUTOR/2007”, “”.

**Resultado esperado:**

No debe permitir el ingreso del libro, ya que no está asociado a una materia.

Entradas: “”, “NUEVO AUTOR”, “123.4/C22AUTOR/2008”, “FILOSOFÍA”.

**Resultado esperado:**

No debe permitir el ingreso del libro, ya que clasificación ya existe en el sistema y además no posee título.

## CASO DE PRUEBA 2:

**Paquete:** Catálogo libros y revistas.

**Caso de uso:** Actualizar Libro, sección: Modificar Libro.

**Atributos de entrada:** ( Registro a modificar, Titulo, autor, clasificación, materia):

Entradas: “0”, “NUEVO LIBRO”, “AUTOR”, “123.6/C22AUTOR/2008”, “INFORMÁTICA”.

**Resultado esperado:**

Permitir el ingreso del libro.

Entradas: “1”, “”, “SIN AUTOR”, “123.4/C22AUTOR/2008”, “INFORMÁTICA”. “”.

**Resultado esperado:**

No permitir el ingreso, porque no contiene título.

No permitir el ingreso por que la clasificación cambiada ya existe.

No permitir ingreso por que el libro no se ha asociado a ninguna materia.

## CASO DE PRUEBA 3:

**Paquete:** Catálogo libros y revistas.

**Caso de uso:** Dar baja material.

**Atributos de entrada:** (registro):

Entradas: “0”.

**Resultado esperado:**

Permitir la baja del material.

Entradas: “1”.

**Resultado esperado:**

Permitir la baja del material.

Entradas: “0”.

**Resultado esperado:**

No permitir la baja del material, porque este ya está de baja.

## CASO DE PRUEBA 4:

**Paquete:** Catálogo libros y revistas.

**Caso de uso:** Actualizar Revistas, sección: Ingresar Revista.

**Atributos de entrada:** (nombre, número, contenido).

Entradas: “NUESTRA IGLESIA”, “1”, “ORDENACIÓN SACERDOTAL”.

**Resultado esperado:**

Permitir el ingreso de la revista.

Entradas: “NUESTRA IGLESIA”, “1”, “NAVIDAD”.

**Resultado esperado:**

No permitir el ingreso de la revista, ya que el número ya existe.

Entradas: “JOVEN CATÓLICO”, “”, “ORDENACIÓN SACERDOTAL”.

**Resultado esperado:**

No permitir el ingreso, ya que esta no posee número.

## CASO DE PRUEBA 5:

**Paquete:** Administración de usuarios.

**Caso de uso:** Actualizar Usuario, sección: Ingresar usuario.

**Atributos de entrada:** (nombre, Rut, curso):

Entradas: “JORGE”, “22.222.222-2”, “PRIMERO DE FILOSOFÍA”.

**Resultado esperado:**

Permitir el ingreso del seminarista.

Entradas: “DANIEL”, “33.333.333-3”, “SEGUNDO DE TEOLOGÍA”.

**Resultado esperado:**

Permitir el ingreso del seminarista.

Entradas: “CAMILO”, “22.222.222-2”, “SEGUNDO DE FILOSOFÍA”.

**Resultado esperado:**

No permitir el ingreso ya que el Rut ya existe en el sistema.

Entradas: “”, “44.4444.4444-4”, “SEGUNDO DE FILOSOFÍA”.

**Resultado esperado:**

No permitir el ingreso ya que no posee nombre.

## CASO DE PRUEBA 6:

**Paquete:** Administración de usuarios.

**Caso de uso:** Actualizar Usuario, sección: Modificar usuario.

**Atributos de entrada:** (Rut a modificar, nombre, curso, ):

Entradas: “22.222.222-2”, “”, “PRIMERO DE FILOSOFÍA”.

**Resultado esperado:**

No permitir la modificación, ya que este no posee nombre.

## CASO DE PRUEBA 7:

**Paquete:** Préstamos y devolución.

**Caso de uso:** Préstamos de material.

**Atributos de entrada:** (registro, Rut seminarista):

Entradas: “0”, “22.222.222-2”

**Resultado esperado:**

Permitir el préstamo del material.

Entradas: “1”, “22.222.222-2”

**Resultado esperado:**

Permitir el préstamo del material.

Entradas: “0”, “33.333.333-3”

**Resultado esperado:**

No permitir el préstamo del material ya que éste está prestado.

## CASO DE PRUEBA 8:

**Paquete:** Préstamos y devolución.

**Caso de uso:** Devolución del material.

**Atributos de entrada:** (registro):

Entradas: “0”.

**Resultado esperado:**

Devolución exitosa.

Entradas: “1”.

**Resultado esperado:**

Devolución exitosa.

Entradas: “0”.

**Resultado esperado:**

No se puede realizar devolución ya que el material no se encuentra en préstamo.

## **8.2 PUESTA EN MARCHA**

La puesta en marcha es una de las actividades importantes en un proyecto de software, ya que es el momento en que todo lo realizado, se lleva a la práctica esperando resultados positivos, pero en este, como en cualquier proyecto se pueden presentar anomalías, por lo que es necesario prever cualquier situación.

En el caso puntual del proyecto en cuestión, existe un sistema manual para la biblioteca que se ha utilizado durante bastante tiempo, lo que ha significado un arduo trabajo de ingreso y clasificación de material.

Sería inapropiado, por no decirlo “injusto” que la clasificación e ingreso existente del material se perdiera para dar paso al nuevo sistema computacional, sin embargo un software no está libre de errores y es necesario mantener el sistema antiguo, por lo menos hasta que el sistema logre estabilidad. La estabilidad para éste software está contemplada entre tres a seis meses.

En primera instancia, el bibliotecario o quien sea designado para aquello deberá poblar la base de datos con al menos 200 libros y la totalidad de los usuarios.

Luego corresponde la “marcha blanca”, tiempo en la cual estarán trabajando los dos sistemas al mismo tiempo. Por consiguiente, si ocurre algún error en el software, el sistema anterior no tendrá ningún problema y podrá seguir funcionando hasta que el software sea reparado.

El tiempo que se estima conveniente para la marcha blanca es de cuatro meses.

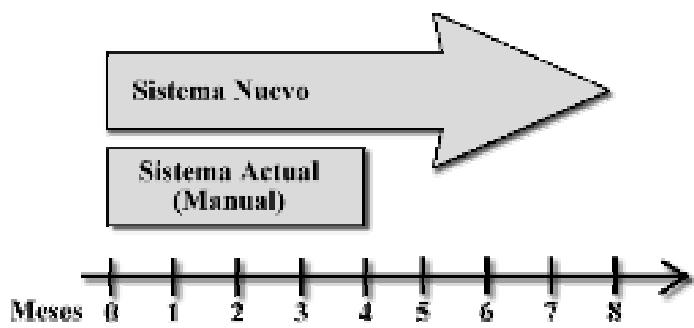


Figura 46: Tiempo marcha blanca

Desde la puesta en marcha hacia adelante se contempla un respaldo de la base de datos mediante discos DVD, al menos una vez cada tres meses, la base de datos es un archivo que podrá ser copiado sin ningún problema.

## **8.3 PLAN DE CAPACITACIÓN**

El plan de capacitación contemplado para el proyecto tiene dos fines, habilitar al bibliotecario para el uso del software de escritorio y habilitar a los seminaristas para hacer uso del sistema Web.

En efecto, el plan de capacitación consta de dos etapas aludiendo a los fines nombrados anteriormente.

### **Capacitación Bibliotecario.**

- Charla, en la cual se explica la teoría de todo lo que el sistema permite hacer.
- Presentación del software donde se expliquen todas las funcionalidades.
- Presentación en detalle de funcionalidades principales.
  - Ingreso de Libros.
  - Ingreso de revistas.
  - Ingreso de Usuario.
  - Préstamos.
  - Devoluciones.
- Actividad práctica en la cual el bibliotecario interactúe directamente con el software.
- Actividad práctica en la cual el Bibliotecario realice al menos una prueba por cada funcionalidad.
- Actividad práctica en la que se explica cómo levantar el sistema de escritorio.
- Actividad práctica en la que se enseña a levantar el sistema Web.
- Actividad práctica en la que se explica cómo instalar el software en otro computador y dejarlo activo para su uso.
- Actividad práctica de cómo utilizar el sistema Web.

## **Capacitación Seminaristas.**

- Charla, en la cual se explica la teoría de todo lo que el sistema permite hacer.
- Actividad práctica en la cual los usuarios interactúan con el sistema.
- Actividad práctica en la cual se explica a grandes rasgos las funcionalidades del software de escritorio.

## CONCLUSIONES

Al dar termino a este proyecto, se puede afirmar con satisfacción que lo escrito en la literatura acerca del paradigma orientado a objetos, se hace vida en quien realizó el análisis, diseño e implementación descritos en este libro, lo más acertado es lo que nos dice Martin en el primer capítulo, pero esto no sería tan tajante, a menos que el paradigma se utilice en todo el ciclo del proyecto, como nos menciona Meyer.

*Las Técnicas Orientadas a Objetos modifican el punto de vista de los analistas de sistemas de información acerca del mundo, en vez de pensar en los procesos y su descomposición, piensan en Objetos y su comportamiento* (Martin, y otros, 1994).

*El enfoque Orientado a objetos es ambicioso: abarca todo el ciclo de vida de desarrollo software. Cuando se examinan soluciones orientadas a objetos se debe comprobar que el método y el lenguaje, así como las herramientas de apoyo, sean aplicables al análisis y al diseño al igual que la implementación y el mantenimiento* (Meyer, 1998)

Al desarrollar un proyecto con el paradigma orientado a objetos en todas sus etapas, ya desde las primeras conversaciones con los usuarios, se comienzan a percibir ¡objetos!, que serán las piezas fundamentales del software, ya que estos interactúan de manera sorprendente con otros objetos, como si estos comenzaran a tener vida, con comportamientos y estados propios. Además al trabajar con una base de datos del mismo paradigma, los objetos persisten sin tener que realizar transformaciones a bases de datos relacionales, manteniendo las características del paradigma.

Algo interesante de rescatar es la importancia que tiene el realizar un buen análisis, ya que las palabras muchas veces se malentienden o tal vez se dijo aquello que realmente no era, entonces un análisis exhaustivo al comienzo, permitirá el desarrollo del proyecto con mas confiabilidad y seguridad, logrando un buen término.

Otro punto interesante, es la elección de una herramientas CASE apropiada para utilizarla en un proyecto determinado, ya que estas al ser bien utilizadas proporcionan en gran medida el éxito del proyecto.

También, es importante resaltar que se cumplieron con todos los objetivos planteados al inicio de este proyecto, lo que se ve reflejado a lo largo de este libro y en el aprendizaje logrado.

Para finalizar, no se puede dejar de mencionar el descubrimiento por parte de quien realiza este proyecto la potencia de un Lenguaje como JAVA ya que permite una gran cantidad de cosas, la más interesante es la capacidad para reutilizar la lógica de negocio en ambas aplicaciones que se desarrollaron, tanto en la aplicación de escritorio, como en la aplicación Web, en ésta última, gracias a la tecnología JSP se logró la utilización del código implementado en la primera aplicación, logrando una coherencia y unión de estos dos software para lograr un gran Sistema de Biblioteca como lo plantea este Proyecto.

## BIBLIOGRAFÍA

**2007, NetBeans.** NetBeans. [En línea] [Citado el: 18 de Agosto de 2007.]  
[http://www.netbeans.org/index\\_es.html](http://www.netbeans.org/index_es.html).

**Benito, Miguel. 1999.** *Manual de aprendizaje de la Clasificación Decimal Universal y breve introducción a la Clasificación Decimal de Dewey*. s.l. : Taranco, 1999.

**Bertino, Elisa y Martino, Lorenzo. 1995.** *Sistemas de bases de datos orientadas a objetos*. Madrid : Addison wesley, 1995.

**creangel UML. 2007.** creangel. [En línea] 2007. [Citado el: 12 de septiembre de 2007.]  
<http://www.creangel.com/uml/diagramas.php>.

**Deitel, Harvey. 2004.** *Como programar en JAVA*. Mexico : Pearson, 2004.

**Larman, Craig. 2003.** *UML y Patrones*. Madrid : Pearson, 2003.

**Lavintman, Alan y Viscuso, German. 2007.** *db4o: una alternativa a la persistencia* . s.l. : [www.db4o.com](http://www.db4o.com), 2007.

**Martin, James y Odell, James. 1994.** *Analisis y Diseño Orientado a Objetos*. Mexico : Prentice Hall, 1994.

**Meyer, Bertrand. 1998.** *Construccion de Software Orientada a Objetos (Segunda edicion)*. España : Prentice Hall, 1998.

**Miksa, Francis L. 1998.** *The DDC, The universe of knowledge and the post-modern library*. s.l. : Forest press, 1998.

**Natxo, Mendez. 2007.** DesarrolloWeb. [En línea] 2007. [Citado el: 1 de Agosto de 2007.]  
<http://www.desarrolloweb.com/articulos/832.php>.

**Paredes, Juan de Dios Bátiz.** *Desarrollo Orientado a Objetos con UML*.

**Pérez, Javier Eguíluz. 2007.** *Introducción a AJAX*. 2007.

**Pressman. 2002.** *Ingeniería del Software Un enfoque Práctico*. s.l. : McGraw Hill., 2002.

**Sabater, Juan Carlos. 2007.** ciberhabitat. [En línea] 2007. [Citado el: 11 de Octubre de 2007.]  
<http://ciberhabitat.gob.mx/comercio/cbarras/index.html>.

**Viscuso, German.** 2007. *Bases de Objetos*. s.l. : www.db4o.com, 2007.

**Wikipedia codigos Barra.** 2007. [En línea] 2007. [Citado el: 12 de Octubre de 2007.]  
[http://es.wikipedia.org/wiki/Código\\_de\\_barras](http://es.wikipedia.org/wiki/Código_de_barras).

**Wikipedia JAVA,** 2007. Wikipedia. [En línea] [Citado el: 27 de Agosto de 2007.]  
[http://es.wikipedia.org/wiki/Java\\_\(Sun\)](http://es.wikipedia.org/wiki/Java_(Sun)).

**Wikipedia JSP,** 2007. Wikipedia. [En línea] [Citado el: 28 de Agosto de 2007.]  
[http://es.wikipedia.org/wiki/Java\\_Server\\_Pages](http://es.wikipedia.org/wiki/Java_Server_Pages).

**Wikipedia Lenguaje Java,** 2007. Wikipedia. [En línea] [Citado el: 20 de Agosto de 2007.]  
[http://es.wikipedia.org/wiki/Lenguaje\\_de\\_programación\\_Java](http://es.wikipedia.org/wiki/Lenguaje_de_programación_Java).

**Wikipedia POO.** 2007. Wikipedia. [En línea] 2007. [Citado el: 12 de Septiembre de 2007.]  
[http://es.wikipedia.org/wiki/Programación\\_orientada\\_a\\_objetos](http://es.wikipedia.org/wiki/Programación_orientada_a_objetos).

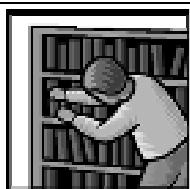
**ANEXO 1**

**MANUAL DE USUARIO DEL SISTEMA BIBLIOTECARIO**

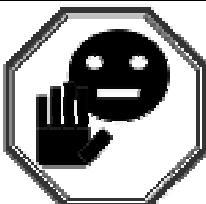
# INTRODUCCIÓN

En éste tutorial del sistema Bibliotecario de la Biblioteca del Seminario Metropolitano Concepción, se plasma de forma gráfica e interesante la forma en que se debe utilizar el sistema bibliotecario. Al leer el tutorial se aprenderá desde el ingreso al sistema, hasta las operaciones que requieren conocimiento en bibliotecas.

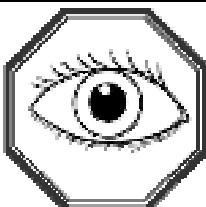
Este tutorial está diseñado de forma que el entendimiento sea más fácil para el usuario. A continuación se describe los símbolos que aparecerán a lo largo del tutorial.



**Solo Bibliotecario:** Cuando aparezca éste símbolo, significa que para realizar la operación se necesitan conocimientos de biblioteca. Como por ejemplo, el cómo clasificar un Libro, el ingreso de una revista, etc.



**Cuidado:** Este símbolo significa que para realizar la operación debe considerar algunas cosas, por lo general si no se considera la operación no se podrá realizar.



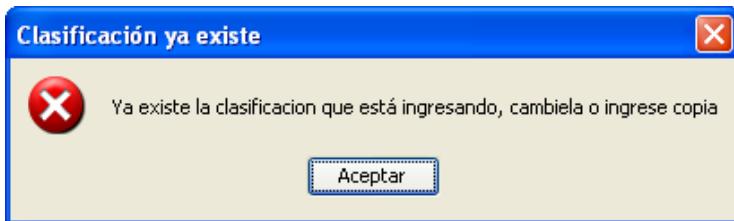
**Observación:** Este símbolo aparece cuando se plasman observaciones generales. La mayor parte de las veces sirve para entender el significado de algunos atributos que el sistema pide.

El tutorial, en general, muestra todas las pantallas que el sistema le arrojará y describe las acciones importantes a través de números que identifican en la pantalla los que se describirá, por lo general son botones o cajas de texto para anotar los atributos que el sistema requiere.

Cuando exista un error en el sistema, este lo hará saber al usuario mediante dos formas, la primera es dentro de la misma pantalla (debajo de la pantalla) con un mensaje en rojo, la segunda forma es mediante un mensaje que saldrá por sobre la pantalla, a la que será necesario dar al botón aceptar para indicarle al sistema que ya ha leído la información.



En la figura anterior se ve un mensaje de error tras haber presionado el botón “Entrar” sin haber ingresado el Rut. Luego podrá nuevamente ingresar el Rut, sin indicarle al sistema que ha leído el mensaje.



En la figura anterior se muestra un mensaje que es necesario que el usuario lo lea y para ello le indica explícitamente al sistema que lo ha leído mediante el botón “Aceptar”.

## INICIO

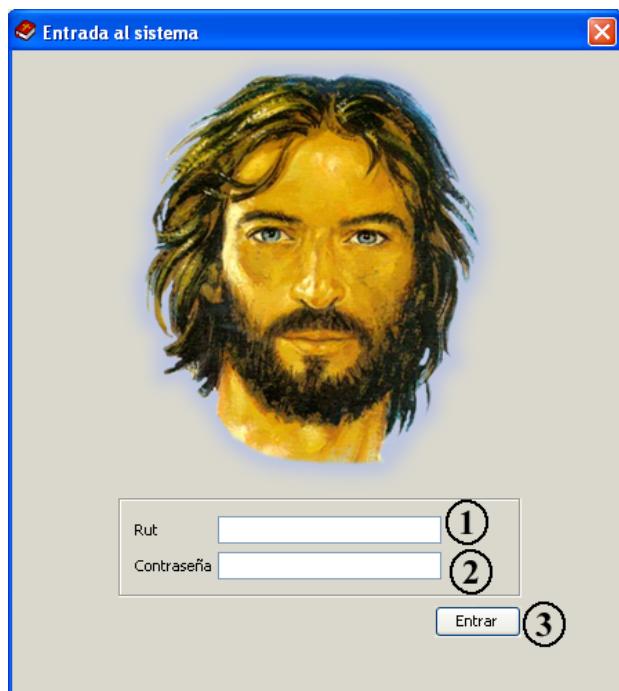
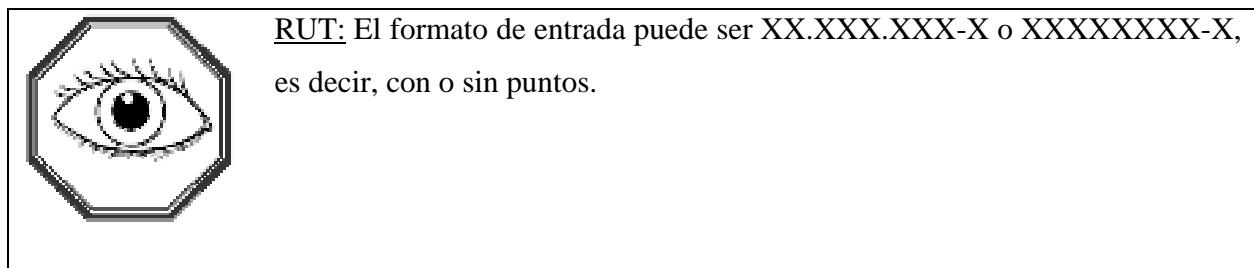
Al iniciar el sistema de biblioteca del Seminario Metropolitano Concepción lo primero que podremos observar, es la pantalla de inicio, esta se presenta a continuación.



- 1.- Indica el progreso de carga del software, en caso de que este no se moviera por un tiempo prolongado, se hace necesario la reiniciación del sistema operativo, ya que lo más seguro, es que el problema provenga de allí.

## IDENTIFICARSE

Pantalla de identificación para el bibliotecario.



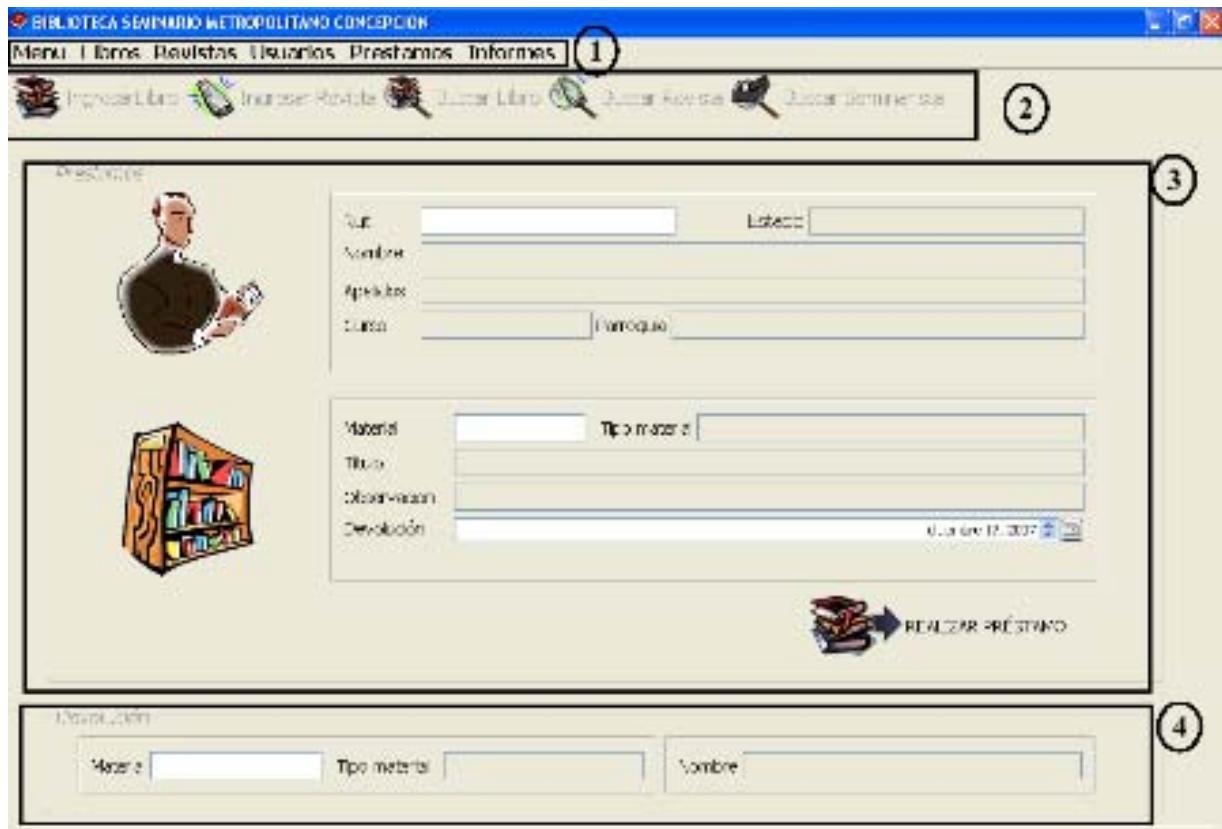
1.- Campo en donde el bibliotecario deberá ingresar su RUT.

2.- Campo en donde el usuario deberá ingresar la clave, al comienzo ésta será el mismo RUT (con puntos), pero posteriormente podrá ser cambiada por la que el bibliotecario estime conveniente.

3.- Botón que le indica al sistema que los datos ya han sido ingresados.

## PANTALLA PRINCIPAL

Pantalla principal del sistema, en la que el usuario tendrá el acceso a todas las opciones mediante los menús respectivos, además como actividad principal, en esta pantalla se gestionarán los préstamos y devoluciones a los seminaristas.



- 1.- Menú principal, en la que tendrá todas las opciones que el sistema posee, estas opciones se describen posteriormente.
- 2.- Accesos rápidos a las principales opciones del sistema, estas serán descritas más adelante.
- 3.- Sección en la cual se realizan los préstamos del material de la biblioteca a los seminaristas.
- 4.- Sección para realizar la devolución del material de un seminarista.

# PRESTAMOS

Préstamos

Rut **1.1** Estado

Nombre

Apellidos

Curso Parroquia

Material **2.1** Tipo material

Título

Observación

Devolución diciembre 17, 2007 **2.2**

REALIZAR PRÉSTAMO **3**

1.- Marco dedicado al identificar al seminarista que desea realizar un préstamo en el sistema.

1.1.- Campo en la cual se deberá ingresar el RUT del seminarista, luego de ser ingresado el sistema proporcionará la información asociada a los demás campos del marco.

2.- Marco dedicado a la identificación del material (libro o revista) que se prestara al seminarista ingresado.

2.1.- Campo en la que se ingresa el número de registro del material que se desea prestar, este puede ser pistoleado por el lector de códigos de barra o ingresado por teclado, luego el sistema proporcionará la información del material.

2.2.- Campo en la cual, por defecto, se indicará la fecha de devolución del material, este podrá ser cambiado al instante por el bibliotecario, para dar una mayor flexibilidad del servicio.

3.- Botón, en la cual se le indica al sistema que realice el préstamo del material al seminarista indicado.

## DEVOLUCIÓN

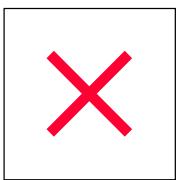
The screenshot shows a software window titled "Devolución". It contains three input fields: "Material" (containing the value "1"), "Tipo material", and "Nombre". A red circle highlights the "Material" field, specifically the digit "1" inside it.

- 1.- Campo en la cual se ingresa el código de registro del material que se está devolviendo a la biblioteca, este puede ser pistoleado con el lector de códigos de barra o ingresado directamente por teclado.

## SECCIÓN LIBROS



### INGRESO DE LIBRO 1/3



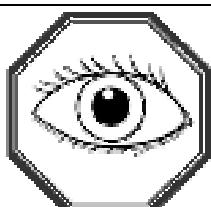
*Para poder ingresar un libro al sistema, el bibliotecario deberá proveer la información que el sistema le solicite, este proceso cuenta con 3 pantallas principales, de las cuales se desprenden acciones como: Ingresar Una Copia, Ingresar Autor, Ingresar Editorial, Ingresar Ediciones, Ingresar Serie y Agregar nuevas materias.*



**Título:** En caso de ingresar un título que ya existe, el sistema dará una alerta. Con la opción de llenar todos los campos con la información del libro que ya había sido ingresado, con el fin de reutilizar esta información.

**ISBN:** Al igual que el título, al ingresar un ISBN que ya ha sido ingresado, el sistema dará una alerta.

**Clasificación:** La clasificación debe ser única, por lo que el sistema no permitirá ingresar dos libros con la misma clasificación. El libro deberá, necesariamente, ser ingresada como copia.



**Tipo:** Indica a que tipo pertenece el libro que se está ingresando, existen dos: **Referencia**, que significa que el libro es de alta demanda y además existen copias limitadas y **No referencia** lo cual no son libros muy codiciados y existen más de una copia.

**Registro anterior:** Este atributo es ingresado con el fin de convivir con el sistema anterior, este corresponde al registro con que el libro había sido

ingresado en el sistema antiguo, en el caso de ingresar un libro que no había sido ingresado anteriormente, no será necesario ser ingresado.

Clasificación: Atributo único, por lo general compuesto por el código Dewey, código cutter, año y otros (podría ser volumen, tomo, etc).

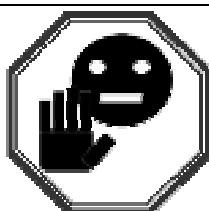
**Ingreso de Libros**

**Ingrese los datos del libro**

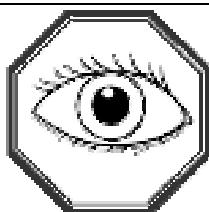
Título	<input type="text"/>	Ingresar copia <b>1</b>
ISBN	<input type="text"/>	Cantidad páginas <input type="text"/>
Volumen	<input type="text"/>	Tomo <input type="text"/>
Tipo	<input type="text"/> NO REFERENCIA <b>2</b>	Registro anterior <input type="text"/>
Autor principal		
Autor	<input type="text"/>	Agrega <b>3</b>
Clasificación		
Edición		
Largo	<input type="text"/>	Año <input type="text"/>
Ancho	<input type="text"/>	Agregar <b>4</b>
Notas	<b>5</b>	
Medidas		
Siguiente > <b>6</b>		

- 1.- Botón que permite el ingreso de una copia de un libro que ya fue ingresado anteriormente, luego se debe identificar el libro con la clasificación para asignarle una copia.
- 2.- Lista en la que aparecen los tipos de libros, sobre el cual se determina la cantidad de días que será prestado el material.
- 3.- Botón que permite ingresar un autor, si es que este no está en el sistema.
- 4.- Botones que permiten el ingreso de una nueva editorial, ediciones y serie respectivamente
- 5.- Permite ingresar cualquier observación acerca del libro.
- 6.- Permite dar paso a la segunda etapa del ingreso de un libro.

## INGRESO DE LIBRO 2/3



Finalizar / Ingresar Libro: El sistema no permitirá el ingreso de un libro si es que éste no ha sido vinculado a, por lo menos, una materia.



Agregar nueva materia: No se debe abusar mucho en el ingreso de materias, la idea es ingresar materias generales y tratar de ocupar las materias existentes para que el sistema de búsqueda por materias sea efectiva.

**Segunda etapa ingreso de libro**

**Segunda etapa ingreso del libro**

Libro \_\_\_\_\_ Autor principal \_\_\_\_\_ Cutter \_\_\_\_\_

Seleccione Autor (es) secundario (s)

**Lista de autores**      **Autor(es) del libro**

①      ②

>>      <<

③      Agregar nuevo autor ④

Seleccione materia (s)

**Lista de materias**      **Materia (s) del libro**

⑤      ⑥

>>      <<

⑦      Agregar nueva materia ⑧

Finalizar / ingresar Libro ⑨

This screenshot shows the second step of a library book entry process. It includes fields for the book title, main author, and cutter. Below these are sections for secondary authors and subjects. The interface features two columns of lists: 'Lista de autores' (List of authors) and 'Autor(es) del libro' (Authors of the book) for managing secondary authors, and 'Lista de materias' (List of subjects) and 'Materia (s) del libro' (Subjects of the book) for managing subjects. Each column has a 'Listado' section (list view) on the left and a 'Materiales' section (material view) on the right. Navigation buttons like '>>' and '<<' are used between the lists. Buttons for adding new entries ('Agregar nuevo autor' and 'Agregar nueva materia') are located at the bottom of each column. A final button at the bottom right allows users to 'Finalizar / ingresar Libro' (Finish / enter book).

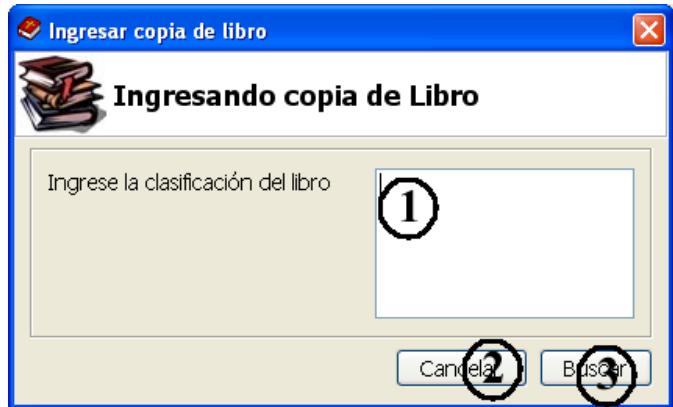
- 1.- Lista que contendrá todos los autores registrados en el sistema, ordenados alfabéticamente, estos se pueden elegir con el mouse o tecleando la primera letra correspondiente al autor, además la elección podrá variar con los botones de flechas que el teclado posea.
- 2.- Lista que contendrá los autores secundarios del libro que se está ingresando.
- 3.- Botones de desplazamiento de autores de una lista a otra, es decir, si se encuentra un autor en la lista de autores generales del sistema (1) podrá ser movida a la lista de autores secundarios del libro mediante los botones de acuerdo a la trayectoria que estos indican.
- 4.- Botón que permite ingresar un nuevo autor en el caso que no exista en las listas.
- 5.- Igual que el punto (1) pero esta vez no se tratarán autores, sino materias.
- 6.- Igual que el punto (2) pero esta vez no se tratarán los autores secundarios, sino las materias que correspondan al libro, estas materias pueden también ser palabras claves que sirvan para efectuar las búsquedas.
- 7.- Igual al punto (3) pero esta vez es para mover las materias de una lista a otra.
- 8.- Botón que permite el ingreso de una nueva materia, es decir, ingresar una materia que no existe en el sistema.
- 9.- Botón que informa al sistema que ya puede registrar el libro.

## INGRESO DE LIBRO 3/3



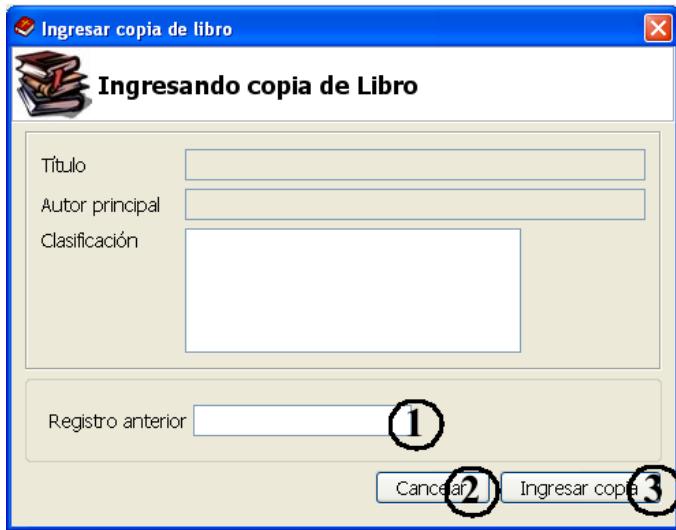
- 1.- Indica el registro con el que se ingresó la copia del libro.
- 2.- Botón que envía instrucción a la impresora para que imprima el código de barras del libro.
- 3.- Botón que envía instrucción a la impresora para que imprima una etiqueta con la clasificación del libro.
- 4.- Botón que indica al sistema que la operación de ingreso de libro ha terminado.

## INGRESAR COPIA DE UN LIBRO 1/3



- 1.- Permite ingresar la clasificación del libro, al que desea ingresar una copia.
- 2.- Botón que permite cancelar el proceso de ingreso de copia.
- 3.- Botón que inicia la búsqueda del libro asociado a la clasificación, de modo que se muestre en pantalla los resultados de la búsqueda para luego ingresar la copia.

## INGRESAR COPIA DE UN LIBRO 2/3



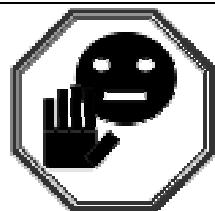
- 1.- Permite el ingreso del número de registro que poseía el libro en el sistema anterior.
- 2.- Botón que cancela el proceso del ingreso de una copia.
- 3.- Botón que informa al sistema que ingrese una copia de un libro.

## INGRESAR COPIA DE UN LIBRO 3/3



- 1.- Indica el registro con el que se ingresó la copia del libro.
- 2.- Botón que envía instrucción a la impresora para que imprima el código de barras del libro.
- 3.- Botón que envía instrucción a la impresora para que imprima una etiqueta de clasificación.
- 4.- Botón que indica al sistema que la operación de ingreso de libro ha terminado.

## INGRESAR AUTOR



F. nacimiento: El sistema no permitirá el ingreso de un autor que presente una fecha de nacimiento igual o mayor a la fecha de muerte.

F. muerte: El sistema no permitirá el ingreso de un autor que presente una fecha de muerte igual o menor que la fecha de nacimiento.

Cutter: El sistema no permitirá el ingreso de un autor con el mismo código cutter que otro autor ingresado con anterioridad.



1.- Sector en la cual se deben ingresar los datos del autor.

2.- Si a un autor se le desconoce la fecha de nacimiento o fecha de muerte puede ser omitida.

3.- Permite ingresar el código cutter del autor, este debe ser único, por lo cual el sistema no permitirá el ingreso de un autor si el código ya ha sido ingresado para otro autor.

4.- Botón que informa al sistema que ingrese el autor.

## INGRESAR EDITORIAL



Ingrese Editorial: El sistema no permitirá el ingreso de una editorial que ya ha sido ingresada en el sistema.

Ingresar editorial

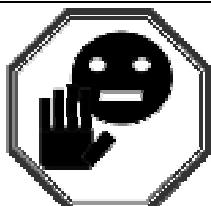
Ingrese editorial

Cancelar 1 Ingresar editorial 2

1.- Botón que cancela el proceso de ingreso de una editorial.

2.- Botón que informa al sistema que ingrese la una editorial.

## INGRESAR SERIE



Ingresar serie: El sistema no permitirá el ingreso de una serie que ya existe en el sistema.

Ingresar serie

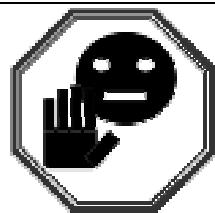
Ingresar Serie

Cancelar 1 Ingresar serie 2

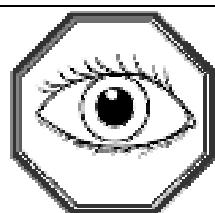
1.- Botón que cancela el proceso de ingreso de una serie.

2.- Botón que informa al sistema que ingrese una serie.

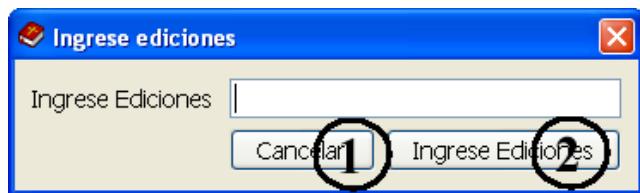
## INGRESAR EDICIONES



Ingrese Ediciones: El sistema no permitirá el ingreso de ediciones que ya existan en el sistema.



Ediciones: En la literatura cristiana existen ediciones, lo que vendría siendo una editorial, por lo que un libro podría tener ediciones o editorial.



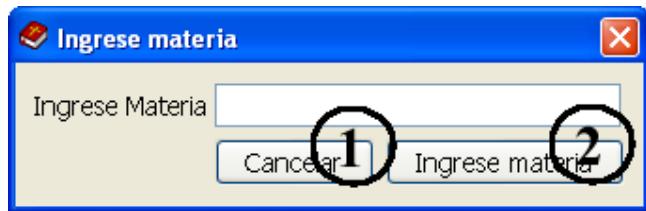
1.- Botón que cancela el proceso de ingreso de ediciones

2.- Botón que informa al sistema que ingrese ediciones.

## AGREGAR NUEVA MATERIA



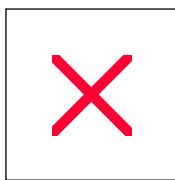
Ingrese materia: El sistema no permitirá el ingreso de una materia que ya exista en el sistema.



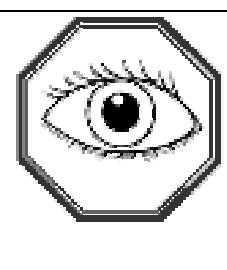
1.- Botón que cancela el proceso de ingreso de una materia.

2.- Botón que informa al sistema que ingrese la materia.

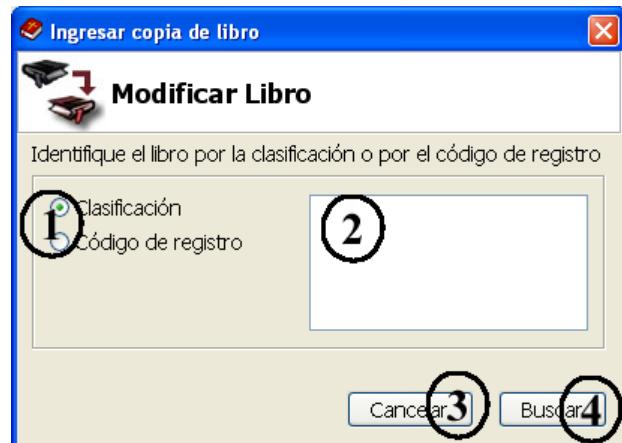
## MODIFICACIÓN DE LIBRO 1/3



*El proceso de modificación de un libro consta de tres pantallas, idénticas a las que se utilizaron para ingresar el libro, permitiendo la modificación de los atributos respectivos, los atributos no serán modificados hasta que el usuario lo especifique explícitamente al término de la operación. Al igual que el ingreso de los libros se desprenden las acciones Ingresar Autor, Ingresar Editorial, Ingresar Ediciones, Ingresar Serie y Agregar nuevas materias (Estas no serán descritas por que utiliza la mismas pantallas y acciones que las mencionadas en el ingreso de Libros.*

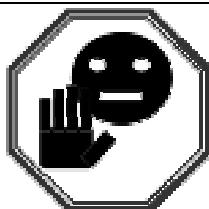


Código de registro: En el caso de pistolear o ingresar por teclado el código de registro, el sistema lo asociará al código de clasificación.

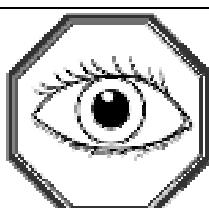


- 1.- Se selecciona si el libro a modificar será identificado mediante el código de registro o por su clasificación.
- 2.- Podrá ingresar la clasificación o código de registro según corresponda.
- 3.- Botón que cancelará el proceso de modificación.
- 4.- Botón que inicia la búsqueda de la información asociada a la clasificación o código de registro, para comenzar a modificar los atributos que se estimen conveniente.

## MODIFICAR LIBRO 2/3



Clasificación: No se puede modificar la clasificación por una que ya exista en el sistema. Además si se modifica por otra inexistente, otro libro podría ocupar la clasificación



Clasificación: Modificar este atributo es una acción riesgosa, se recomienda no modificarla a menos que sea estrictamente necesario.

**Modificando libro**

**Modificar Libro**

Título	LIBRO DE PRUEBA	Cantidad páginas	250
ISBN	11.1111.1111-XX	Tomo	II
Volumen	5		
Tipo	NO REFERENCIA		

*Autor principal*

Autor ULLOA MERINO, CLAUDIO ENRIQUE (CU22) Agregar

*Clasificación*

123 . 4  
CU22  
2007

*Edición*

Edición NUEVA EDICION Año 2007  
Editorial EDITORIAL PRUEBA Agregar  
Ediciones EDICIONES PRUEBA Agregar  
Serie SERIE PRUEBA Agregar  
País CHILE Ciudad TALCAHUANO

*Medidas*

Largo 40  
Ancho 15

*Nota*

ESTE LIBRO FUE INGRESADO PARA UNA PRUEBA DEL SOFTWARE

Siguiente >

## MODIFICAR LIBRO 3/3

 **Modificar libro** X

### Modificar Libro

Título (anterior) LIBRO DE PRUEBA  
Autor principal (anterior) ULLOA MERINO, CLAUDIO ENRIQUE Cutter CU22

Libro (modificado) LIBRO DE PRUEBA  
Autor principal (modificado) ULLOA MERINO, CLAUDIO ENRIQUE Cutter CU22

Seleccione Autor(es) secundario(s)

Lista de autores	Autor(es) del libro

>>  
<<

Agregar nuevo autor

Seleccione materia(s)

Lista de materias	Materia(s) del libro
<p>RELIGION</p> <p>TEOLOGIA</p>	<p>MATERIA DE PRUEBA</p> <p>PRUEBA</p>

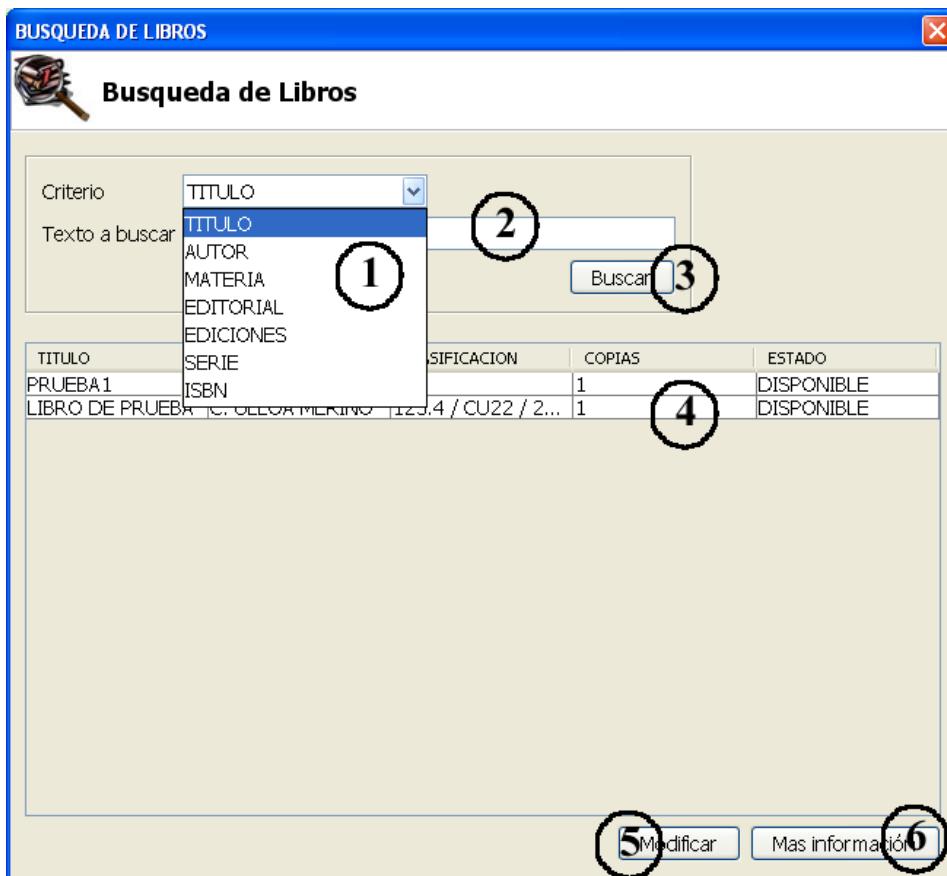
>>  
<<

Agregar nueva materia

Finalizar / Modificar Libro

## BUSCAR LIBRO

*El proceso de búsqueda de libros se puede realizar por medio de varios criterios y estos serán buscados dentro de todos los libros que existen en el sistema, la palabra puede ser exacta o puede ser una palabra contenida en otra (teo esta contenida en teología)*



- 1.- Criterios con las cuales se pueden realizar las búsquedas de los libros en el sistema.
- 2.- Permite ingresar las palabras que serán asociadas a las búsquedas de acuerdo al criterio que se escogió en (1).
- 3.- Botón que da inicio a la búsqueda de los libros.
- 4.- Tabla en la que aparecerán las coincidencias de la búsqueda realizada.
- 5.- Botón que permite modificar un libro, si es que este se ha seleccionado explícitamente con el mouse, una vez que el botón se ha presionado se inicia la modificación del libro desde la pantalla 2 de modificación, obviando la pantalla de identificación del libro.

6.- Botón que permite ver toda la información de un libro que se ha seleccionado con el mouse.

## MÁS INFORMACIÓN

**Información del libro**

 **Información del Libro**

Título	LIBRO DE PRUEBA		
Autor principal	ULLOA MERINO, CLAUDIO ENRIQUE (CU22)	Cantidad páginas	250
ISBN	11.1111.1111-XX	Volúmen	5
Tomo	II	Tipo	NO REFERENCIA

**Clasificación**

123.4
CU22
2007

**Edición**

Edición	NUEVA EDICION	Año	2007
Editorial	EDITORIAL PRUEBA	País	CHILE
Ediciones	EDICIONES PRUEBA	Ciudad	CAHUANO
Serie	SERIE PRUEBA		

**Autores secundarios**

**Materias**

MATERIA DE PRUEBA
PRUEBA

**Medidas**

Largo	40
Ancho	15

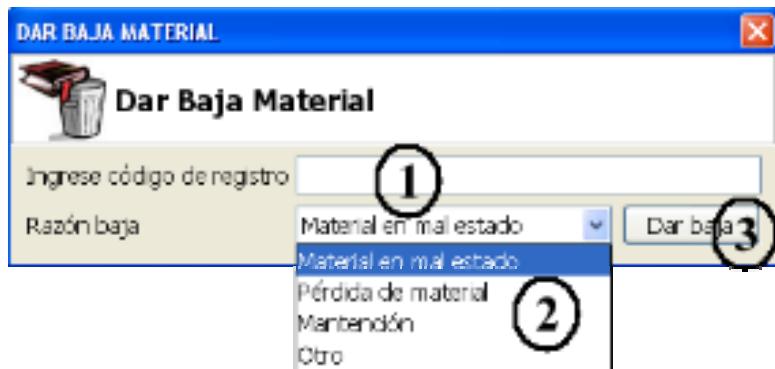
**Notas**

RO FUE INGRESADO PARA UNA PRUEBA DEL SOFTWARE

OK

## DAR BAJA MATERIAL

El proceso de Dar de baja un material, permite que un libro o revista quede al margen de toda operación sobre él. Para ello el sistema le pedirá que se seleccione la opción por la cual el libro se está dando de baja.



- 1.- Permite que se identifique el libro que se está dando de baja, este puede ser pistoleado por el lector de códigos de barra o ingresado por teclado.
- 2.- Lista en la cual se encuentran las razones para dar de baja un material.
- 3.- Botón que informa al sistema que inicie la baja del libro.

## DAR ALTA MATERIAL

*El proceso de Dar Alta a un material, permite que un libro o revista pueda volver a ser utilizado después que se ha dado de baja.*

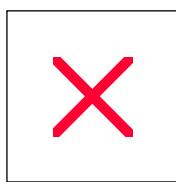


- 1.- Permite que se identifique el libro que se está dando de alta, este puede ser pistoleado por el lector de códigos de barra o ingresado mediante el teclado.
- 2.- Botón que informa al sistema que de inicio al alta del material.

## SECCIÓN REVISTAS

The screenshot shows a software application window titled "BIBLIOTECA SEMINARIO METROPOLITANO CONCEPCIÓN". The menu bar includes "Menú", "Libros", "Revistas" (which is highlighted in blue), "Usuarios", "Prestamos", and "Informes". A sub-menu for "Revistas" is open, listing the following options: "Nuevo de Revista", "Modificación de Revista", "Modificación de Número", "Nuevo de Número", "Dar alta en stock", and "Dar alta general".

### INGRESO DE REVISTA



Para poder ingresar una revista al sistema, el bibliotecario deberá proveer la información que el sistema le solicite. Además de éste proceso se desprenden acciones como: Agregar nueva revista, Agregar copia de un número y agregar contenido.

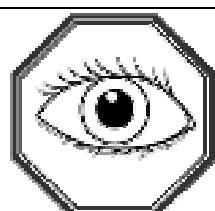


Número: El sistema no permitirá el ingreso de una revista de con el mismo número y nombre, esta deber ser una copia de numero de revista.

Desde: El sistema no permitirá que una periodicidad comience con una fecha mayor a la del término.

Hasta: El sistema no permitirá que la periodicidad termine con una fecha menor a la del inicio.

Contenido: El sistema no permitirá el ingreso de una revista si esta no tiene algún contenido que se le asocie.



Desde: Al momento de cambiar el comienzo de la periodicidad (Desde) el término (Hasta) cambiará automáticamente según la periodicidad de la revista que se ingresó como dato general de la revista.

Número: Al ingresar un número que ya existe asociado a la revista que se escogió el sistema le alertara y dará opción de ingresar una copia, de no ser

así no permitirá el ingreso del número.



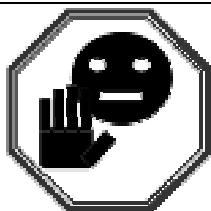
- 1.- Botón que da inicio al proceso de agregar una nueva revista, es decir, una revista a la cual jamás se le ha asociado un número de revista y el nombre no aparece en el sistema.
- 2.- Botón que da inicio al proceso de agregar una copia de revista, es decir, agregar el número de una revista en particular más de una vez.
- 3.- Lista de la cual despliegan todas las revistas que están ingresadas en el sistema, a fin de asociarle un nuevo número.
- 4.- Lista en la cual se encuentran todos los contenidos existentes en el sistema, estos aparecen ordenados alfabéticamente y se pueden elegir mediante mouse, teclado (flechas o mediante la primera letra del contenido). Estos contenidos no están asociados al número que se está ingresando pero se puede asociar.
- 5.- Botones que permiten desplazar de un lado a otro el contenido de una revista, es decir, el contenido puede pasar a formar parte del número que se está ingresando.

6.- Lista de todos los contenidos que se están asociando al número de revista que se está ingresando.

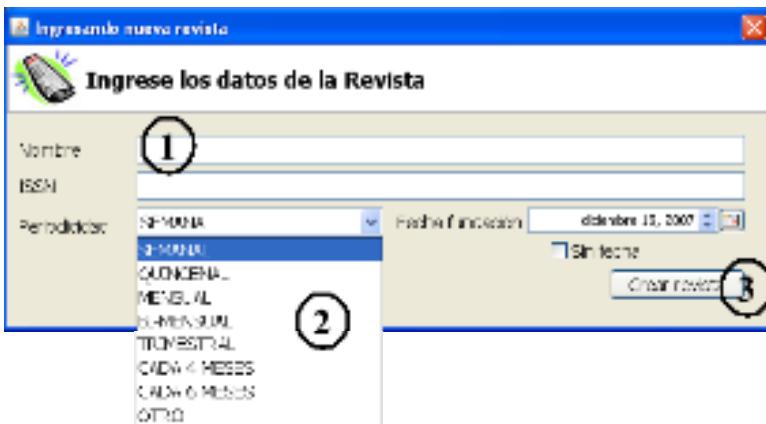
7.- Botón que permite el ingreso de un contenido que no existe en el sistema.

8.- Botón que informa al sistema que ingrese la revista.

## AGREGAR NUEVA REVISTA



Nombre: El sistema no permitirá el ingreso de la revista si esta no se ha digitado el nombre.

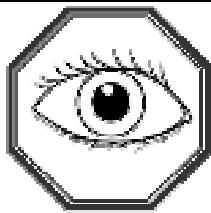


1.- Permite ingresar el nombre de la revista que se está ingresando.

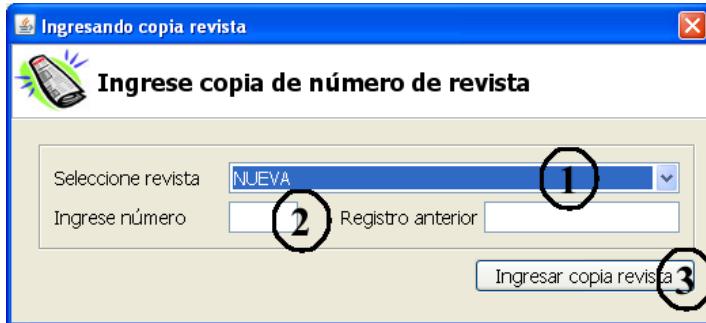
2.- Lista de la cual se despliegan las periodicidades existentes en el sistema.

3.- Botón que informa al sistema que ingrese la nueva revista.

## AGREGAR COPIA DE UN NÚMERO



Número: El número ingresado en el sector dispuesto para aquello debe existir previamente en el sistema.



- 1.- Lista de la cual aparecen todas las revistas existentes en el sistema.
- 2.- Permite el ingreso del número de revista a la cual se desea ingresar una copia.
- 3.- Botón que informa al sistema que ingrese la copia de un número de revista.

## AGREGAR CONTENIDO



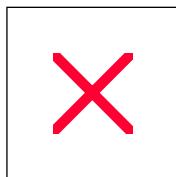
Ingrese Contenido: El sistema no permitirá el ingreso de un contenido que ya existan en el sistema.



1.- Botón que cancela el proceso de ingresar un nuevo contenido.

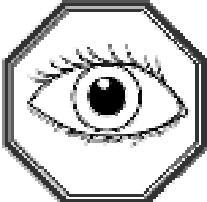
2.- Botón que informa al sistema que ingrese el contenido.

## MODIFICACIÓN DE REVISTA 1/2



*El proceso de modificación de una revista consta de dos pantallas, idénticas a las que se utilizaron para ingresar la nueva revista, permitiendo la modificación de los atributos respectivos, los atributos no serán modificados hasta que el usuario lo especifique explícitamente al término de la operación.*

*Es necesario recordar que al modificar una revista, se modifican los atributos generales, es decir estos atributos serán cambiados para todos los números que la revista posea.*

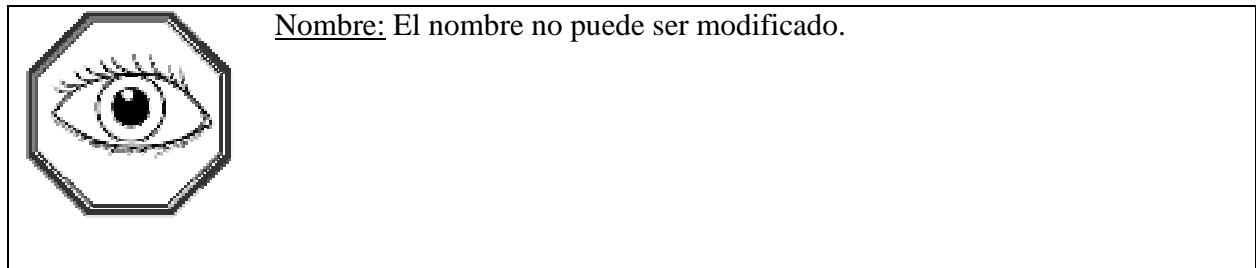


Código de registro: El código de registro puede ser ingresado por teclado o pistoleado por el lector de códigos de barra.



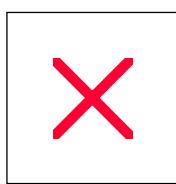
- 1.- Lista de la cual despliegan la formas de identificación del a revista que se desea modificar.
- 2.- Permite el ingreso de la revista que se desea modificar (por título o código de registro).
- 3.- Botón que da inicio a la búsqueda de la revista que se desea modificar.

## MODIFICACIÓN DE REVISTA 2/2

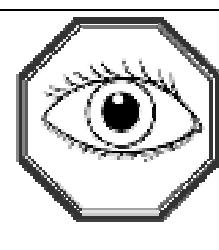


1.- Botón que informa al sistema que modifique la revista.

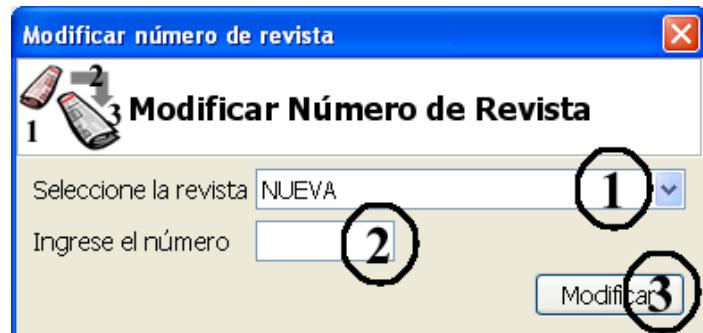
## MODIFICACIÓN DE NÚMERO 1/2



*El proceso de modificación de un número de revista consta de dos pantallas, idénticas a las que se utilizaron para ingresar un nuevo número, permitiendo la modificación de los atributos respectivos, los atributos no serán modificados hasta que el usuario lo especifique explícitamente al término de la operación. Es necesario recordar que al modificar un número de una revista, se modificarán solo los atributos que le corresponden a ese número en particular, si existiesen copias de ese número, estos también serán modificados. No se dará detalle de la modificación ya que estos se detallan en las pantallas de ingreso de revistas.*

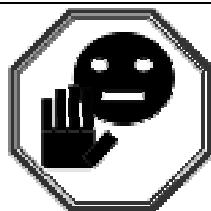


Número: Al modificar una copia de una revista se debe ingresar el número que corresponda y este debe existir en el sistema.



- 1.- Lista en la que aparecen los nombres de todas las revistas que han sido ingresadas en el sistema.
- 2.- Permite el ingreso del número asociado a la revista que se desea modificar.
- 3.- Botón que inicia la búsqueda del número de la revista que se desea modificar.

## MODIFICACIÓN DE NÚMERO 2/2



Desde: Al igual que en el ingreso de revista, el sistema no permitirá la modificación de la periodicidad con una fecha de inicio mayor que la de termino.

Hasta: Al igual que en el ingreso de revista, el sistema no permitirá la modificación de la periodicidad con una fecha de termino mayor que la de termino.

Contenido: El sistema no permitirá que se eliminen todos los contenidos asociados a la revista.

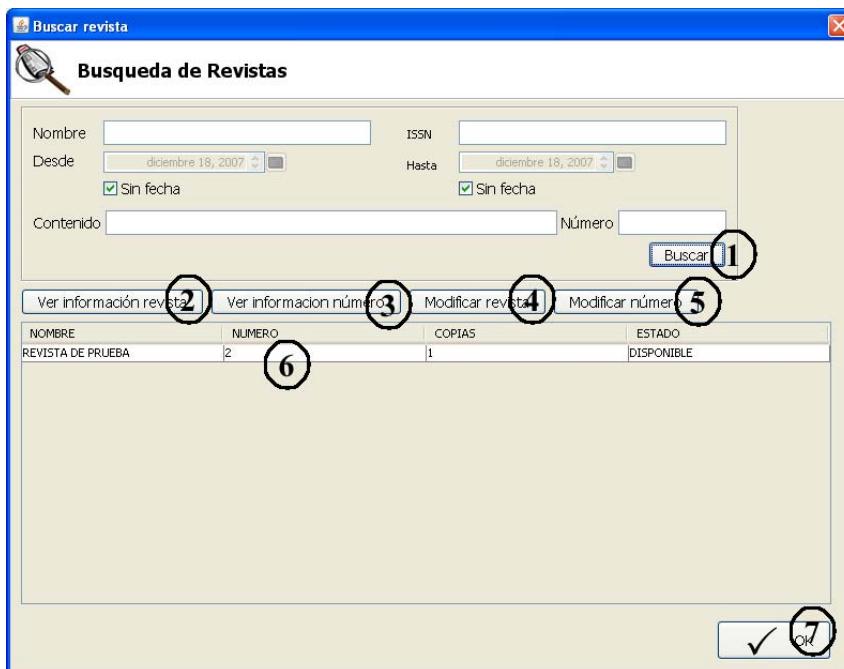
Modificar número de revista

1 2 3 Modificar Número de Revista

General		
Nombre	REVISTA DE PRUEBA	
ISSN	99.999.999-9	
Año	1985	
Volumen		
Periodicidad		
Desde	diciembre 1, 1984	
Hasta	enero 1, 1985	
Contenido		
OTRO CONTENIDO	>> <<	CONTENIDO DE PRUEBA
Agregar contenido		
Modificar número de revista		

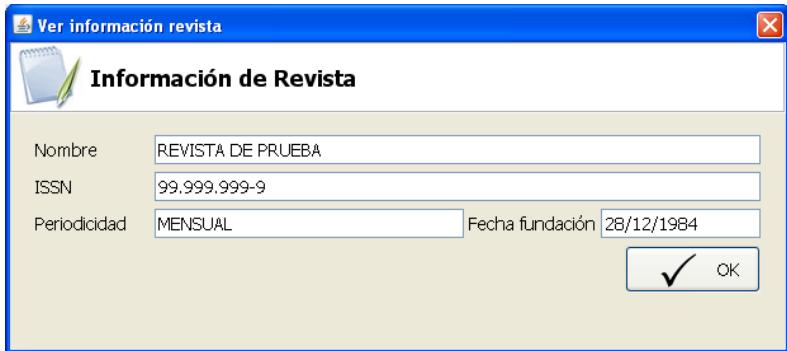
## BUSCAR REVISTA

*El proceso de búsqueda de una revista se puede realizar por medio de varios criterios y estos serán buscados dentro de todas las revistas que existan en el sistema, la palabra puede ser exacta o contenida dentro de otra (teo esta contenida en teología).*

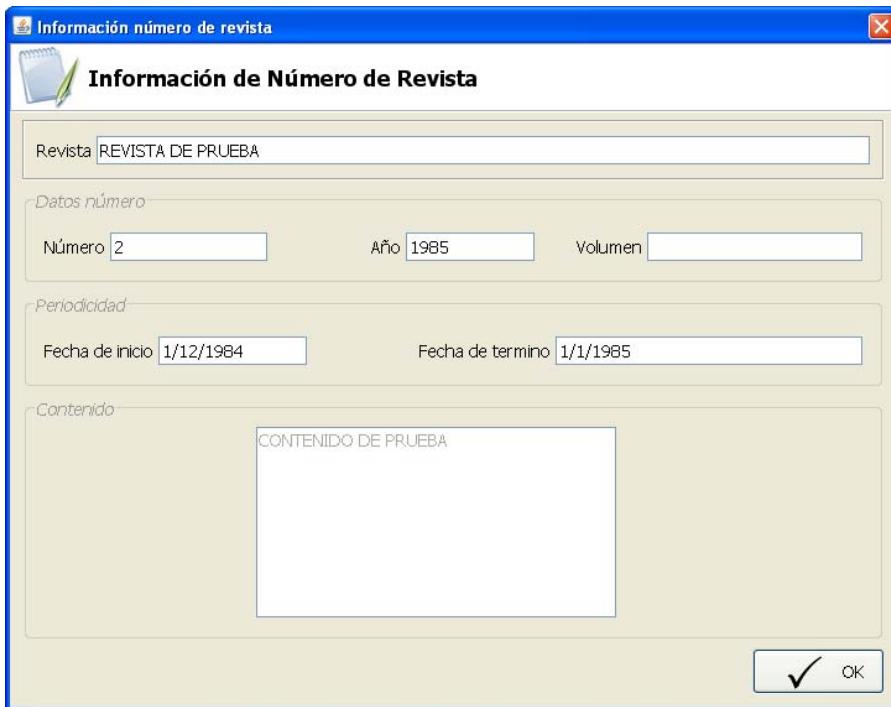


- 1.- Botón que da inicio a la búsqueda de las revistas según las concordancias que se encuentren respecto a los atributos que se ingresaron.
- 2.- Botón que permite ver la información de una revista que se seleccionó en la tabla de resultados.
- 3.- Botón que permite ver la información del número que se seleccionó de la tabla de resultados.
- 4.- Botón que permite la modificación de la revista previamente seleccionada de la tabla de resultados.
- 5.- Botón que permite la modificación del número de una revista previamente seleccionada de la tabla de resultados.
- 6- Tabla de resultados que arroja la búsqueda de revistas.
- 7.- Botón que indica al sistema, que ya se realizó lo que se necesitaba.

## VER INFORMACIÓN REVISTA



## VER INFORMACIÓN NÚMERO



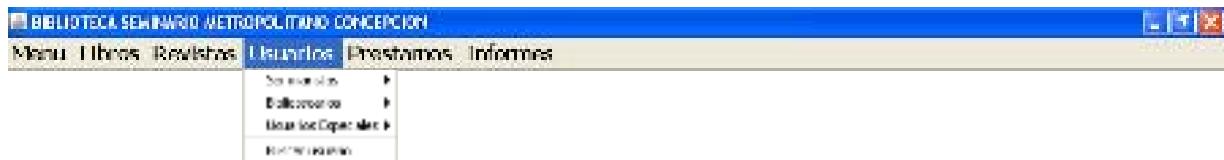
## DAR ALTA MATERIAL

*Es exactamente lo mismo que dar el de un libro, incluso se puede dar de alta una revista en la sección de libros y un libro en la sección de revistas.*

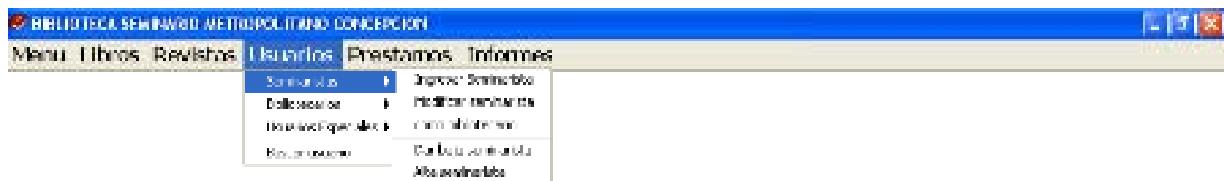
## DAR BAJA MATERIAL

*Es exactamente lo mismo que dar de alta un libro, incluso se puede dar de baja una revista en la sección de libros y un libro en la sección de revistas.*

## USUARIOS



## SEMINARISTAS



## INGRESAR SEMINARISTA

Para poder ingresar un seminarista, el bibliotecario deberá proveer la información que el sistema le solicite. Además de este proceso se desprenden acciones como: Ingresar Curso e Ingresar Parroquia.



RUT: El sistema no permitirá el ingreso de un seminarista al que se le asocie un Rut que ya existe en el sistema.

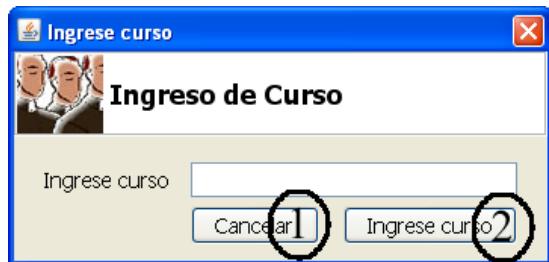
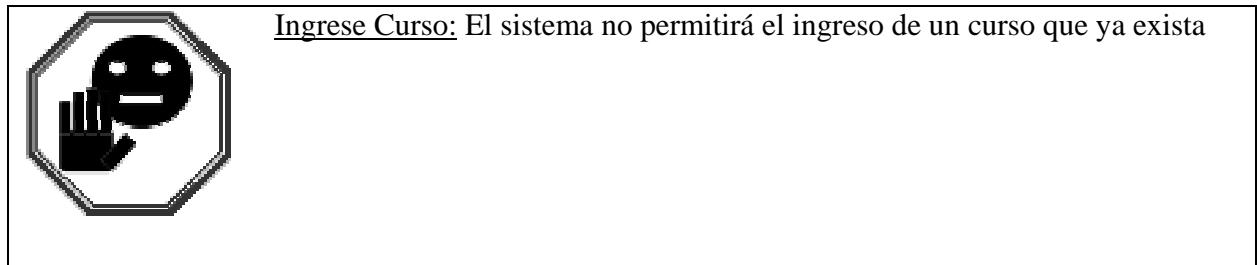
**Ingresar seminarista**

**Ingreso de Seminarista**

Nombres	<input type="text"/>	
Apellidos	<input type="text"/>	
Rut	<input type="text"/>	Dirección <input type="text"/>
Teléfono	<input type="text"/>	Celular <input type="text"/>
Año Ingreso	<input type="text"/>	Fecha nacimiento <input type="text"/> diciembre 19, 2007 <input type="button" value=""/>
<input type="checkbox"/> Sin fecha		
Curso	<input type="text"/> PROPE	<input type="button" value="Agregar"/> <input type="button" value=""/>
Parroquia origen	<input type="text"/> SANTA CECILIA / CONCEPCION	<input type="button" value="Agregar"/> <input type="button" value=""/>
<input type="button" value="Finalizar Ingreso"/> <b>(3)</b>		

- 1.- Botón que inicia el proceso de ingresar un nuevo curso.
- 2.- Botón que inicia el proceso de ingresar una nueva parroquia.
- 3.- Botón que informa al sistema que puede ingresar al seminarista.

## AGREGAR CURSO



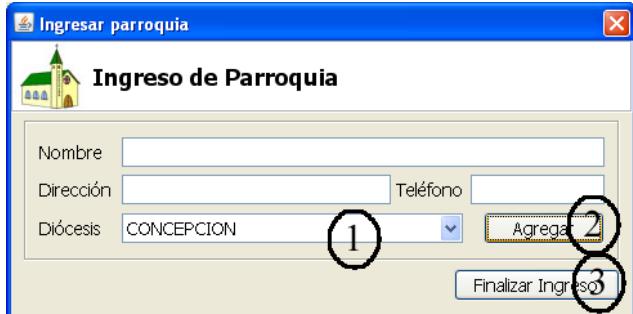
1.- Botón que cancela el proceso de ingresar un curso.

2.- Botón que informa al sistema que ingrese el curso.

## AGREGAR PARROQUIA



Nombre: El sistema no permitirá el ingreso de un nombre de parroquia que ya exista.

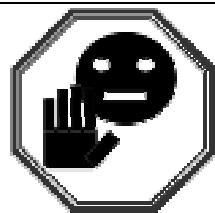


1.- Lista que contiene todas las diócesis existentes en el sistema.

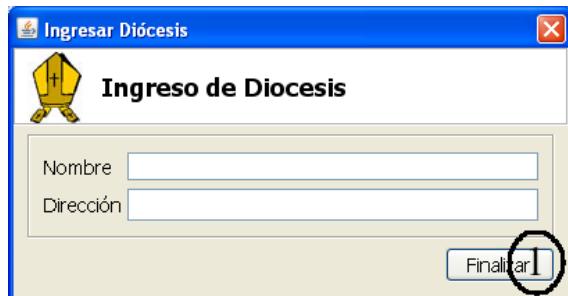
2.- Botón que inicia l proceso de ingreso de una nueva diócesis.

3.- Botón que indica al sistema que ingrese la parroquia.

## AGREGAR DIÓCESIS



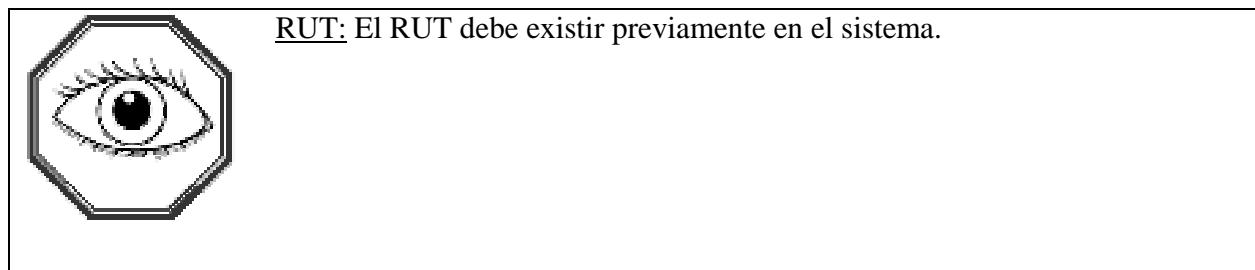
Nombre: El sistema no permitirá el ingreso de un nombre de diócesis que ya exista.



1.- Botón que indica al sistema que ingrese la diócesis.

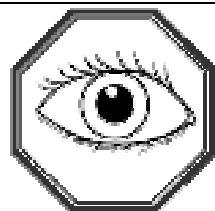
## MODIFICAR SEMINARISTA 1/2

*El proceso de modificación de un seminarista consta de dos pantallas, una de identificación del usuario que se desea modificar y otra idéntica al ingreso de seminaristas por lo cual se esta no se dará mayor detalle.*



- 1.- Permite ingresar el RUT del seminarista.
- 2.- Botón que inicia la búsqueda del seminarista que se desea modificar.

## MODIFICAR SEMINARISTA 2/2



ESTADO: Si un seminarista se encuentra suspendido y el bibliotecario desea de todas formas permitirle un préstamo, puede cambiar el estado a REGULAR.

**Modificación seminarista**

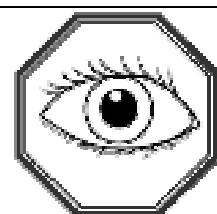
**Modificar Seminarista**

Nombres	CRISTIAN ANTONIO		
Apellidos	ULLOA MERINO		
Rut	9.281.375-4	Dirección	LAGO PETROHUE 235
Teléfono	5555555	Celular	09-99999
Año Ingreso	2002	Fecha nacimiento	diciembre 19, 2007
<input checked="" type="checkbox"/> Sin fecha			
Curso	PROPE	Agregar	
Parroquia origen	SANTA CECILIA / CONCEPCION	Agregar	
ESTADO	REGULAR	1	2 Reinicar clave de usuario
Finalizar modificación			

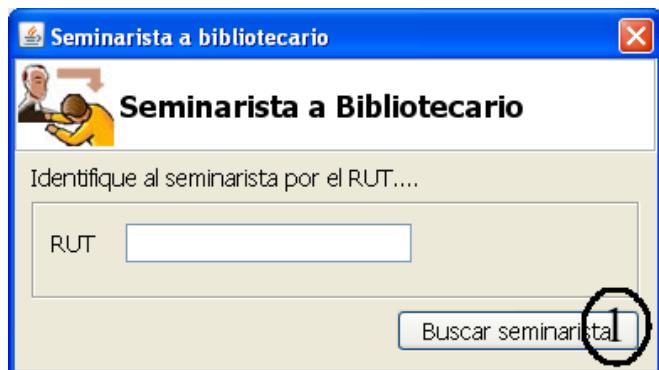
- 1.- Lista en la cual se encuentran los estados que puede tomar el seminarista. (REGULAR, SUSPENDIDO)
- 2.- Opción para reiniciar la clave del seminarista, esta será reiniciada con el RUT en formato XX.XXXX.XXX-X.
- 3.- Botón que indica al sistema que puede modificar al usuario.

## COMO BIBLIOTECARIO

*Un seminarista puede ejercer funciones de bibliotecario. Para ello se debe ingresar como bibliotecario identificándolo con su RUT.*



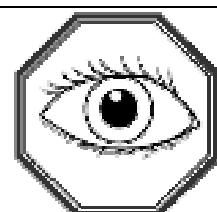
RUT: El RUT debe existir previamente en el sistema.



1.- Botón que indica al sistema que puede ingresar al seminarista como bibliotecario.

## DAR BAJA SEMINARISTA

*Dar de baja a un seminarista significa que este quedará al margen de toda operación, en otras palabras el seminarista se está eliminando del sistema.*



RUT: El RUT no puede estar asociado a un seminarista que ya se encuentra de baja o que posee préstamos activos.

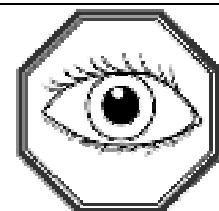


1.- Lista en donde se podrá indicar la razón de porque se está dando de baja al seminarista.

2.- Botón que indica al sistema que puede efectuar la baja.

## DAR ALTA SEMINARISTA

*Dar de alta a un seminarista significa que éste fue dado de baja por alguna razón y este necesita nuevamente hacer uso del sistema.*



RUT: El RUT no puede estar asociado a un seminarista que ya se encuentra de baja o que posee préstamos activos.



- 1.- Botón que indica que se puede efectuar el alta del seminarista que se encuentra de baja.

# BIBLIOTECARIOS



## INGRESAR BIBLIOTECARIO

Para poder ingresar un nuevo bibliotecario, un bibliotecario existente deberá proveer la información que el sistema le solicite para que este pueda tener acceso a todas las funciones que el sistema provee.



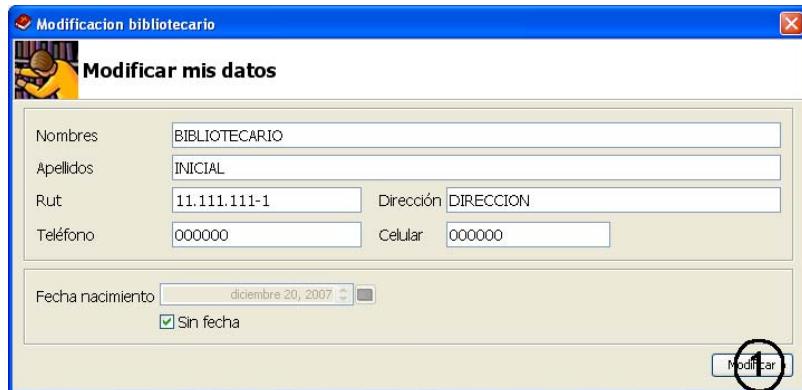
RUT: El sistema no permitirá el ingreso de un Bibliotecario con el mismo RUT que otro Bibliotecario. Además si se digita el RUT de un seminarista este no podrá ser ingresado, deberá hacerlo desde la opción “Usuarios -> Seminaristas -> como Bibliotecario” de modo que se utilice la misma información del seminarista.

A screenshot of a Windows dialog box titled "Ingreso de Bibliotecario". The form contains fields for "Nombres" (Names), "Apellidos" (Last names), "Rut" (Rut), "Dirección" (Address), "Teléfono" (Phone), and "Celular" (Cellular). Below these are fields for "Fecha nacimiento" (Birth date) with a calendar icon and a checkbox for "Sin fecha" (No date). At the bottom right is a button labeled "Finalizar Ingreso" (Finish Input) with a circled number "1" above it.

1.- Botón que indica al sistema que se ingrese al bibliotecario.

## MODIFICAR MIS DATOS

*El bibliotecario tendrá el derecho de modificar sus datos y nadie más podrá hacerlo.*



- 1.- Botón que indica al sistema que puede modificar los datos.

## CAMBIAR CLAVE

Si un bibliotecario desea cambiar su clave, deberá ingresar la clave actual para posteriormente ingresar la nueva clave que deseé.



1.- Clave que actualmente posee el bibliotecario.

2.- Permite ingresar 2 veces la nueva clave, a fin que no se digite erróneamente.

3.- Botón que indica al sistema que ya puede realizar el cambio.

## DAR BAJA BIBLIOTECARIO

*Si se necesita eliminar un bibliotecario del sistema, deberá ser dado de baja e ingresar la razón por la cual es dado de baja.*



- 1.- Lista que contiene las funciones por el cual un bibliotecario podría ser dado de baja.
- 2.- Botón que indica al sistema que ya puede realizar la baja.

## DAR ALTA BIBLIOTECARIO

*Dar de alta a un bibliotecario significa que éste fue dado de baja por alguna razón y este necesita nuevamente hacer uso del sistema.*



1.- Botón que indica al sistema

# USUARIOS ESPECIALES



## INGRESAR USUARIO ESPECIAL

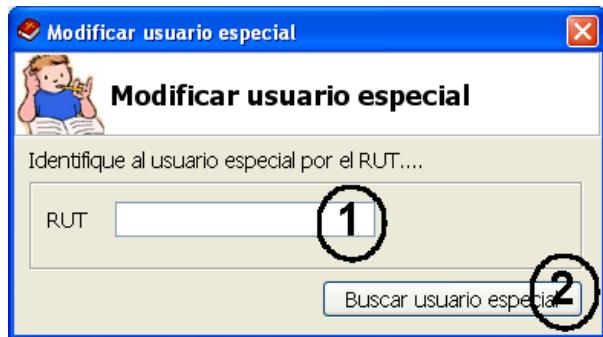
El sistema permite el ingreso de usuarios especiales (usuarios que no pertenecen al seminario) y para aquello se deberá ingresar la información necesaria para que posteriormente puedan efectuar los prestamos del material que necesitan.



1.- Botón que indica al sistema que puede realizar el ingreso del usuario especial

## MODIFICAR USUARIO ESPECIAL 1/2

*Es necesario mantener la información actualizada de un usuario especial, el proceso cuenta de dos pantallas, en la primera se deberá identificar al usuario con el Rut, para luego modificar sus datos.*



- 1.- Permite ingresar el Rut asociado al usuario especial que se desea modificar.
- 2.- Botón que indica al sistema que busque el usuario asociado al Rut ingresado, Para modificar sus datos.

## MODIFICAR USUARIO ESPECIAL 2/2



- 1.- Lista en la que se encuentran los estados que puede tomar un usuario especial.
- 2.- Permite reiniciar la clave asociada al usuario especial.
- 4.- Botón que indica al sistema que puede modificar los datos del usuario.

## DAR BAJA USUARIO ESPECIAL

*El usuario especial, igual puede ser dado de baja con el fin de dejarlo inactivo de toda actividad que permite el sistema.*



Lista en la que se encuentran las razones por las que se puede dar de baja un usuario especial.

Botón que indica al sistema que puede realizar la Baja.

## DAR ALTA USUARIO ESPECIAL

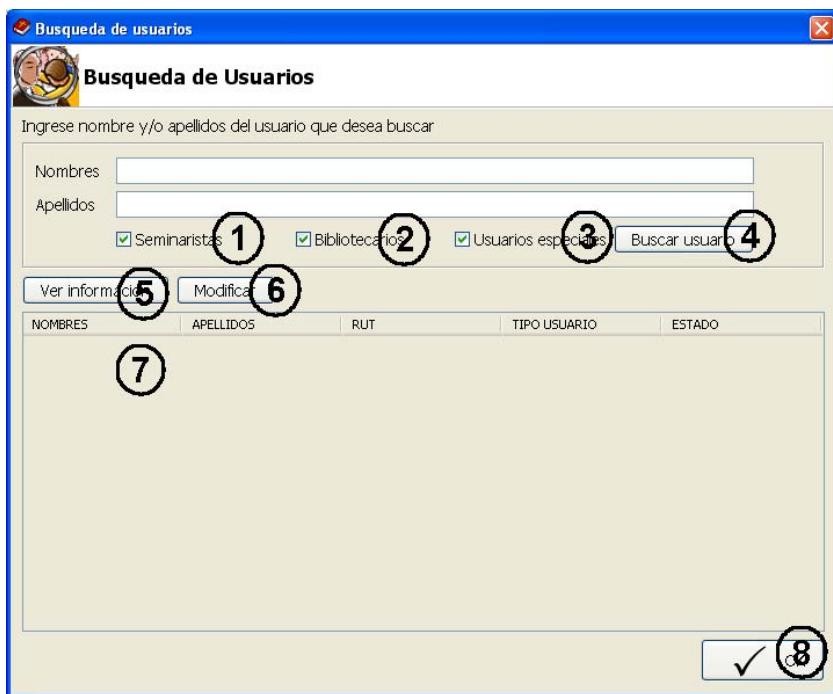
*Al igual que los Seminaristas y Bibliotecarios el usuario especial puede ser dado de alta después de haber sido dado de baja.*



- 1.- Botón que indica al sistema que puede dar de alta al usuario asociado al Rut que se ingresó.

## BUSCAR USUARIO

*Si se necesita la información de un usuario para realizar alguna actividad y no se cuenta con el Rut, este podrá ser buscado por medio del nombre y/o apellido. Luego que el sistema arroje los resultados, estos podrán ser seleccionados con el fin de modificar o ver toda la información asociada al seminarista, bibliotecario y usuario especial.*



- 1.- Opción que puede ser marcada a fin que en la búsqueda se incluyan todos los seminaristas.
- 2.- Opción que puede ser marcada a fin que en la búsqueda se incluyan todos los bibliotecarios.
- 3.- Opción que puede ser marcada a fin que en la búsqueda se incluyan todos los usuarios especiales.
- 4.- Botón que indica al sistema que puede realizar la búsqueda de los usuarios de acuerdo a los criterios que se hayan ingresado.
- 5.- Botón que permite ver la información asociada al usuario que se ha seleccionado con el mouse de la tabla de resultados.

6.- Botón que inicia el proceso de modificación de un usuario seleccionado con el mouse de la tabla de resultados.

7.- Sector donde se ubica la tabla de resultados que arrojará la búsqueda de usuarios.

8.- Botón que indica al sistema que el proceso de búsqueda ha terminado.

## VER INFORMACIÓN SEMINARISTA

Información seminarista

Información de Seminarista

Nombres	CRISTIAN ANTONIO	
Apellidos	ULLOA MERINO	
Rut	9.281.375-4	Dirección LAGO PETROHUE 235
Teléfono	5555555	Celular 09-99999
Año Ingreso	2002	Fecha nacimiento SIN FECHA
Curso	PROPE	
Parroquia origen	SANTA CECILIA / CONCEPCION	
ESTADO	REGULAR	

**OK** 1

- 1.- Botón que indica al sistema que la información del seminarista ya fue utilizada por lo tanto la ventana puede ser cerrada.

## VER INFORMACIÓN BIBLIOTECARIO

Información bibliotecario

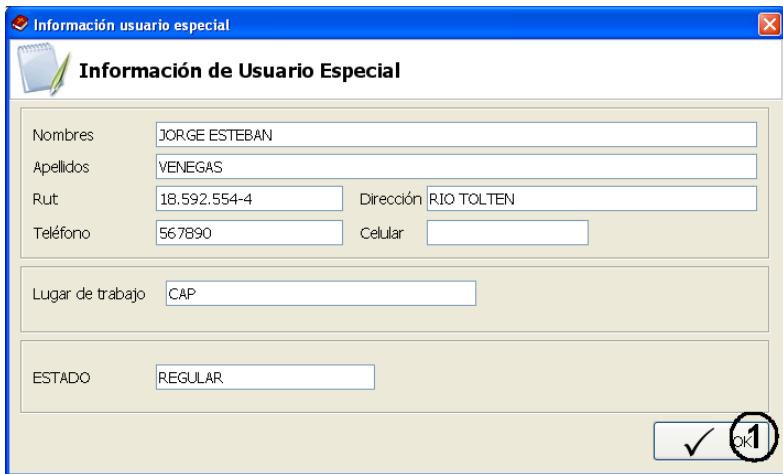
Información de Bibliotecario

Nombres	BIBLIOTECARIO	
Apellidos	INICIAL	
Rut	11.111.111-1	Dirección DIRECCION
Teléfono	000000	Celular 000000
Fecha nacimiento	SIN FECHA	

**OK** 1

- 1.- Botón que indica al sistema que la información del bibliotecario ya fue utilizada por lo tanto la ventana puede ser cerrada.

## VER INFORMACIÓN USUARIO ESPECIAL



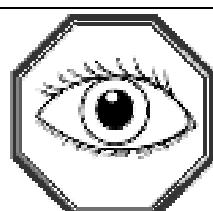
- 1.- Botón que indica al sistema que la información del usuario especial ya fue utilizada por lo tanto la ventana puede ser cerrada.

# PRESTAMOS

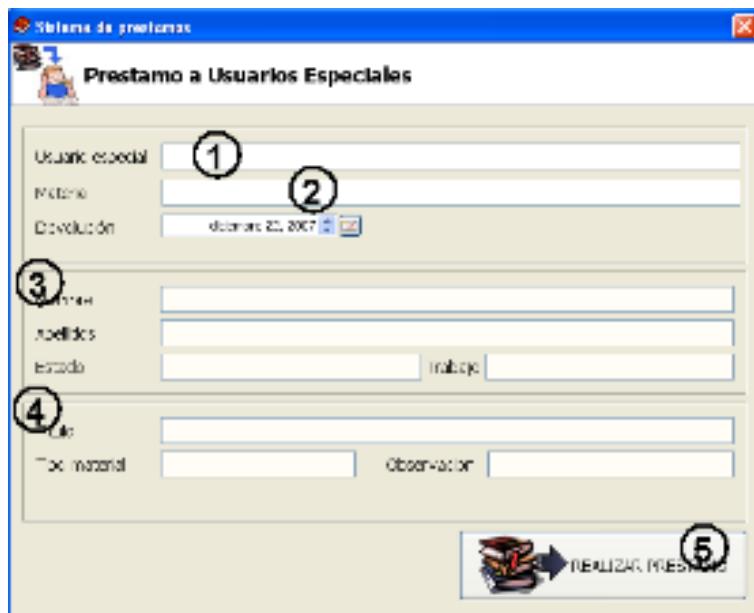


## INGRESAR PRÉSTAMO USUARIO ESPECIAL

Cuando un usuario especial acude al seminario para solicitar un libro, se debe acceder a la sección especial de préstamos, en ella podrá identificar al usuario especial y luego el material.



Devolución: La fecha de devolución de un material solicitado por un usuario especial, siempre será de un día de plazo, pero este puede ser cambiado según el criterio del bibliotecario.



1.- Permite identificar al usuario especial mediante el Rut.

2.- Permite identificar el material mediante el código de registro, este puede ser tecleado o leído por el lector de códigos de barra.

3.- Marco en donde aparecerá la información referente al usuario especial.

4.- Marco donde aparecerá la información del material.

5.- Botón que le indica al sistema que realice el préstamo.

## DEVOLUCIÓN DE USUARIO ESPECIAL

Luego de que un usuario especial realice un préstamo, deberá devolverlo, para aquello el bibliotecario deberá ingresar a la sección de préstamos y luego indicar la devolución.



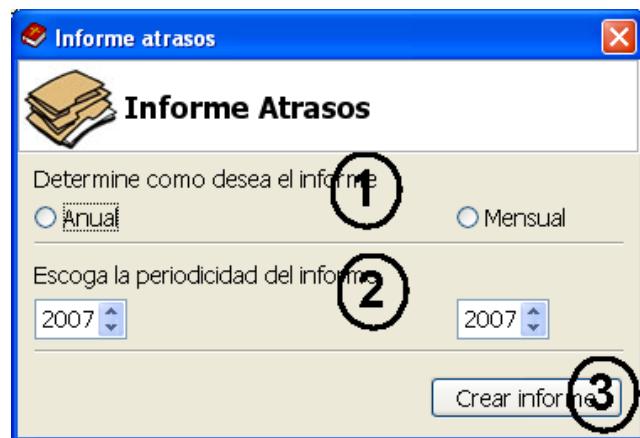
- 1.- Permite identificar el material que se está devolviendo, puede ser ingresado por teclado o leído por el lector de códigos de barra.
- 2.- Botón que le indica al sistema que puede comenzar con la devolución.

# INFORMES



Para la generación de informes, bastará con escoger desde el menú el informe que desea generar. Luego si es pertinente el sistema le solicitará algunos datos como rangos de fecha etc. Para luego apreciar en pantalla la información que solicitó, los informes podrán ser generados en archivos PDF para luego ser impresos. A continuación se describe un ejemplo del informe "Atrasos de seminaristas".

## ATRASOS SEMINARISTAS 1/2



- 1.- Se indica si el informe debe detallado por año o por mes.
- 2.- Se escoge el rango de años para que se detalle el informe.
- 3.- Se indica al sistema que puede crear el informe.

## ATRASOS SEMINARISTAS 2/2



- 1.- La fecha en que fue generado el informe.
- 2.- Tabla en la que aparecen los resultados de la petición.
- 3.- Botón que permite pasar el informe a pdf para su posterior impresión.
- 4.- Botón que indica al sistema que ya se ha visualizado la información y que puede cerrar la ventana.

**ANEXO 2**

**CASOS DE USO Y DIAGRAMAS CONCEPTUALES DE  
TODO EL SISTEMA**

## **INTRODUCCIÓN**

En este documento se da a conocer todos los casos de uso del sistema bibliotecario perteneciente al proyecto Sistema Orientado a Objetos para Biblioteca del Seminario Concepción. Los casos de uso son separados por los seis paquetes que contiene el sistema, los que además contienen su diagrama conceptual.

## DIAGRAMA DE PAQUETES

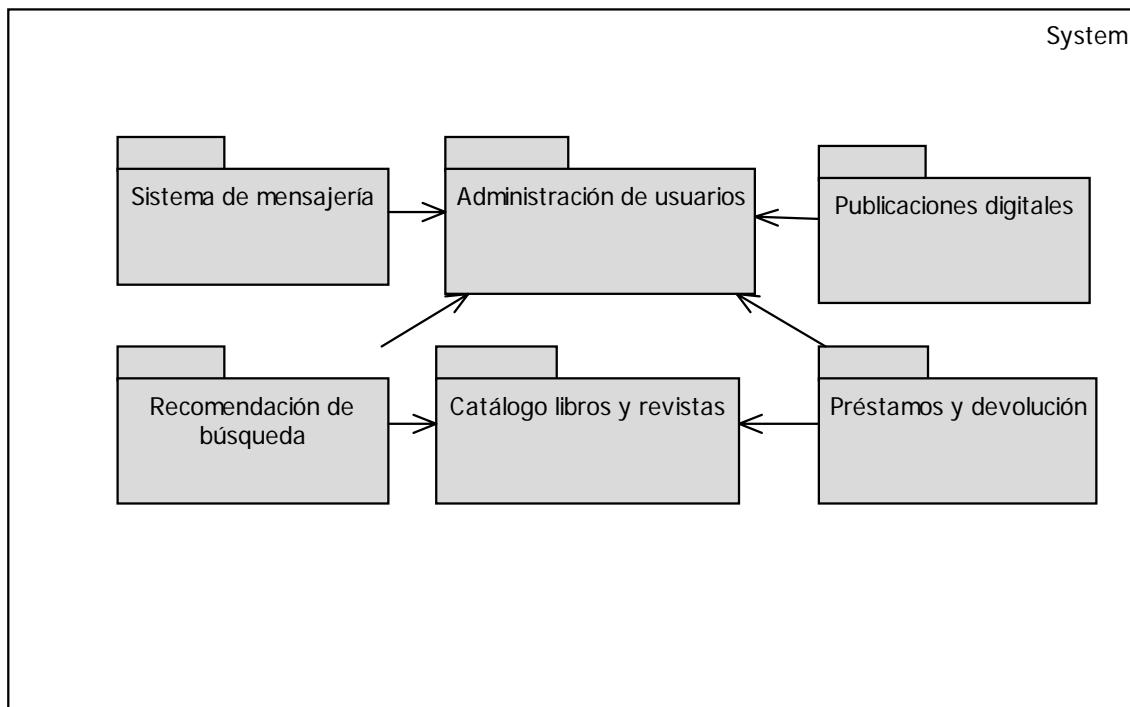


Figura 47: Diagrama de paquetes.

Figura en la que se muestra como el sistema está dividido en agrupaciones lógicas denominadas paquetes, estos son dependientes unos de otros, exceptuando “Catalogo libros y revistas” y “Administración de usuarios” que se pueden trabajar por separado al ser totalmente independientes de los demás paquetes.

# PAQUETE: CATÁLOGO LIBROS Y REVISTAS.

## DIAGRAMA DE CASOS DE USO

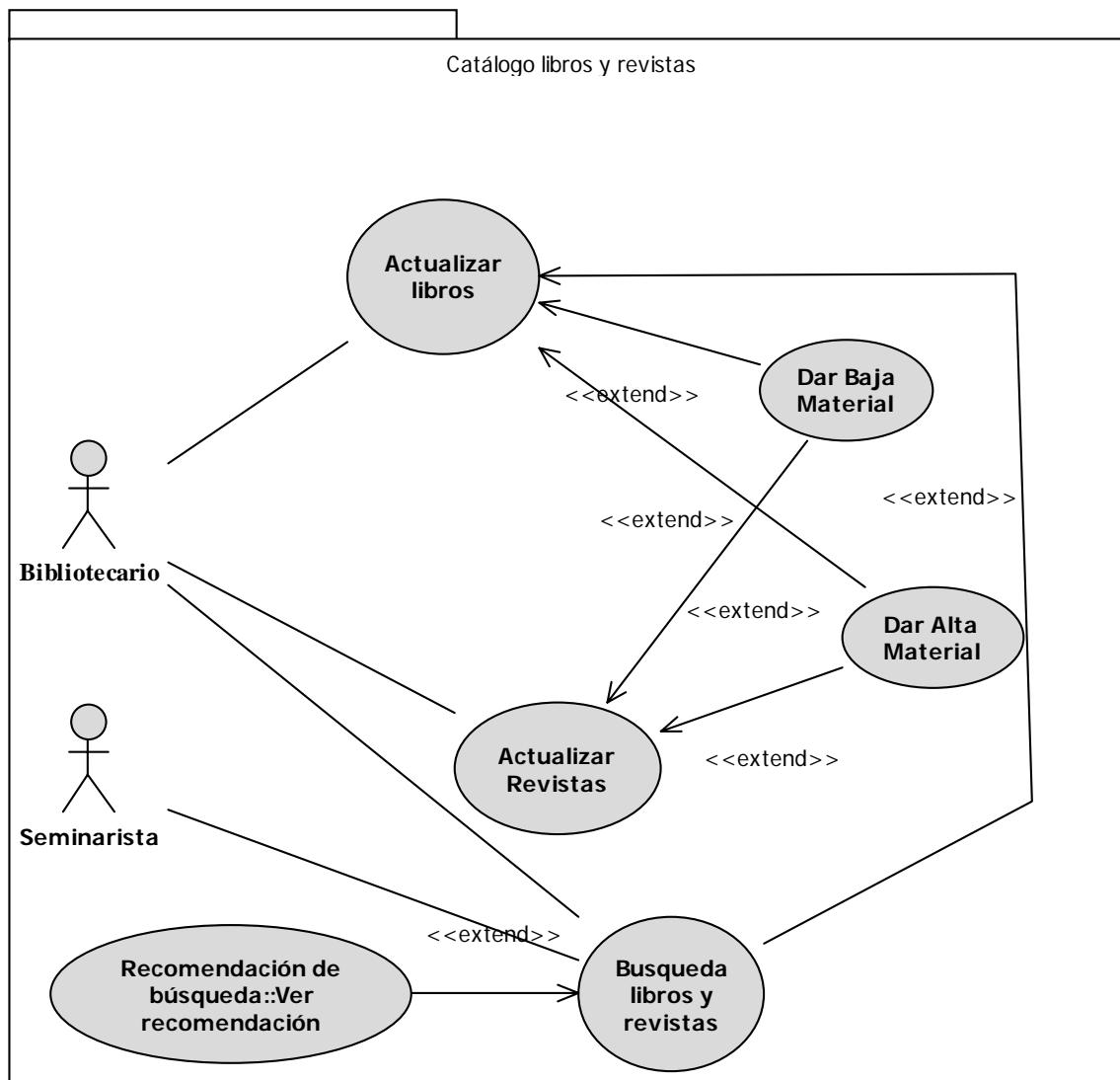


Figura 48: Paquete catálogo libros y revistas.

Figura en la que se muestra el diagrama de casos de uso perteneciente al paquete de catálogo de libros y revistas interactuado por el bibliotecario y seminaristas. En el diagrama se puede observar el caso de uso ACTUALIZAR LIBROS en el que se tiene la opción de ingresar un libro, modificar un libro y además de extender a los casos de uso DAR BAJA MATERIAL y DAR ALTA MATERIAL que a su vez son extendidos por ACTUALIZAR REVISTAS el que tiene la opción de Ingresar y Modificar revista. Para realizar la búsqueda del material existe BÚSQUEDA

**LIBROS Y REVISTAS**, el que extiende el caso de uso VER RECOMENDACIÓN del paquete recomendación de búsqueda.

## **CASO DE USO: ACTUALIZAR LIBROS.**

**Actores** : Bibliotecario.

**Propósito** : Mantener actualizados los libros de la biblioteca.

**Tipo** : Primario y esencial.

**Referencias cruzadas** : Requerimientos R1, R2, R3.

**Descripción** :

El bibliotecario, cuando desea actualizar un libro debe poseer una cierta cantidad de información que el sistema le solicitará, esta será aún mayor al momento de hacer ingreso de material, el bibliotecario contará con las siguientes opciones:

- Ingresar Libro.
- Modificar Libro.
- Dar Baja Libro.
- Dar Alta Libro.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario selecciona la opción que desea para mantener actualizada la información de un libro o revista.	
	Valida la opción elegida por el bibliotecario y despliega opción correspondiente.
	Deriva a la sección o caso de uso extendido según corresponda a la elección del

	<p>bibliotecario.</p> <ul style="list-style-type: none"><li>- <i>Ingresar libro:</i> Deriva a la sección ingresar libro.</li><li>- <i>Modificar libro:</i> Deriva a la sección modificar libro.</li><li>- <i>Dar Baja material:</i> Deriva al caso de uso extendido dar baja material.</li><li>- <i>Dar Alta material:</i> Deriva al caso de uso extendido dar alta material.</li></ul>
--	---

## CURSOS ALTERNOS

## **SECCIÓN: INGRESAR LIBRO.**

**Descripción** :

Si existe un libro que no está ingresado en el sistema, el bibliotecario debe tomar la información necesaria para poder ingresarla y dejar disponible el material a los usuarios de la biblioteca del seminario. A continuación, se nombran los datos que pertenecen a un Libro.

**DATOS:** Título / ISBN / Cantidad de páginas / Volumen / Tomo / Tipo / Registro anterior / Autor principal / Clasificación / Edición / año publicación / Editorial / Ediciones / Serie / País publicación / Ciudad publicación / Largo / Ancho / Notas / Materias /Autores secundarios.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
	Muestra en pantalla el formulario para ingresar los demás datos.
Digita en el formulario: Título / ISBN / Cantidad de páginas / Volumen / Tomo / Registro anterior / Autor principal / Clasificación / Edición / año publicación / País publicación / Ciudad publicación / Largo / Ancho / Notas. Y selecciona desde las listas correspondientes: Tipo / Serie / Editorial / Ediciones.	
Ingresar Materias y autores secundarios	Verifica que la clasificación ingresada no exista y muestra formulario con lista de materias y autores.

	Ingresá el libro en el sistema, muestra en pantalla número de registro y da la opción de imprimir etiqueta de códigos de barra perteneciente al libro
Selecciona opción para imprimir etiqueta de códigos de barra.	
	Ordena a la impresora que imprima la etiqueta de códigos de barra.

## CURSOS ALTERNOS.

**Ítem 2:** Ingresa un título de libro que ya existe en el sistema. Da a conocer al bibliotecario la existencia de otro libro con el mismo título, él decide si ingresar una copia, ocupar la información del libro que existe para el nuevo libro o ingresar el libro con el mismo título.

**Ítem 2:** En las listas no existe el valor que se desea ingresar para: Tipo / Serie / Editorial / Ediciones. El usuario debe ingresar al sistema el valor para posteriormente escogerlo.

**Ítem 4:** La clasificación que el usuario ingresó ya existe en el sistema. Informa al usuario y no permite el ingreso del libro.

## **SECCIÓN: MODIFICAR LIBRO.**

**Descripción** :

Si por equivocación, se ingresan datos erróneos de un libro, o se desea ingresar nueva información perteneciente al material, se deberá identificar el libro con el código de clasificación o código de registro, para que el sistema muestre un formulario en el cual se podrá indicar las modificaciones del libro.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
Ingresa el código de clasificación o se pistolea el código de barras para asociarlo al código de registro.	
	Busca el libro y muestra formulario en pantalla.
Modifica los datos que estime pertinentes y envía al sistema (se pueden modificar todos los datos).	
	Registra el cambio.

## **CURSOS ALTERNOS**

**Ítem 2:** El código de clasificación o código de registro ingresado no existe en el sistema. Se indica error y muestra nuevamente opción de ingreso de los datos.

## **CASO DE USO: DAR BAJA MATERIAL.**

**Actores** : Bibliotecario.

**Propósito** : Inhabilitar un libro o revista del sistema.

**Tipo** : Primario y esencial.

**Referencias cruzadas** : Requerimientos R2, R5.

**Descripción** :

Si un libro o revista se estropea, extravía o simplemente es necesario eliminarlo del sistema, podrá ser dado de baja. El bibliotecario deberá indicar el código de registro del material para luego darlo de baja.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario ingresa el código de registro (Manual o pistoleando el código de barras) del libro o revista.	
	Busca el material en la base de datos y muestra formulario en pantalla para indicar razones de la baja.
Indica razones de baja y envía el formulario al sistema.	
	Registra la baja.

## **CURSOS ALTERNOS**

**Ítem 1:** El bibliotecario ingresa un código inválido. Se indica error y muestra nuevamente opción de ingreso del código.

**Ítem 2:** El código ingresado no existe en el sistema. Se indica error y muestra nuevamente opción de ingreso del código.

## **CASO DE USO: DAR ALTA MATERIAL.**

<b>Actores</b>	: Bibliotecario.
<b>Propósito</b>	: Dejar nuevamente activo un libro o revista en el sistema.
<b>Tipo</b>	: Primario y esencial.
<b>Referencias cruzadas</b>	: Requerimientos R2, R5.

**Descripción** :

Un libro o revista después de ser dado de baja, puede nuevamente quedar activo en el sistema, para ello el bibliotecario deberá ingresar el código de registro del material, con el cual el libro o revista automáticamente volverá a estar a disposición de los usuarios.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario ingresa el código de registro (Manual o pistoleando el código de barras) del libro o revista que se desea dar alta.	
	Busca el material en la base de datos y vuelve a preguntar si desea dar alta.
Ingresa respuesta positiva a dar el alta al libro.	
	Registra el alta y deja el libro activo.

## **CURSOS ALTERNOS**

**Ítem 1:** El bibliotecario ingresa un código inválido. Se indica error y muestra nuevamente opción de ingreso del código.

**Ítem 2:** El código ingresado no existe dentro del material que se encuentra de baja. Se indica error y muestra nuevamente opción de ingreso del código.

**Ítem 3:** Ingresa respuesta negativa. El sistema cancela el proceso.

## **CASO DE USO: ACTUALIZAR REVISTAS.**

**Actores** : Bibliotecario.

**Propósito** : Mantener actualizado el material de la Biblioteca.

**Tipo** : Primario y esencial.

**Referencias cruzadas** : Requerimientos R4, R5, R6.

**Descripción** :

El bibliotecario, cuando desea actualizar una revista debe manejar una cierta cantidad de información que el sistema le solicitará, dependiendo de la acción que el bibliotecario seleccione. Las opciones se listan a continuación.

- Ingresar Revista.
- Modificar Revista.
- Dar Baja Revista.
- Dar Alta Revista.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario selecciona la opción que desea para mantener actualizada la información de una revista.	
	Valida la opción elegida por el bibliotecario y despliega opción correspondiente.
	Deriva a la sección o caso de uso extendido según corresponda a la elección del

	<p>bibliotecario.</p> <ul style="list-style-type: none"><li>- <i>Ingresar revista:</i> Deriva a la sección ingresar revista.</li><li>- <i>Modificar revista:</i> Deriva a la sección modificar revista.</li><li>- <i>Dar Baja material:</i> Deriva al caso de uso extendido dar baja material.</li><li>- <i>Dar Alta material:</i> Deriva al caso de uso extendido dar alta material.</li></ul>
--	---

## CURSOS ALTERNOS

## **SECCIÓN: INGRESAR REVISTA.**

**Descripción :**

Si existe una revista que no está ingresada en el sistema, el bibliotecario debe tomar la información necesaria para ingresarla y dejarla disponible a los usuarios de la biblioteca del seminario. El bibliotecario deberá seleccionar el título de la revista a la cual desea ingresar un nuevo número, en caso que la revista que desea seleccionar no exista en el sistema, deberá ingresar algunos datos adicionales.

**DATOS REVISTA:** Periodicidad (Fecha de edición y caducación) / número / volumen / año / contenido.

**DATOS ADICIONALES NUEVA REVISTA:** Nombre / fecha fundación / Periodicidad (cantidad de días) / ISSN.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
Selecciona nombre de la revista a la cual desea ingresar un nuevo número.	
	Muestra en pantalla el formulario para ingresar los demás datos.
Digita en el formulario: Periodicidad / número / volumen / año.	
	Verifica que los datos correspondan al formato que se pide e ingresa los datos al sistema con un número de registro.
Selecciona el contenido de la revista	

	Vincula el contenido a la revista ingresada.
	Ingresa la revista al sistema y muestra en pantalla el número de registro dando la opción de imprimir el código de barras.
Selecciona opción de imprimir el código de barras.	
	Ordena a la impresora que imprima la etiqueta de códigos de barra.

## CURSOS ALTERNOS.

**Ítem 1:** La revista que desea seleccionar no existe. En ese instante debe ingresar los “datos adicionales nueva revista”, para posteriormente seleccionar la revista en el ítem 1.

**Ítem 4:** Ingreso incorrecto según formato. Se indica error y no permite crear la revista hasta que ingrese los valores correctos.

**Ítem 5:** El contenido de la revista no existe en la lista. Debe ingresarlo para posteriormente ser elegido.

**Ítem 5:** La revista posee más de un contenido. El bibliotecario debe indicar al sistema el ingreso de otro contenido para posteriormente pasar del ítem 6 al ítem 5.

## **SECCIÓN: MODIFICAR REVISTA.**

**Descripción :**

Si por equivocación, se ingresan datos erróneos de una revista, o se desea ingresar nueva información perteneciente al material, se deberá identificar la revista con el número de registro o el nombre de la revista. En caso de ingresar el número de registro de una revista se podrán modificar los datos propios del número, en el caso de seleccionar el nombre de la revista, se modificarán los datos generales. A continuación se nombran los datos que se pueden modificar en cada caso:

**DATOS NUMERO:** Periodicidad (Fecha de edición y caducación) / número / volumen / año / contenido.

**DATOS REVISTA:** fecha fundación / Periodicidad (cantidad de días) / ISSN.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
Ingresa el código de registro o el título de la revista.	
	Busca la revista y muestra formulario.
Modifica los datos pertinentes y envía al sistema.	
	Registra el cambio.

## **CURSOS ALTERNOS**

**Ítem 2:** El código de registro o el nombre ingresado no existe. Indica error y muestra nuevamente opción de ingreso de datos.

## **CASO DE USO: BÚSQUEDA LIBROS Y REVISTAS.**

<b>Actores</b>	: Bibliotecario, Seminarista.
<b>Propósito</b>	: Encontrar material por medio de criterios de búsqueda.
<b>Tipo</b>	: Primario y esencial.
<b>Referencias cruzadas</b>	: Requerimiento R18.

**Descripción** :

Si un seminarista o bibliotecario necesita realizar la búsqueda de material, primero debe especificar qué tipo de material es, ya sea libro o revista, luego debe seleccionar el criterio de búsqueda que estime conveniente. Los criterios de búsqueda se mencionan a continuación:

**PARA LIBRO:** Título / Autor / Materia / Editorial / Serie / Ediciones / ISBN / Registro material.

**PARA REVISTA:** Nombre / Contenido / ISSN / Fechas / Número de revista.

Luego de escoger el criterio de búsqueda, se debe digitar la palabra que se desea buscar, para que el sistema arroje los resultados encontrados con la información más importante del libro.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario o seminarista indica al sistema si lo que se desea buscar es un libro o revista.	
	Muestra los criterios de búsqueda asociados al material (Libro o revista)
Selecciona el criterio de búsqueda, digita	

palabras a buscar y envía al sistema.	
	Según el criterio escogido y la palabra digitada, realiza la búsqueda.
	Muestra en pantalla las coincidencias de la búsqueda.

## CURSOS ALTERNOS

**Ítem 3:** Envía formulario sin escribir nada. Indica error.

**Ítem 5:** No se encuentran coincidencias, se indica que no hubo éxito en la búsqueda.

# MODELO CONCEPTUAL

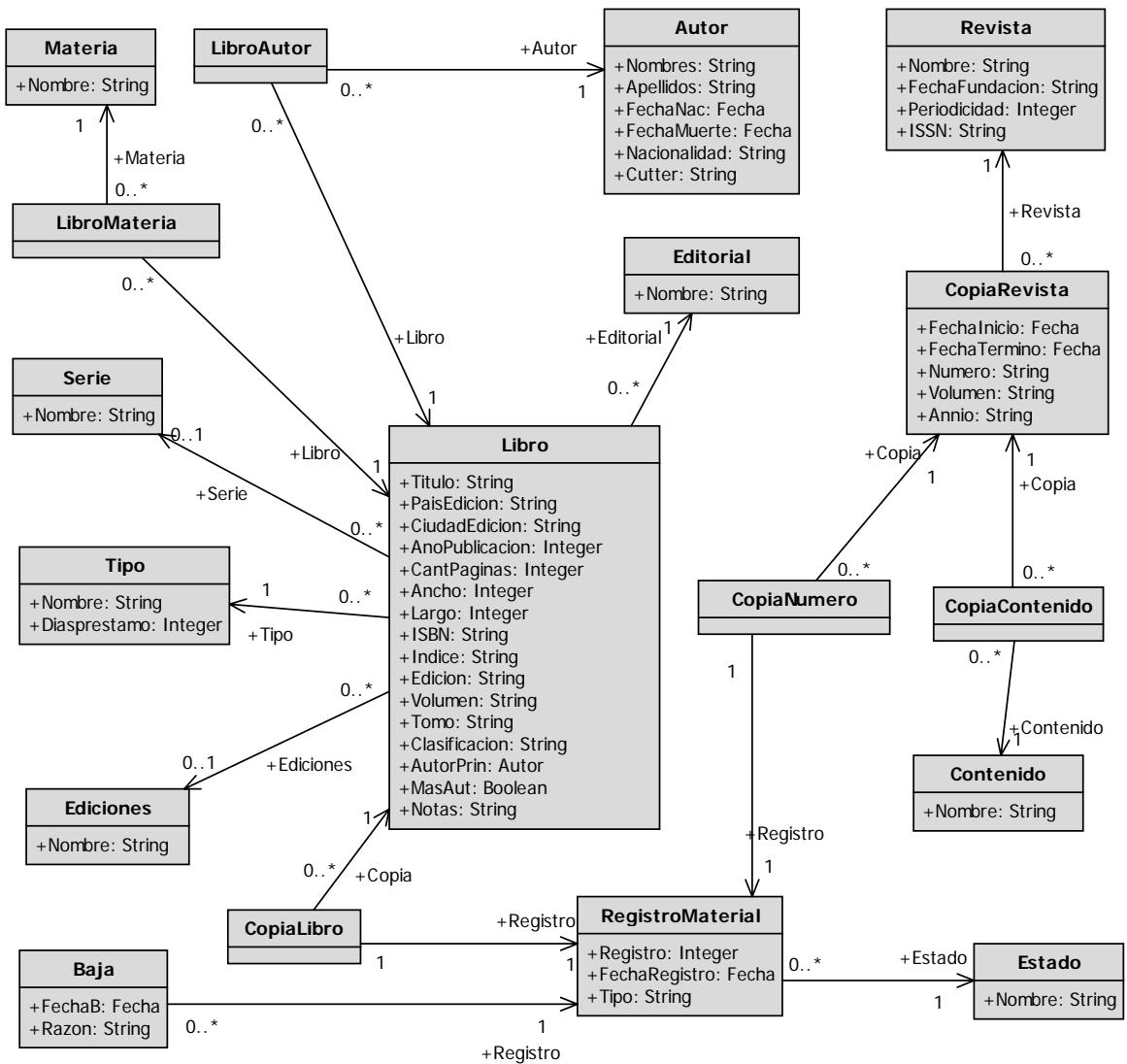


Figura 49: Modelo conceptual paquete Catálogo libros y revistas.

## PAQUETE: ADMINISTRACIÓN DE USUARIOS.

### DIAGRAMA DE CASOS DE USO

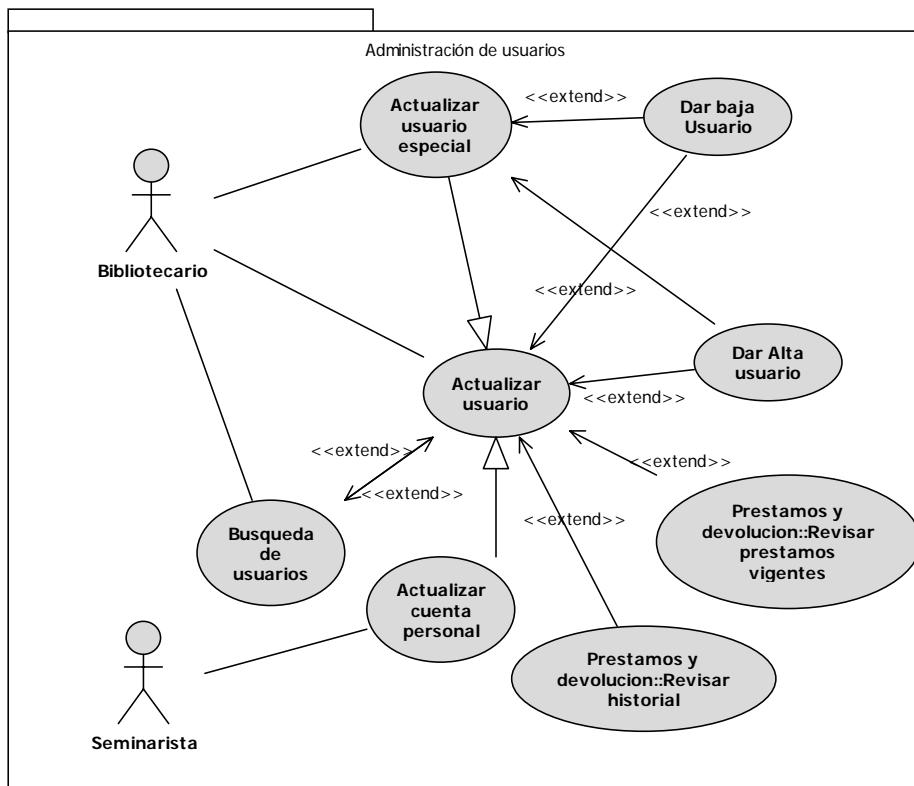


Figura 50: Paquete Administración de usuarios.

Figura en la que se muestra el diagrama de casos de uso perteneciente al paquete de administración de usuarios interactuado por el bibliotecario y seminaristas. En el diagrama se puede observar el caso de uso ACTUALIZAR USUARIO en el que se tiene la opción de ingresar un usuario (bibliotecario o seminarista), modificar un usuario (bibliotecario o seminarista) y extiende DAR BAJA USUARIO y DAR ALTA USUARIO a estos también los extiende ACTUALIZAR USUARIO ESPECIAL el que tiene la opción de ingresar usuario externo y modificar usuario externo. También se observa ACTUALIZAR CUENTA PERSONAL en a que un seminarista puede modificar sus datos personales. Para realizar la búsqueda de un usuario existe BÚSQUEDA DE USUARIOS. En el diagrama se puede apreciar que el caso de uso Actualizar usuario extiende a los casos de uso REVISAR HISTORIAL Y REVISAR PRESTAMOS VIGENTES pertenecientes al paquete de préstamos y devolución.

## CASO DE USO: ACTUALIZAR USUARIO.

<b>Actores</b>	: Bibliotecario.
<b>Propósito</b>	: Mantener actualizada la información de los usuarios.
<b>Tipo</b>	: Primario y esencial.
<b>Referencias cruzadas</b>	: Requerimientos R7, R8, R9, R10, R11, R12.
<b>Descripción</b>	: Para poder actualizar una cuenta de usuario, el bibliotecario deberá en primera instancia elegir qué tipo de usuario desea actualizar, ya sea seminarista o bibliotecario para luego tener la opción de crear, modificar, dar de baja y dar de alta al usuario que desee.

## CURSO NORMAL DE LOS EVENTOS

Acción del actor	Acción del sistema
El caso de uso comienza cuando el bibliotecario selecciona la opción que desea para actualizar la información relacionada a un usuario.	
	Valida la opción elegida por el bibliotecario y despliega opciones existentes.
	Deriva a la sección o caso de uso extendido según corresponda a la elección del bibliotecario.  - <i>Ingresar usuario:</i> Deriva a la sección ingresar usuario.  - <i>Modificar Usuario:</i> Deriva a la

	<p>sección modificar usuario.</p> <ul style="list-style-type: none"><li>- <i>Dar Baja Usuario:</i> Deriva al caso de uso extendido dar baja usuario.</li><li>- <i>Dar Alta Usuario:</i> Deriva al caso de uso extendido dar alta usuario.</li></ul>
--	---

## CURSOS ALTERNOS.

## **SECCIÓN: INGRESAR USUARIO.**

**Descripción :**

Para crear un usuario, el bibliotecario debe en primera instancia indicar el tipo de usuario que desea ingresar, estos pueden ser seminarista y bibliotecario, además debe tener en su poder una serie de datos que deben ser ingresados en el sistema para crear la cuenta, la información asociada a cada usuario se lista a continuación.

**PARA SEMINARISTA:** Nombres / Apellidos / RUT / Fecha de Nacimiento / Teléfono / Celular / Parroquia / Diócesis / Año de ingreso / Curso

**PARA BIBLIOTECARIO:** Nombres / Apellidos / RUT / Fecha de Nacimiento / Teléfono

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
Indica al sistema si lo que se desea ingresar es un seminarista o bibliotecario.	
	Muestra en pantalla el formulario correspondiente al tipo de usuario que se desea ingresar.
Ingresa los datos correspondientes al usuario a ingresar.	
	Valida que el RUT ingresado no exista en el sistema.
	Crea la nueva cuenta y la deja activa para su uso.

## **CURSOS ALTERNOS.**

**Ítem 3:** El RUT ya existe en el sistema. Indica que el usuario que desea ingresar ya está en el sistema.

## **SECCIÓN: MODIFICAR USUARIO.**

**Descripción** :

Si un usuario de la biblioteca cambia un dato, el bibliotecario puede modificar la información, además si el bibliotecario por alguna razón desea cambiar el estado de un seminarista puede hacerlo de la siguiente manera:

De INACTIVO a REGULAR

De REGULAR a INACTIVO

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
Ingresa el RUT del usuario al cual desea modificar los datos.	
	Busca al usuario y muestra formulario en pantalla.
Modifica los datos pertinentes y envía (Puede modificar todos los datos a excepción del RUT).	
	Registra el cambio.

## **CURSOS ALTERNOS**

**Ítem 1:** El bibliotecario ingresa un RUT inválido. Se indica error y muestra nuevamente opción de ingreso de RUT.

**Ítem 2:** El RUT ingresado no existe en el sistema. Se indica error y muestra nuevamente opción de ingreso de RUT.

## **CASO DE USO: DAR BAJA USUARIO.**

**Actores** : Bibliotecario.

**Propósito** : Dejar inactivo a un usuario del sistema de todo acto.

**Tipo** : Primario y esencial.

**Referencias cruzadas** : Requerimientos R9, R12, R16.

**Descripción** :

Si por alguna razón un usuario del sistema, ya sea seminarista, bibliotecario o un usuario especial, debe perder su condición de usuario, el bibliotecario deberá ingresar que tipo de usuario es el que se desea dar de baja, luego ingresar el RUT, y por último indicar cual es la razón de la baja.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario ingresa que tipo de usuario desea dar baja (seminarista, bibliotecario, usuario especial) y luego el RUT correspondiente al usuario.	
	Busca al usuario y muestra formulario en pantalla para ingresar las razones de baja.
Indica razones de baja y envía el formulario.	
	Registra la baja.

## **CURSOS ALTERNOS**

**Ítem 1:** El bibliotecario ingresa un RUT inválido. Se indica error y muestra nuevamente opción de ingreso de RUT.

**Ítem 1:** El bibliotecario ingresa el RUT asociado a su cuenta. Indica error y no permite la baja.

**Ítem 2:** El RUT ingresado no existe en el sistema. Se indica error y muestra nuevamente opción de ingreso de RUT.

**Ítem 2:** El RUT ingresado ya se encuentra de baja. Se indica error y cancela el proceso.

## **CASO DE USO: DAR ALTA USUARIO.**

**Actores** : Bibliotecario.

**Propósito** : Dejar nuevamente activo a un usuario en el sistema.

**Tipo** : Primario y esencial.

**Referencias cruzadas** : Requerimientos R9, R12, R16.

**Descripción** :

Si un usuario, ya sea seminarista, bibliotecario o un usuario especial, después de ser dado de baja necesita volver a utilizar el sistema, un bibliotecario puede darlo de alta para volver a realizar las actividades que requiera. El bibliotecario deberá indicar el tipo de usuario que desea dar de alta y el RUT del usuario.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario ingresa el tipo de usuario que desea dar de alta (seminarista, bibliotecario, usuario especial) y posteriormente el RUT correspondiente al usuario.	
	Busca al usuario y vuelve a preguntar si desea dar alta.
Ingresar respuesta positiva a dar el alta al usuario.	
	Registra el alta.

## **CURSOS ALTERNOS**

**Ítem 1:** El bibliotecario ingresa un RUT inválido. Se indica error y muestra nuevamente opción de ingreso de RUT.

**Ítem 2:** El RUT ingresado no existe en el sistema. Se indica error y muestra nuevamente opción de ingreso de RUT.

**Ítem 3:** Ingresa respuesta negativa. El sistema cancela el proceso.

## **CASO DE USO: ACTUALIZAR USUARIO ESPECIAL.**

<b>Actores</b>	: Bibliotecario.
<b>Propósito</b>	: Mantener actualizados a usuarios que no pertenecen al seminario.
<b>Tipo</b>	: Secundario y esencial.
<b>Referencias cruzadas</b>	: Requerimiento R16.

**Descripción** :

Para mantener actualizada la información perteneciente a usuarios que no pertenecen al seminario, el bibliotecario tendrá las siguientes opciones:

- Ingresar usuario externo.
- Modificar usuario externo.

### **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario selecciona la opción que desea para actualizar la información relacionada a los usuarios externos.	
	Valida la opción elegida por el bibliotecario y despliega opciones.
	Deriva a la sección o caso de uso extendido según corresponda a la elección del bibliotecario.  - <i>Ingresar usuario externo</i> : Deriva a la sección ingresar usuario externo.  - <i>Modificar usuario externo</i> : Deriva a

	<p>la sección modificar usuario externo.</p> <ul style="list-style-type: none"><li>- <i>Dar Baja usuario:</i> Deriva al caso de uso extendido dar baja usuario.</li><li>- <i>Dar Alta usuario:</i> Deriva al caso de uso extendido dar alta usuario.</li></ul>
--	--

## CURSOS ALTERNOS

## **SECCIÓN: INGRESAR USUARIO EXTERNO.**

**Descripción** :

Si una persona ajena al seminario desea hacer uso del sistema bibliotecario, deberá poseer una cuenta de “usuario externo” en la que el bibliotecario deberá ingresar los siguientes datos:

**DATOS USUARIOS ESPECIALES:** Nombres / Apellidos / RUT / Dirección / Teléfono /  
Lugar de trabajo o estudio

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
Ingresa los datos del usuario externo, estos deben ser verificados con la cedula de identidad.	
	Valida que los datos correspondan al formato que se pide.
	Crea la nueva cuenta y la deja activa para su uso.

## **CURSOS ALTERNOS.**

**Ítem 2:** Ingreso incorrecto según formato. Se indica error y no permite crear la cuenta de usuario externo.

## **SECCIÓN: MODIFICAR USUARIO EXTERNO.**

**Descripción :**

Un usuario externo, es un usuario esporádico que necesita del material existente en la biblioteca del seminario, por lo cual cada vez que un usuario externo acude al seminario es importante mantener los datos actualizados y es necesario verificar que la información continua vigente, en caso de que un usuario cambie algún dato se debe corregir. El bibliotecario deberá ingresar el RUT del usuario externo del que desea realizar un cambio, para que luego el sistema le muestre en pantalla un formulario para ingresar los cambios.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
Ingresa el RUT del usuario perteneciente a la cuenta que desea modificar.	
	Busca al usuario externo en el sistema y muestra formulario en pantalla.
Modifica los datos pertinentes y envía al sistema.	
	Registra el cambio.

## **CURSOS ALTERNOS**

**Ítem 1:** El bibliotecario ingresa un RUT inválido. Se indica error y muestra nuevamente opción de ingreso de RUT.

**Ítem 2:** El RUT ingresado no existe en el sistema. Se indica error y muestra nuevamente opción de ingreso de RUT.

## **CASO DE USO: ACTUALIZAR CUENTA PERSONAL.**

<b>Actores</b>	: Seminarista.
<b>Propósito</b>	: Actualizar la información personal de un seminarista.
<b>Tipo</b>	: Secundario y esencial.
<b>Referencias cruzadas</b>	: Requerimiento R21.
<b>Descripción</b>	:
	Si un seminarista desea actualizar algún dato de su cuenta, puede hacerlo ingresando al sistema, en el cual tendrá la opción de modificar solo sus datos.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza una vez que el seminarista está en su cuenta y necesita modificar su información, este modifica los datos que estime conveniente a excepción del RUT.	
	Comprueba que los datos cumplan con el tipo de dato correspondiente.
	Registra el cambio.

## **CURSOS ALTERNOS**

**Ítem 2:** Ingreso incorrecto según tipo de dato. Se indica error y no permite modificación de la cuenta.

## **CASO DE USO: BÚSQUEDA DE USUARIOS.**

<b>Actores</b>	: Bibliotecario
<b>Propósito</b>	: Buscar a un usuario en el sistema.
<b>Tipo</b>	: Secundario y esencial.
<b>Referencias cruzadas</b>	: Requerimientos R8, R9, R11, R12

**Descripción** :

El bibliotecario, si no cuenta con el RUT de un usuario y le es necesario visualizar los datos, modificar o dar de baja tendrá un sistema de búsqueda según los siguientes criterios.

- Apellido.
- Nombre.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
Ingresa Apellido o Nombre del usuario a buscar.	
	Busca coincidencias de apellidos o nombre en la base de datos.
	Muestra en pantalla las coincidencias encontradas.
Selecciona usuario a visualizar.	

## **CURSOS ALTERNOS**

**Ítem 3:** No se encontraron coincidencias, se indica y vuelve a ítem 1.

# MODELO CONCEPTUAL

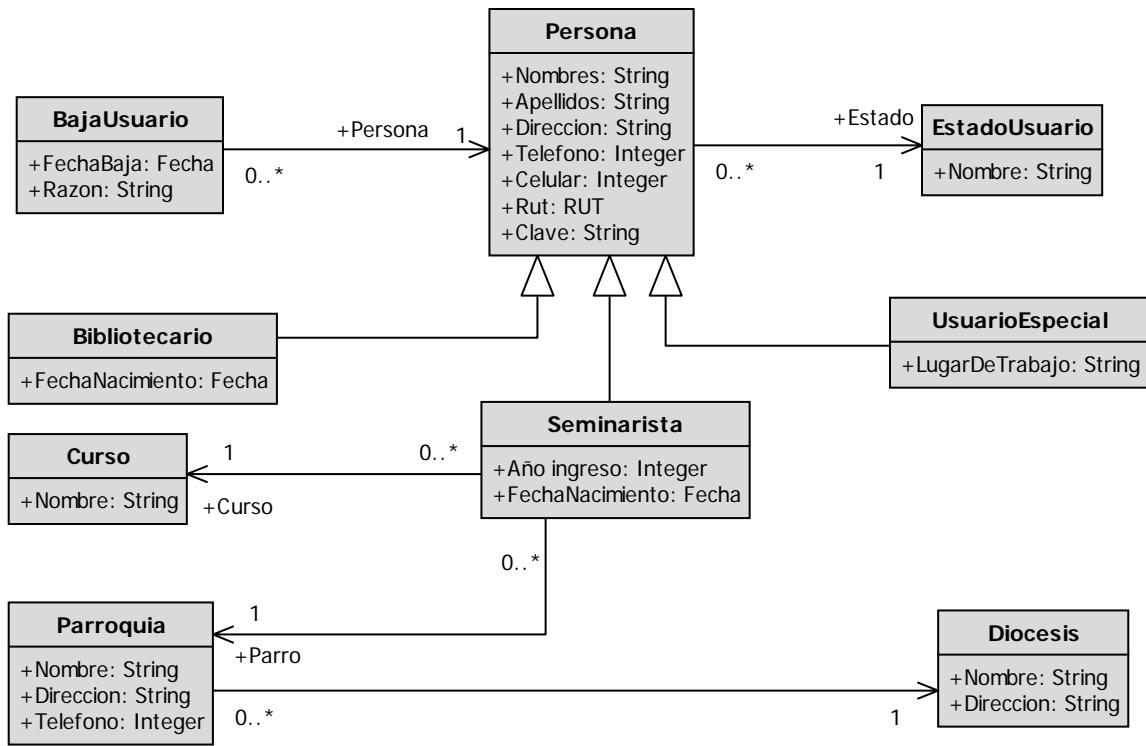


Figura 51: Modelo conceptual del paquete Administración de Usuarios.

## PAQUETE: PRÉSTAMOS Y DEVOLUCIONES.

### DIAGRAMA DE CASOS DE USO

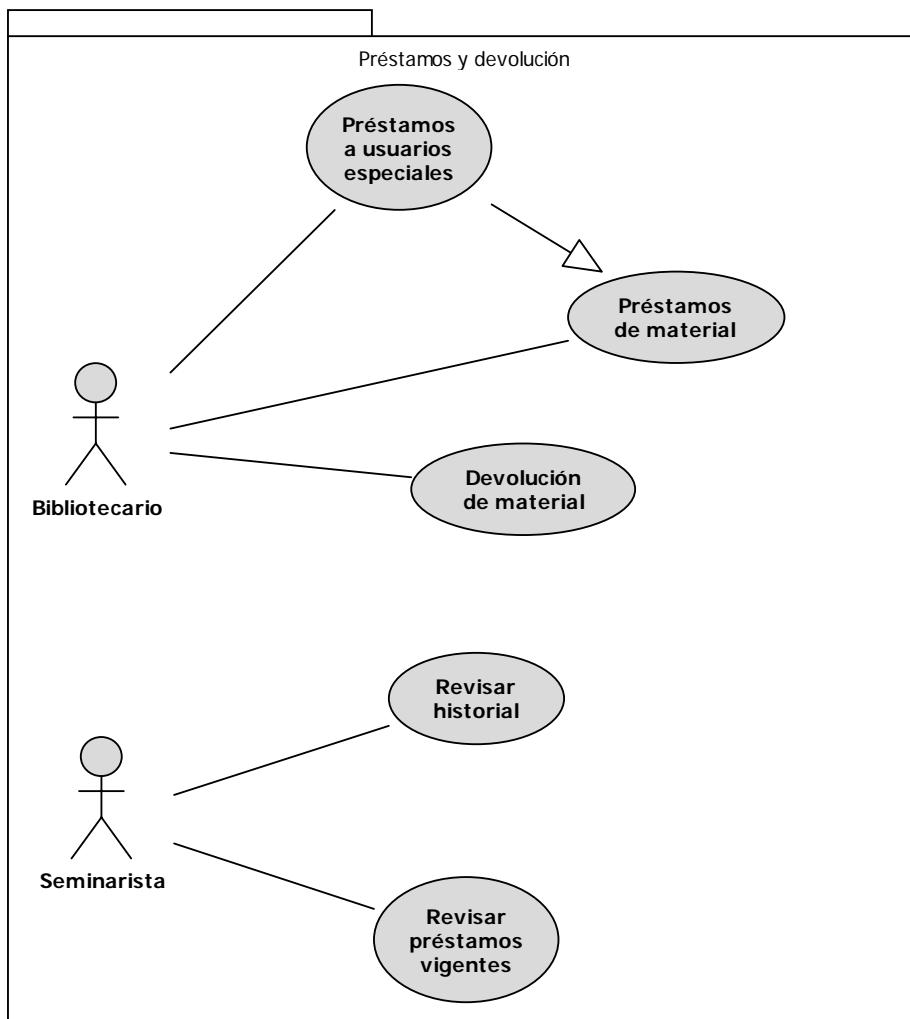


Figura 52: Paquete prestamos y devoluciones.

Figura en la que se muestra el diagrama de casos de uso perteneciente al paquete de préstamos y devoluciones interactuado por el bibliotecario, seminaristas y usuarios especiales. En el diagrama se puede observar el caso de uso PRÉSTAMOS DE MATERIAL que a su vez posee un caso especial como PRÉSTAMOS A USUARIOS ESPECIALES. Después que se produce un préstamo de material, debe ser devuelto mediante el caso de uso DEVOLUCIÓN DE MATERIAL. Un seminarista como usuarios especiales tiene acceso a REVISAR HISTORIAL y a REVISAR PRÉSTAMOS VIGENTES.

## **CASO DE USO: PRÉSTAMOS A USUARIOS ESPECIALES.**

<b>Actores</b>	: Bibliotecario.
<b>Propósito</b>	: realizar préstamos a usuarios que no pertenecen al seminario.
<b>Tipo</b>	: Secundario y esencial.
<b>Referencias cruzadas</b>	: Requerimiento R16.

### **Descripción :**

Si un usuario externo posee una cuenta, se le podrá realizar prestamos de material, pero estos serán solo por el día, sea de referencia o no, este tendrá que ser devuelto antes del término de la jornada, a no ser que el bibliotecario indique lo contrario. Para realizar el préstamo, el bibliotecario debe ingresar el RUT del usuario interesado, luego deberá verificar que la información existente en el sistema continúe vigente y realizar el préstamo.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario ingresa el RUT del usuario externo que desea realizar el préstamo.	
	Verifica que el usuario externo exista, y muestra información del usuario en pantalla.
Verifica que la información que está registrada en la cuenta continúe vigente.	
Ingresa el código del libro o revista (ya sea de forma manual o pistoleando el código de barras) para el préstamo.	

	Genera el préstamo e indica la fecha de devolución del libro o revista.
--	---

## CURSOS ALTERNOS

**Ítem 1:** Ingreso del RUT erróneo o inexistente, El sistema indica error y muestra nuevamente la opción de ingresar el RUT del seminarista.

**Ítem 2:** La verificación indica que el usuario externo no existe, Se cancela el préstamo en curso.

**Ítem 3:** La información registrada no está vigente. El bibliotecario primero debe modificar la información del usuario para que permanezca actualizada.

**Ítem 4:** Se ingresa un código inexistente, El sistema indica error y queda a la espera de cancelación del Bibliotecario o ingreso de un nuevo código.

**Ítem 5:** Si el bibliotecario desea cambiar la fecha de devolución que el sistema genera, puede hacer el cambio.

## **CASO DE USO: PRÉSTAMOS DE MATERIAL.**

<b>Actores</b>	: Bibliotecario.
<b>Propósito</b>	: Realizar préstamo del material de la biblioteca (libros y revistas).
<b>Tipo</b>	: Primario y esencial.
<b>Referencias cruzadas</b>	: Requerimiento R13

### **Descripción :**

El bibliotecario, una vez que recibe una petición de préstamo por parte del seminarista, lo ingresa al sistema identificándolo con su RUT. Luego, el seminarista le indica el código de clasificación del libro o revista, para que este lo busque en los estantes y lo registre con el código de barras, y así registrar el préstamo. Los préstamos de los libros serán según correspondan a los siguientes criterios:

- Libro de **referencia** (Libros más solicitados): 1 día de préstamo.
- Libro de **No referencia**: 1 mes de préstamo.
- Revista: 1 día de préstamo.

Es importante mencionar que todo préstamo debe quedar con la opción de cambiar la fecha de devolución, según el criterio del bibliotecario.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario ingresa el RUT del usuario que desea realizar el préstamo.	
	Verifica que el usuario se encuentre en condición regular (sin atraso de devolución)

	para realizar el préstamo.
Ingresá el código de registro del libro o revista (ya sea de forma manual o pistoleando el código de barras).	
	Genera el préstamo e indica la fecha de devolución del libro o revista.

## CURSOS ALTERNOS

**Ítem 1:** Ingreso del RUT erróneo o inexistente, El sistema indica error y muestra nuevamente la opción de ingresar el RUT del seminarista.

**Ítem 2:** La verificación indica que el usuario no puede realizar préstamo, Se cancela el préstamo en curso y se indica la fecha en que el usuario estará en condición regular.

**Ítem 3:** Se ingresa código inexistente, El sistema indica error, queda a la espera de otro código

**Ítem 4:** Si el bibliotecario desea cambiar la fecha de devolución que el sistema genera, puede hacer el cambio.

## **CASO DE USO: DEVOLUCIÓN DE MATERIAL.**

**Actores** : Bibliotecario.

**Propósito** : Realizar la devolución de un libro o revista.

**Tipo** : Primario y esencial.

**Referencias cruzadas** : Requerimiento R14

**Descripción** :

Para concretar una devolución, se debe ingresar el código del libro, y en caso de que el libro se entregue fuera de la fecha establecida, el usuario vinculado al libro quedará suspendido de realizar préstamos por los días equivalentes a los días de atraso, es decir, por cada día de atraso se suspende un día de la posibilidad de realizar prestamos. La suspensión del usuario se manifestará por el estado en que se mantendrá, INACTIVO.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el bibliotecario ingresa el código del libro al sistema para efectuar la devolución (ya sea de forma manual o pistoleando el código de barras).	
	Vincula el libro al préstamo al que pertenece, para registrar la devolución del seminarista al que está vinculado y verificar que se entregó dentro de la fecha estipulada.
	Informa la devolución satisfactoria.

## **CURSOS ALTERNOS**

**Ítem 1:** Se ingresa un código que no pertenece a un préstamo. Se indica error, y se cancela el proceso.

**Ítem 2:** El libro devuelto se entrega fuera del plazo de entrega. Se indica error y se genera sanción de acuerdo a los días de atraso indicando en pantalla.

## **CASO DE USO: REVISAR HISTORIAL.**

<b>Actores</b>	: Seminarista, Bibliotecario.
<b>Propósito</b>	: Revisar los préstamos que ha realizado un usuario.
<b>Tipo</b>	: Secundario y esencial.
<b>Referencias cruzadas</b>	: Requerimiento R13.
<b>Descripción</b>	:  Un usuario puede revisar todos los préstamos que ha realizado en la biblioteca, además de aparecer el listado de libros o revistas que pertenecen a los préstamos, aparecerán todas las suspensiones asociadas.  Un bibliotecario también puede revisar el historial de un usuario, cuando está actualizando sus datos.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el Seminarista o bibliotecario desean ver el historial de materiales que se ha pedido. En el caso del bibliotecario debe indicar el RUT del usuario al que desea revisar el historial.	
	Busca los préstamos históricos en la base de datos.
	Muestra en pantalla todos los prestamos que ha realizado el usuario.

## **CURSOS ALTERNOS**

**Ítem 3:** El usuario no posee préstamos históricos. Indica que no posee préstamos históricos.

## **CASO DE USO: REVISAR PRÉSTAMOS VIGENTES.**

<b>Actores</b>	: Seminarista, Bibliotecario.
<b>Propósito</b>	: Revisar los préstamos Vigentes que posee usuario.
<b>Tipo</b>	: Secundario y esencial.
<b>Referencias cruzadas</b>	: Requerimiento R13.
<b>Descripción</b>	:
	Un usuario puede revisar todos los préstamos vigentes que mantiene en la biblioteca. Estos aparecerán en una lista indicando la fecha de préstamo y la fecha tope en que debería ser devuelto el material.
	Un bibliotecario también puede revisar los préstamos vigentes de un usuario, cuando está actualizando sus datos.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el seminarista o bibliotecario desean ver los préstamos vigentes, en el caso del bibliotecario debe indicar el RUT del usuario al que desea ver los prestamos vigentes.	
	Busca los préstamos vigentes en la base de datos.
	Muestra en pantalla todos los préstamos vigentes que posee.

## **CURSOS ALTERNOS**

**Ítem 3:** El usuario no posee préstamos vigentes. Indica que no posee préstamos vigentes.

# MODELO CONCEPTUAL

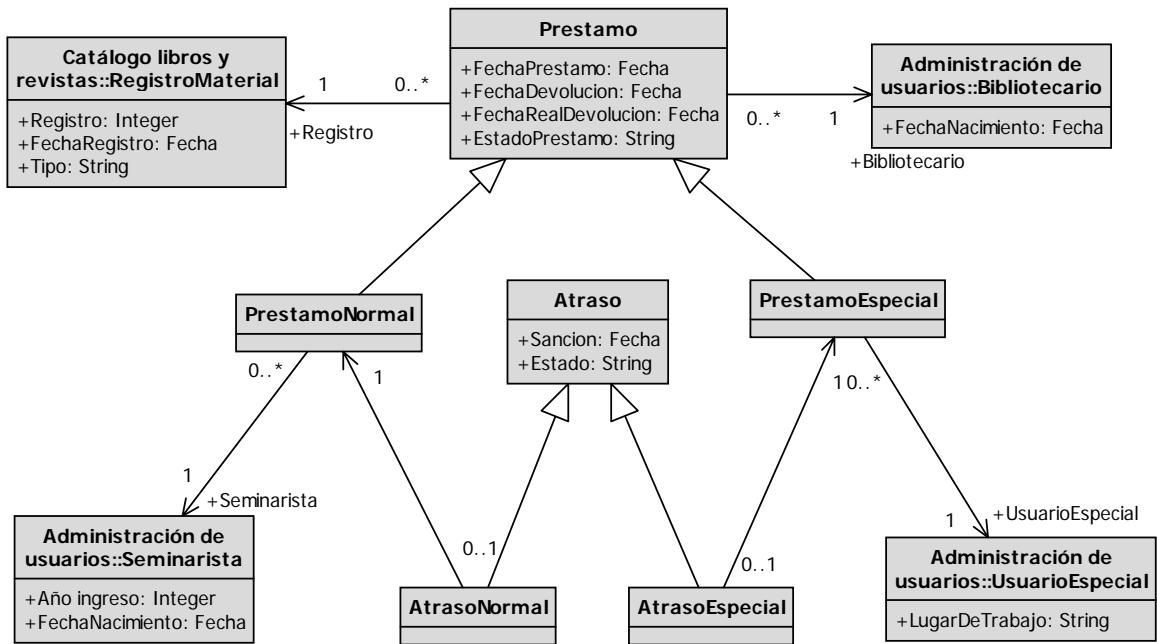


Figura 53: Modelo conceptual del paquete Préstamos y devoluciones

## PAQUETE: PUBLICACIONES DIGITALES.

### DIAGRAMA DE CASOS DE USO

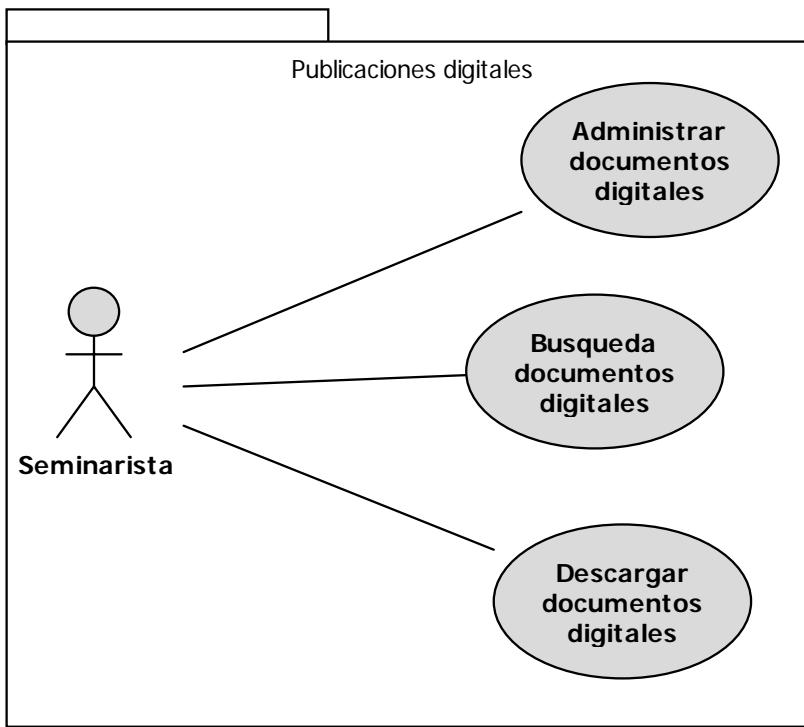


Figura 54: Paquete Publicaciones digitales.

Figura en la que se muestra el diagrama de casos de uso perteneciente al paquete de publicaciones digitales interactuado por seminaristas. En el diagrama se puede observar el caso de uso ADMINISTRAR DOCUMENTOS DIGITALES en el que se tiene la opción de ingresar archivo, eliminar archivo y modificar propiedades. Luego se aprecia BÚSQUEDA DOCUMENTOS DIGITALES para buscar con más exactitud algún archivo y finalmente se aprecia DESCARGAR DOCUMENTOS DIGITALES para poder observar el contenido del material.

## **CASO DE USO: ADMINISTRAR DOCUMENTOS DIGITALES.**

**Actores** : Seminarista.

**Propósito** : La administración de los documentos digitales.

**Tipo** : Secundario y esencial.

**Referencias cruzadas** : Requerimiento R22.

**Descripción** :

Un seminarista tiene derecho a subir archivos a su cuenta, y a la vez tiene el deber de eliminar cuando sea necesario.

El seminarista a modo de administración podrá:

- Agregar archivo.
- Eliminar archivo.
- Modificar propiedades.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
Selecciona la opción que desea para administrar los documentos digitales que le pertenecen.	
	Valida la opción elegida por el seminarista y despliega opciones.
	Deriva a la sección que corresponda según la elección:  - <i>Agregar archivo</i> : Deriva a la sección

	<p>agregar archivo.</p> <ul style="list-style-type: none"><li>- <i>Eliminar archivo:</i> Deriva a la sección eliminar archivo</li><li>- <i>Modificar propiedades:</i> Deriva a la sección Modificar propiedades.</li></ul>
--	--

## CURSOS ALTERNOS

## **SECCIÓN: AGREGAR ARCHIVO.**

**Descripción :**

Un seminarista, puede almacenar en su cuenta archivos de texto para que estos sean visualizados por los demás usuarios. El Seminarista debe examinar en el computador el archivo a subir o indicar la ruta, para posteriormente indicarle un titulo y una descripción que serán enviadas al sistema para vincularlas a la cuenta.

**INFORMACIÓN ARCHIVO:** Titulo del archivo / Descripción del archivo

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
Examina en el computador el archivo que desea almacenar en la cuenta.	
Especifica Título y descripción del archivo y envía al sistema.	
	Comprueba el tamaño del archivo.
	Almacena el archivo.

## **CURSOS ALTERNOS**

## **SECCIÓN: ELIMINAR ARCHIVO.**

**Descripción** :

Un seminarista puede eliminar documentos digitales de su cuenta cuando lo estime conveniente, ya sea por liberar espacio de su cuenta o por cualquier motivo, este deberá seleccionar en pantalla cual de sus archivos desea eliminar.

### **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
Indica en pantalla cual de sus archivos desea eliminar, ya que aparecerán en pantalla todos los archivos ligados a su cuenta.	
	Elimina el archivo del sistema.

### **CURSOS ALTERNOS**

## **SECCIÓN: MODIFICAR PROPIEDADES.**

**Descripción** :

Si un seminarista, después de haber almacenado un documento digital, concluye que el título o la descripción no son acordes al archivo o piensa que existe otro mejor, puede modificar las propiedades, escogiendo en pantalla cual es el archivo al que desea modificar las propiedades para luego modificar la información que estime conveniente.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
Indica en pantalla cual es el archivo al que desea modificar sus propiedades, ya que aparecerán en pantalla todos los archivos ligados a su cuenta.	
	Muestra formulario en pantalla con las propiedades del documento al que se desea modificar.
Modifica la información que estime conveniente y envía formulario al sistema.	
	Registra los cambios.

## **CURSOS ALTERNOS**

## **CASO DE USO: DESCARGAR DOCUMENTOS DIGITALES.**

**Actores** : Seminarista.

**Propósito** : Descargar un documento digital asociado a cualquier seminarista.

**Tipo** : Secundario y esencial.

**Referencias cruzadas** : Requerimiento R23

**Descripción** :

En caso que un seminarista necesite apreciar la información que se encuentra en algún documento digital, ya sea de él mismo o de los compañeros, después de realizar una búsqueda podrá descargarlo para luego quedárselo en su poder.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el seminarista selecciona el archivo que desea descargar.	
	Muestra el archivo en pantalla.
Visualiza el archivo y si quiere lo almacena en algún medio para aquello.	

## **CURSOS ALTERNOS**

**Ítem 2:** El archivo que se escoge en pantalla está dañado o no existe. Se indica error y se cancela el proceso.

## **CASO DE USO: BÚSQUEDA DOCUMENTO DIGITALES.**

<b>Actores</b>	: Seminarista.
<b>Propósito</b>	: Encontrar documentos digitales mediante criterios de búsqueda.
<b>Tipo</b>	: Secundario y esencial.
<b>Referencias cruzadas</b>	: Requerimientos R22, R23.

**Descripción** :

Si un seminarista necesita realizar una búsqueda de documentos digitales almacenados por otros usuarios o por él, debe seleccionar el criterio de búsqueda que estime conveniente. Los criterios de búsqueda se mencionan a continuación:

### **PARA DOCUMENTOS DIGITALES: Título / Descripción**

Luego de escoger el criterio de búsqueda, se debe digitar la o las palabras que se desean buscar, para que el sistema muestre el link del archivo

### **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el seminarista selecciona el criterio de búsqueda, digita palabras a buscar para luego enviar al sistema.	
	Según el criterio escogido y la o las palabras digitadas, realiza la búsqueda.
	Muestra los links de los archivos encontrados, además del título y descripción.

## **CURSOS ALTERNOS**

**Ítem 1:** Envía formulario sin escribir nada. Indica error.

**Ítem 3:** No se encuentran coincidencias, se indica que no hubo éxito en la búsqueda.

## MODELO CONCEPTUAL

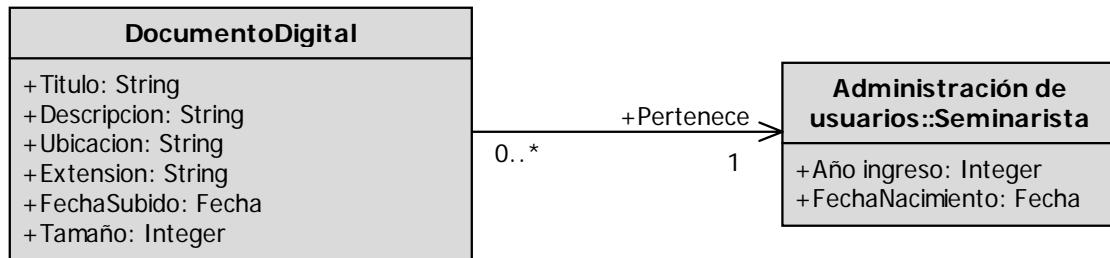


Figura 55: Modelo conceptual del paquete Publicaciones Digitales

## PAQUETE: SISTEMA DE MENSAJERÍA.

### DIAGRAMA DE CASOS DE USO

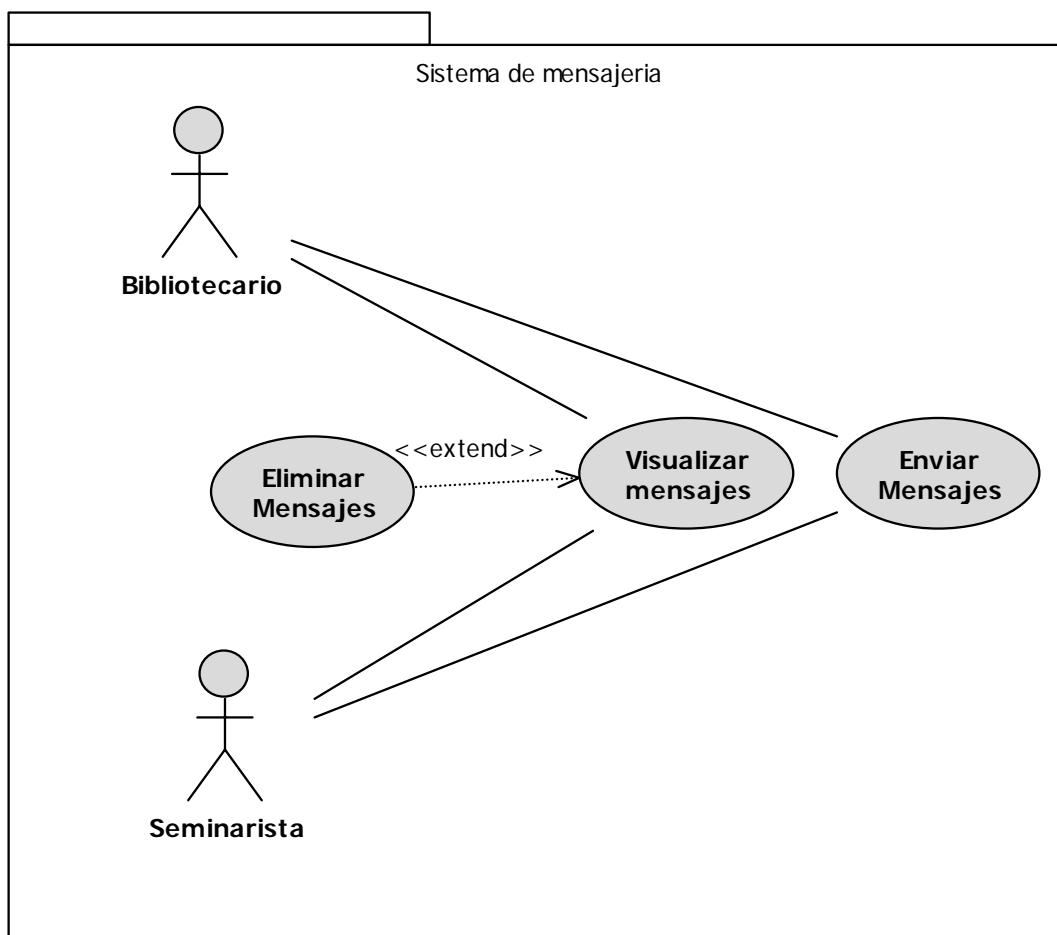


Figura 56: Paquete sistema de mensajería.

Figura en la que se muestra el diagrama de casos de uso perteneciente al paquete sistema de mensajería interactuado por el bibliotecario y seminaristas. En el diagrama se puede observar el caso de uso ENVIAR MENSAJES los cuales podrán ser vistos por los receptores mediante el caso de uso VISUALIZAR MENSAJES y si este lo desea podrá ELIMINAR MENSAJES.

## **CASO DE USO: VISUALIZAR MENSAJES.**

<b>Actores</b>	: Seminarista, Bibliotecario.
<b>Propósito</b>	: Visualizar mensajes que han enviado a la cuenta personal.
<b>Tipo</b>	: Secundario y esencial.
<b>Referencias cruzadas</b>	: Requerimiento R20.

**Descripción** :

En caso que un seminarista o bibliotecario desee revisar si le han enviado mensajes, debe ver en su cuenta, en la cual aparecerán todos los mensajes incluyendo los que ya han sido vistos.

El usuario primero debe seleccionar el mensaje que desea ver, ya que aparecerá una lista con los asuntos, el emisor, y la fecha de emisión, para luego tener acceso al cuerpo del mensaje.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el seminarista o bibliotecario selecciona el mensaje que desea visualizar.	
	Muestra el mensaje en pantalla.
Visualiza el mensaje.	

## **CURSOS ALTERNOS**

## **CASO DE USO: ELIMINAR MENSAJES PERSONALES.**

**Actores** : Seminarista, Bibliotecario.

**Propósito** : Eliminar un mensaje del sistema.

**Tipo** : Secundario y esencial.

**Referencias cruzadas** : Requerimiento R20.

**Descripción** :

Un usuario si lo desea, puede eliminar un mensaje de su cuenta, ya sea para liberar espacio o para tener los mensajes de forma más ordenada. El mensaje a eliminar primero debe ser seleccionado, luego el sistema preguntará al usuario si está seguro de la eliminación y una vez que el usuario de una respuesta positiva, el sistema eliminara el mensaje.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el seminarista o bibliotecario selecciona el mensaje que desea eliminar del sistema.	
	Pregunta al usuario si está seguro de la eliminación del mensaje.
Responde positivamente al sistema.	
	Elimina el mensaje.

## **CURSOS ALTERNOS**

**Ítem 3:** Responde negativamente, el sistema cancela la eliminación.

## **CASO DE USO: ENVIAR MENSAJES.**

**Actores** : Seminarista, Bibliotecario.

**Propósito** : Enviar un mensaje.

**Tipo** : Secundario y esencial.

**Referencias cruzadas** : Requerimiento R20.

**Descripción** :

Un usuario puede enviar un mensaje a otro usuario, a un curso o a todos los usuarios de la biblioteca. Los cursos se mencionan a continuación:

**CURSOS:** Propedéutico / 2° Filosofía / 3° Filosofía / 1° Teología / 2° Teología / 3° Teología / 4° Teología.

El usuario primero debe ingresar el o los destinatarios, escribir un asunto, luego escribir el cuerpo del mensaje y enviar el mensaje.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso da uso comienza cuando el seminarista o bibliotecario desea enviar un mensaje.	
Selecciona los destinatarios del mensaje.	
Ingresa el asunto del mensaje	
Ingresa el cuerpo del mensaje.	
Envía el mensaje al sistema.	

	Envía el mensaje a o los destinatarios.
--	---

## CURSOS ALTERNOS

**Ítem 4:** No ha seleccionado destinatarios, asunto o cuerpo del mensaje. Indica error y en caso de que existan destinatarios da posibilidad de enviar de todas formas.

## MODELO CONCEPTUAL

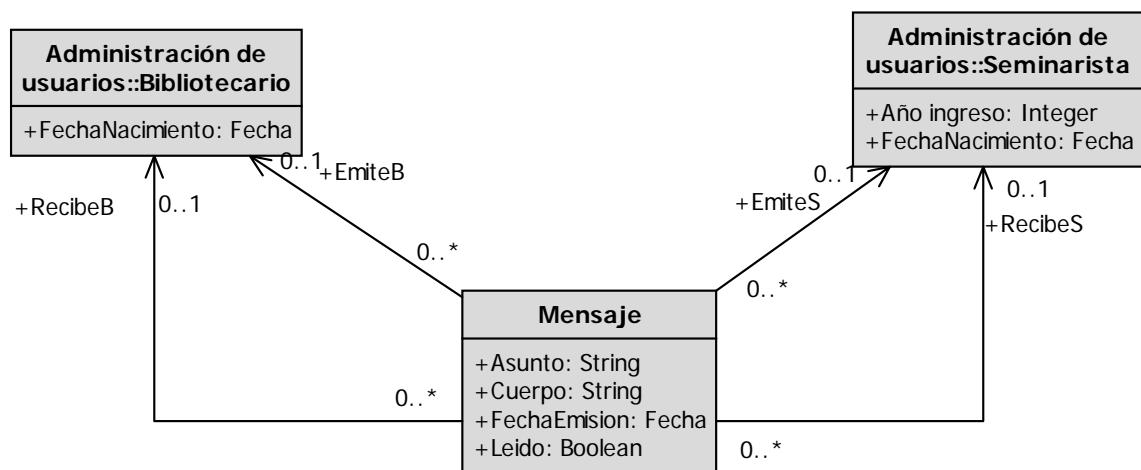


Figura 57: Modelo conceptual del paquete Publicaciones Digitales

## PAQUETE: RECOMENDACIÓN DE BÚSQUEDA.

### DIAGRAMA DE CASOS DE USO

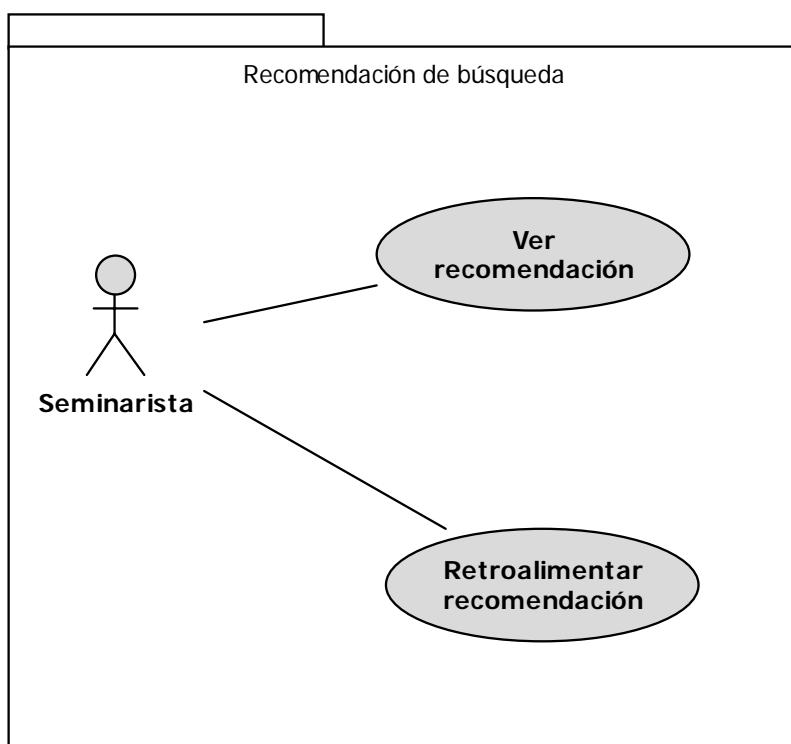


Figura 58: Paquete recomendación de búsqueda.

Figura en la que se muestra el diagrama de casos de uso perteneciente al paquete recomendación de búsqueda interactuado por seminaristas. En el diagrama se puede observar el caso de uso VER RECOMENDACIÓN que sirve para tomar la decisión de que libro elegir de la búsqueda que arroja el sistema, para ello cada seminarista puede RETROALIMENTAR RECOMENDACIÓN para que la experiencia de leer un material sea de utilidad para otros.

## **CASO DE USO: VER RECOMENDACIÓN.**

<b>Actores</b>	: Seminarista.
<b>Propósito</b>	: Elegir un material con más seguridad.
<b>Tipo</b>	: Secundario y esencial
<b>Referencias cruzadas</b>	: Requerimiento R18.

### **Descripción :**

Un seminarista, cuando está buscando un libro en el sistema, aparecerá una lista con todo el material que ha coincidido con la búsqueda, pero cada material estará asociado a un promedio (1 a 10) y además el usuario podrá apreciar los comentarios que han escrito sus pares que ya han leído el libro, las opiniones estarán ordenadas por fecha, de la más reciente a la menos reciente, pero al comienzo aparecerá el comentario del usuario que puso la mejor y peor nota.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el seminarista ha buscado libros en el sistema y desea visualizar las recomendaciones que se arrojaron de acuerdo a la búsqueda, que ha realizado.	
Indicar a cual libro desea revisar las recomendaciones.	
	Muestra los comentarios y calificaciones que han realizado los usuarios que han leído el material.

## **CURSOS ALTERNOS**

**Ítem 1:** No existen promedios asociados. Solo muestra información del libro.

## **CASO DE USO: RETROALIMENTAR RECOMENDACIÓN.**

**Actores** : Seminarista.

**Propósito** : Retroalimentar recomendaciones.

**Tipo** : Secundario y esencial.

**Referencias cruzadas** : Requerimiento R20.

**Descripción** :

Cuando un seminarista ha leído un libro, es muy recomendable que este deje su opinión y una nota hacia el material (de 1 a 10) para que otros usuarios puedan apreciarla.

Esta recomendación puede realizarse desde la cuenta de cada seminarista, en donde aparece el historial de los prestamos que ha realizado, por cada préstamo el seminarista puede realizar sólo una recomendación y no puede recomendar un libro si han pasado 30 días desde que ha realizado el préstamo.

## **CURSO NORMAL DE LOS EVENTOS**

<b>Acción del actor</b>	<b>Acción del sistema</b>
El caso de uso comienza cuando el seminarista desea recomendar un libro que ya ha leído.	
	Muestra formulario en pantalla para que el seminarista rellene con el comentario y nota correspondiente al libro.
Determina una nota y el comentario correspondiente.	

	Almacena en el sistema la recomendación para ser visualizada por otros usuarios.
--	--

## CURSOS ALTERNOS

**Ítem 3:** El seminarista no escribe nota o comentario. Sistema indica permitiendo la continuación de este en caso que falte el comentario.

## MODELO CONCEPTUAL

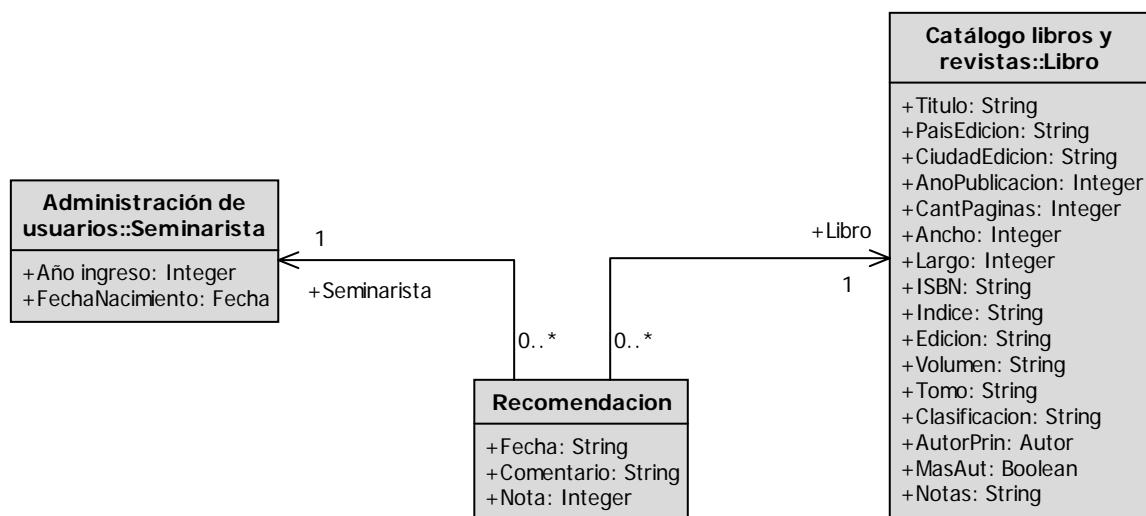


Figura 59: Modelo conceptual del paquete Recomendación de búsqueda.

**ANEXO 3**  
**DOCUMENTO DE APOYO DB4O**

## **INTRODUCCIÓN**

El presente documento de apoyo, intentará apoyar la decisión de implementar el proyecto “Sistema Orientado a Objetos para Biblioteca del Seminario Metropolitano Concepción” con el motor de base de datos orientado a objetos “Db4o”. Además se dará énfasis en cuál es la importancia de realizar el proyecto con una base de datos con tecnología a objetos.

## **BD ORIENTADA A OBJETOS.**

Una base de datos tradicional solo almacena datos, sin procesarlos, de modo que resulten independientes de los procedimientos. Los datos son accesibles a diferentes usuarios, con diversos propósitos. Por lo contrario, una base de datos orientada a objetos almacena objetos junto con los métodos propios de su comportamiento.

Las bases de datos orientadas a objetos permiten el desarrollo y mantenimiento de aplicaciones complejas con un costo significativamente menor. Permiten que el mismo modelo conceptual se aplique al análisis, diseño, programación, definición y acceso a la base de datos. (Martin, y otros, 1994)

## DE RELACIONAL A OBJETOS

Al trabajar con un lenguaje orientado a objetos es imprescindible la persistencia<sup>8</sup>, por lo que en la actualidad esta problemática se resuelve con una base de datos relacional, luego los objetos deben ser mapeados a tablas (Figura 60) y posteriormente transformarlos a objetos, lo que hace empobrecer el paradigma orientado a objetos.

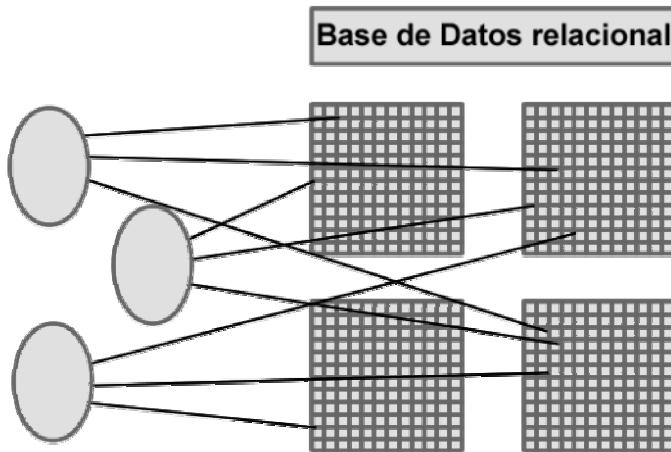


Figura 60: Forma en que se guardan los objetos en una base de datos relacional

Por lo contrario, una base de datos orientada a objetos toma cada objeto y lo guarda con su estado actual (Figura 61), sin importar lo complejo que éste sea, encontrando la persistencia ideal cuando se necesita una base de datos transparente y fácil de utilizar. Esto hace que se pueda incorporar rápidamente en un proyecto sin sacrificar desempeño (Lavintman, y otros, 2007).

<sup>8</sup> Es la capacidad que tienen los objetos de conservar su estado e identidad entre distintas ejecuciones del programa que los creó o de otros programas que accedan a ellos.

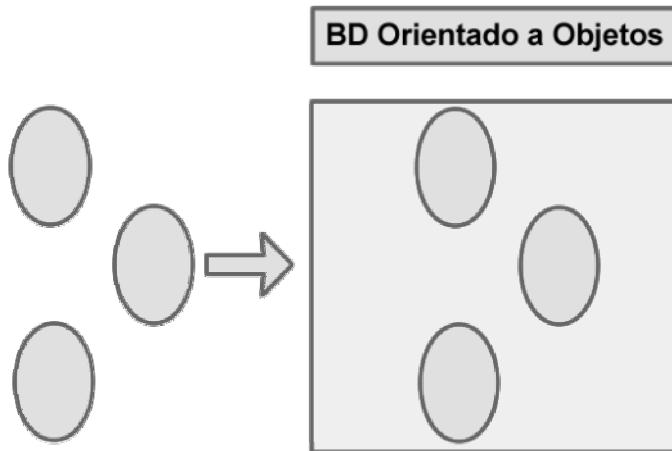


Figura 61: Forma en que se guardan los Objetos en una base de datos Orientada a Objetos

## CARACTERÍSTICAS PRINCIPALES DE LAS BASES DE OBJETOS

Hay una serie de características que las bases de objetos poseen y nos permiten conocer algunos detalles de la tecnología. Las bases de objetos no solo permiten trabajar transparentemente con un entorno de programación basado en objetos sino que soportan todos los conceptos de la tecnología de objetos:

- **Agregación:** objetos que están compuestos por otros objetos
- **Encapsulamiento:** almacenamiento de atributos junto con métodos. No todas las bases soportan métodos pero se apoyan en las clases definidas en los esquemas para reconstruir el objeto con sus métodos.
- **Herencia:** los objetos heredan atributos y comportamiento de sus objetos padre.
- **Polimorfismo:** permite a los objetos responder de forma distinta a un mismo mensaje (por ejemplo implementando un método de otra manera). Se soportan distintas versiones de los objetos. (Viscuso, 2007)

## QUE ES Db4o

DB4o es un novedoso motor de base de datos orientada a objetos. Sus siglas corresponden a la expresión "**D**ata**B**ase **4** (for) **O**bjects", que a su vez es el nombre de la compañía que lo desarrolla: *db4objects, Inc.*

DB4o es una solución de persistencia ideal cuando se necesita una base transparente y fácil de utilizar, para incorporar rápidamente en un proyecto Java o .NET sin sacrificar desempeño. Logra acortar los tiempos de entrega de nuestras aplicaciones de forma significativa, ya que permite concentrarnos directamente en el dominio del problema (nuestros objetos). Además, ofrece características avanzadas, como soporte transparente a cambios en las versiones de los objetos (evolución del esquema), cero administración (no requiere DBA) y consultas nativas.  
(Lavintman, y otros, 2007)

El motor de la base de datos de db4o consiste en un solo archivo .jar. Esto es todo lo que usted tiene para programar.

## INSTALACIÓN

Es necesario mencionar que la base de datos Db4o está creada para los lenguajes de programación JAVA y .NET, en este caso la instalación estará definida solo para JAVA.

La instalación de Db4o es extremadamente sencilla, este motor es un archivo .jar, es decir, una librería de Java, por lo que es necesario agregarla al CLASSPATH o simplemente al trabajar un proyecto asignarla como librería.

## CARACTERÍSTICAS DE DB40

El motor de base de objetos Db4o permite cubrir (y exceder) los siguientes requerimientos.

**Consumo mínimo de recursos:** db4o está diseñado para ser embebido en clientes u otros componentes de software, de manera totalmente invisible para el usuario final. Es por eso que viene como una librería fácil de incorporar, con un tamaño que ronda los 400 Kb. Como el motor corre en el mismo proceso de la aplicación, el usuario cuenta con control completo sobre la administración de memoria, y puede realizar procesos de profiling<sup>9</sup> y debugging<sup>10</sup> del desempeño sobre todo el sistema. Si la aplicación está corriendo, la base también lo está, sin excepción. Y aún más importante, db4o es extremadamente flexible a la hora de actualizar una base existente con un modelo de objetos que ha cambiado. Siempre asume que no hay un administrador de base de datos y, por lo tanto, permite a la aplicación cambiar del modelo viejo al modelo nuevo de modo transparente.

**Alto rendimiento:** El rendimiento de db4o es equiparable al de los mejores sistemas de bases de datos tradicionales. Los benchmarks que se muestran a continuación indican el desempeño de db4o comparado con el de una base de datos SQL corriendo en una máquina de escritorio:

Test	Db4o	SQL de escritorio
Insert Complejo	1	48,25
Select Complejo	1	2,32
Insert Entero	1	27,74
Select Entero	1	0,7
Insert String	1	20,26
Select String	1	0,48

Tabla 1: Tabla que muestra el desempeño de db4o comparado con el de una base de datos SQL de escritorio

<sup>9</sup> significa medir la rapidez con la que se ejecuta tu programa.

<sup>10</sup> es el proceso metodológico para encontrar y reducir bugs (errores).

**Fácil implementación:** Sólo hay que agregar la librería única de db4o (.jar o .dll) al entorno de desarrollo, abrir el archivo de base de datos y almacenar cualquier objeto (sin importar su complejidad) con una sola línea de código. Por ejemplo, en Java:

```
public void guardar(Objeto obj){  
  
    ObjectContainer db = Db4o.openFile("Objetos.yap");  
  
    db.set(obj);  
  
    db.commit();  
  
    db.close();  
}
```

Esta facilidad de uso sin precedentes se traduce en una reducción drástica de los tiempos de desarrollo. Se elimina efectivamente todo el trabajo de diseñar, implementar y mantener el esquema de base de datos, ya que el propio modelo de objetos conforma este esquema.

**Portabilidad:** Pocas bases de datos embebibles corren de manera nativa en tantas plataformas orientadas a objetos como db4o. Ésta permite desarrollar aplicaciones para desplegar en varias plataformas (por ejemplo, en PDAs) o en combinaciones heterogéneas de clientes Windows y servidores Java. Este motor soporta desde el JDK 1.1.x de Java hasta el 5.0, y es soportado en Java EE y Java SE.

También es compatible con todas las plataformas .NET, el Compact Framework y Mono; y soporta todos los lenguajes .NET administrados, como C#, VB.NET, ASP.NET, Boo y Managed C++.

**Confiabilidad:** Finalmente, db4o soporta todas las características ACID<sup>11</sup>. Múltiples usuarios simultáneos de una base db4o son efectivamente aislados, y sus operaciones son serializadas de forma transparente por la librería. Las transacciones se terminan con los métodos **commit()** y **rollback()** de la clase **ObjectContainer**. Y en caso de que el sistema se caiga durante una actualización de la base de datos, cuando el **ObjectContainer** de db4o es reabierto, se completan de forma correcta todas las transacciones interrumpidas. (Lavintman, y otros, 2007)

## ALGUNOS EJEMPLOS.

Ya se mencionó que el motor de base de datos es una librería .jar, y la base de datos es un archivo que tiene extensión .yap, tomemos por ejemplo una base de datos de nombre “based.yap”. En ella se almacenaran todos los objetos que necesiten de persistencia.

```
ObjectContainer db=Db4o.openFile("based.yap") ;
```

```
try{
```

```
}
```

```
finally{
```

```
    db.close();
```

```
}
```

Para cerrar la base de datos se utilizó el método db.close(). El objeto db de tipo ObjectContainer contiene el archivo based.yap que fue abierto por el método Db4o.openFile().

Para guardar un objeto primero se necesita que la base de datos se encuentre abierta. En el siguiente ejemplo se tomará el Objeto obj que será guardado con el método db.set().

```
Objeto obj=new Objeto();
```

```
Db.set(obj);
```

Para recuperar objetos, se debe crear un objeto con las propiedades por defecto y aquellas que queremos que sean idénticas se deben ingresar con sus respectivos métodos o mediante el constructor, por ejemplo, si se quiere recuperar objetos personas que contengan como nombre “Alejandro”, se debe ingresar sólo aquel atributo, pero los demás se deben dejar por defecto. Para recuperar todos los objetos de nombre Alejandro se utiliza el método db.get().

```
Persona obj = new Persona("Alejandro", null, null);
```

```
ObjectSet result=db.get(obj);
```

---

<sup>11</sup> ACID (Atomicidad, Consistencia, Aislamiento, Durabilidad)

```
While(result.hasNext()){  
    System.out.println(result.next());  
}
```

Al recuperar objetos se almacenan en un objeto de tipo ObjectSet los que pueden ser manipulados por sus métodos, consideremos el objeto result de tipo ObjectSet, utilizamos el método result.hasNext() que comprueba si existe otro objeto o no, con resultNext() extraemos el siguiente objeto que se recupera.

## CONCLUSIÓN

Db4o es la base de datos de código libre ideal para el proyecto “Sistema orientado a Objetos para Biblioteca del Seminario Metropolitano Concepción”, ya que contiene las propiedades de la Tecnología Orientada a Objetos, que es lo que busca el proyecto. Además, es una base de datos que cumple con los requerimientos de ser bajo en consumo de recursos y poseer un alto rendimiento. En definitiva, con Db4o logramos lo que estamos buscando, “**PERSISTENCIA**”.

## BIBLIOGRAFÍA

**Lavintman, Alan y Viscuso, German.** 2007. *db4o: una alternativa a la persistencia* . s.l. : www.db4o.com, 2007.

**Martin, James y Odell, James.** 1994. *Analisis y diseño orientado a objetos*. s.l. : Prentice Hall, 1994.

**Viscuso, German.** 2007. *Bases de Objetos*. s.l. : www.db4o.com, 2007.

**ANEXO 4**  
**DOCUMENTO DE APOYO JSP**

## INTRODUCCIÓN

El presente documento de apoyo, relacionado a la tecnología JSP, intentará apoyar la decisión de implementar con JSP el proyecto “Sistema Orientado a Objetos para Biblioteca del Seminario Metropolitano Concepción”. Además se dará énfasis en dar una definición clara a JSP con el fin de controlar de mejor forma esta tecnología.

El actual documento también menciona algunas definiciones importantes como, que es un servlets y la relación con JSP a través de su arquitectura. También se detalla el funcionamiento de esta tecnología para aclarar los aspectos más importantes de Java Server Pages.

## QUE ES JSP.

JSP (Java Server Pages) es una Tecnología Java que permite generar contenido dinámico para web, en forma de documentos HTML, XML o de otro tipo.

Las JSP's permiten la utilización de código Java mediante scripts. Además es posible utilizar algunas acciones JSP predefinidas mediante etiquetas. Estas etiquetas pueden ser enriquecidas mediante la utilización de Librerías de Etiquetas (TagLibs o Tag Libraries) externas e incluso personalizadas.

## Tecnología JAVA.

La tecnología Java se creó como una herramienta de programación en una pequeña operación secreta y anónima denominada "the Green Project" en Sun Microsystems en el año 1991.

El lenguaje de programación Java ha sido totalmente mejorado, ampliado y probado por una comunidad activa de unos cuatro millones de desarrolladores de software.

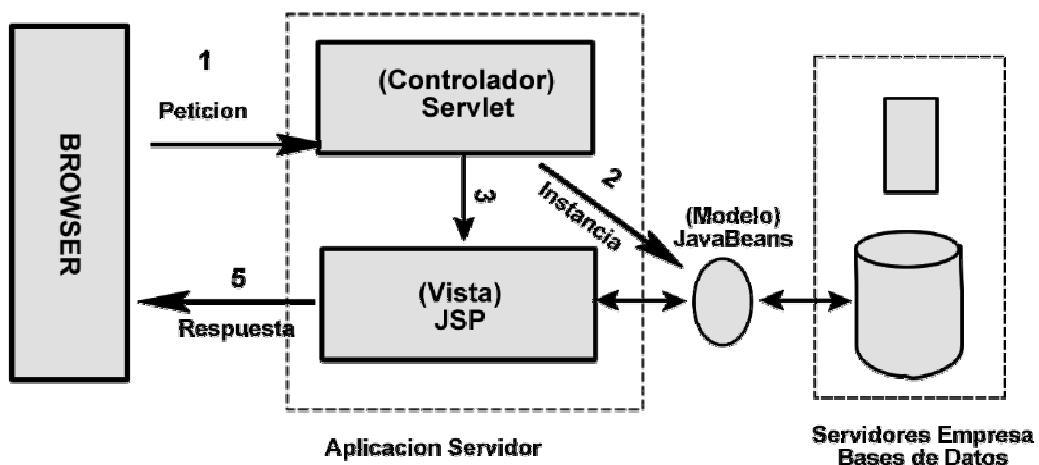
La tecnología Java, una tecnología madura, extremadamente eficaz y sorprendentemente versátil, se ha convertido en un recurso inestimable ya que permite a los desarrolladores:

- Desarrollar software en una plataforma y ejecutarlo en prácticamente cualquier otra plataforma
- Crear programas para que funcionen en un navegador Web y en servicios Web
- Desarrollar aplicaciones para servidores como foros en línea, tiendas, encuestas, procesamiento de formularios HTML, etc.
- Combinar aplicaciones o servicios basados en la tecnología Java para crear servicios o aplicaciones totalmente personalizados
- Desarrollar potentes y eficientes aplicaciones para teléfonos móviles, procesadores remotos, productos de consumo de bajo coste y prácticamente cualquier dispositivo digital
- (Java, 2007)

## ARQUITECTURA JSP

Desde el punto de vista de la arquitectura, podríamos situar esta tecnología como una capa superior a las Servlets dentro de nuestra aplicación, ya que extiende la Especificación Servlet 2.1. Ambas tecnologías fueron desarrolladas originalmente por Sun Microsystem.

El funcionamiento general de la tecnología JSP es que el Servidor de Aplicaciones interpreta el código contenido en la página JSP para construir un Servlet, cuya salida será un documento estático (típicamente HTML) que se presentará en la pantalla del Navegador del usuario.



## Que es un SERVLET

Los Servlets son componentes del servidor. Estos componentes pueden ser ejecutados en cualquier plataforma o en cualquier servidor, debido a la tecnología Java que se usa para implementarlos. Los Servlets incrementan la funcionalidad de una aplicación web. Se cargan de forma dinámica por el entorno de ejecución Java del servidor cuando se necesitan. Cuando se recibe una petición del cliente, el contenedor/servidor web inicia el servlet requerido. El Servlet procesa la petición del cliente y envía la respuesta de vuelta al contenedor/servidor, que es enrutada al cliente (Allamaraju, 2000)

## FUNCIONAMIENTO JSP

**JSP** no se puede considerar un script al 100%, ya que antes de ejecutarse el Servidor de Aplicaciones compila el contenido del documento JSP (script y etiquetas) y genera una clase Servlet. Por lo tanto, se puede decir que aunque este proceso sea transparente para el programador no deja de ser una tecnología compilada.

La principal ventaja de **JSP** frente a otros lenguajes, es que permite integrarse con clases Java (.class), lo que permite separar en niveles las aplicaciones web, almacenando en clases java las partes que consumen más recursos (así como las que requieren más seguridad) y dejando la parte encargada de formatear el documento HTML en el archivo JSP. La idea fundamental detrás de este criterio, es el de separar la lógica del negocio de la presentación de la información.

Independientemente de la certeza de la aseveración, Java es conocido por ser un lenguaje muy portable (su lema publicitario reza: *escribelo una vez, córrelo donde sea*) y sumado a las capacidades de JSP se hace una combinación muy atractiva.

## VENTAJAS JSP

El uso de la tecnología JSP, tiene fuertes ventajas en las aplicaciones Web sobre otras tecnologías como ASP, PHP, etc. estas se mencionan a continuación. (Mendez)

- **Compatibilidad con servidores:** JSP sigue la filosofía de la arquitectura JAVA de "escribe una vez ejecuta donde quieras". JSP se puede ejecutar en los sistemas operativos y servidores web más populares, como por ejemplo: Apache Tomcat, JSWDK, Sun Java System Web Server, Macromedia JRun, ServletExec, etc.
- **Proceso de desarrollo abierto (open source):** El API JSP se beneficia de la extendida comunidad JAVA existente.
- **TAGS:** Los desarrolladores pueden crear nuevos tags y no depender tanto de los scripts.
- **Reusabilidad entre plataformas:** Los componentes JSP son reusables en distintas plataformas (UNIX, Windows).
- **La ventaja Java:** La tecnología JSP usa Java como lenguaje de Script. Java es un lenguaje muy potente y escalable que los lenguajes de Script. Las páginas JSP son compilados en Servlets por lo que actúan como una puerta a todos los servicios Java de Servidor y librerías Java para aplicaciones http. Java hace el trabajo del desarrollador más fácil p. e. ayuda a proteger el sistema contra las "caídas, también ayuda en el manejo de la memoria protegiendo contra fallos de memoria y el duro trabajo de buscar los fallos de pérdida de punteros de memoria que pueden hacer mas lento el funcionamiento de una aplicación.
- **Mantenimiento** Las aplicaciones que usan JSP tiene un fácil mantenimiento.
  - Los lenguajes de Script están bien para pequeñas aplicaciones, pero no encajan bien para aplicaciones grandes.
  - Java es un lenguaje estructurado y es más fácil de construir y mantenimientos grandes como aplicaciones modulares.
  - La tecnología JSP hace mayor énfasis en los componentes que en los Scripts, esto hace que sea más fácil revisar el contenido sin que afecte a la lógica o revisar la lógica sin cambiar el contenido.

- La arquitectura EJB encapsula la lógica de p. e.: acceso a BD, seguridad, integridad transaccional y aislamiento de la aplicación.
- Debido a que la tecnología JSP es abierta y multiplataforma, los servidores web, plataformas y otros componentes pueden ser fácilmente actualizados o cambiados sin que afecte a las aplicaciones basadas en la tecnología JSP.

## CONCLUSIÓN

Al indagar en el uso de JSP para el proyecto en cuestión, podemos afirmar que el uso de esta tecnología nos permite manipular las clases del Lenguaje Orientado a Objetos (JAVA). Por lo tanto, el núcleo del sistema o la parte más importante que requiere de un diseño acabado por la complejidad puede ser implementada en JAVA y por otra parte la interfaz con el usuario la podemos manejar con JSP. Además, JSP provee fuertes ventajas sobre otras tecnologías de programación Web, por lo que el uso de ella es apropiada para este proyecto, cabe recordar que lo que se busca es el manejo de la Tecnología Orientada a Objetos al 100%.

En efecto, el uso de una base de datos orientada a objetos es mejor que interactúe directamente con un lenguaje de programación que se caracterice por ser orientado a objetos. JAVA es una tecnología apropiada, por lo tanto el uso de Java Server Pages nos permitirá tener una interfaz Web apropiada con el usuario.

## BIBLIOGRAFÍA

**Allamaraju.** 2000. *Professional Java Server Programming J2EE Edition.* s.l. : Wrox Press, 2000.}

**Java.** 2007. Java. [En línea] 2007. [Citado el: 30 de Julio de 2007.]  
<http://www.java.com/es/about/>.

**Mendez, Natxo.** DesarrolloWeb. [En línea] [Citado el: 1 de Agosto de 2007.]  
<http://www.desarrolloweb.com/articulos/832.php>.

**ANEXO 5**  
**DOCUMENTO DE APOYO NETBEANS**

## **INTRODUCCIÓN**

El presente documento de apoyo, intentará apoyar la decisión de implementar el proyecto “Sistema Orientado a Objetos para Biblioteca del Seminario Metropolitano Concepción” con la herramienta de desarrollo para aplicaciones NetBeans.

## QUE ES NETBEANS.

NetBeans es un proyecto de código abierto de gran éxito con una gran base de usuarios, una comunidad en constante crecimiento. Sun MicroSystems fundó el proyecto de código abierto NetBeans en junio 2000 y continúa siendo el patrocinador principal de los proyectos.

El NetBeans IDE es un entorno de desarrollo - una herramienta para programadores pensada para escribir, compilar, depurar y ejecutar programas. Está escrito en Java - pero puede servir para cualquier otro lenguaje de programación. Existe además un número importante de módulos para extender el IDE NetBeans. El IDE NetBeans es un producto libre y gratuito sin restricciones de uso. (NetBeans)

## HISTORIA

NetBeans comenzó como un proyecto estudiantil en República Checa (originalmente llamado Xelfi), en 1996 bajo la tutoría de la Facultad de Matemáticas y Física en la Universidad de Charles en Praga. La meta era escribir un entorno integrado de desarrollo (IDE) para Java parecida a la de Delphi. Xelfi fue el primer entorno integrado de desarrollo escrito en Java, con su primer pre-release en 1997.

Xelfi fue un proyecto divertido para trabajar, ya que las IDEs escritas en Java eran un territorio desconocido en esa época. El proyecto atrajo suficiente interés, por lo que los estudiantes, después de graduarse, decidieron que lo podían convertir en un proyecto comercial. Prestando espacios web de amigos y familiares, formaron una compañía alrededor de esto. Casi todos ellos siguen trabajando en NetBeans. (Wikipedia)

## PLATAFORMA NETBEANS

Durante el desarrollo del NetBeans IDE ocurrió una cosa interesante. La gente empezó a construir aplicaciones usando el NetBeans core runtime con sus propios plug-ins, de hecho, esto se convirtió en un mercado bastante grande.

**La Plataforma NetBeans** es una base modular y extensible usada como una estructura de integración para crear aplicaciones de escritorio grandes. Empresas independientes asociadas, especializadas en desarrollo de software, proporcionan extensiones adicionales que se integran fácilmente en la plataforma y que pueden también utilizarse para desarrollar sus propias herramientas y soluciones.

## FUNCIONES DE LA PLATAFORMA NETBEANS

Con la plataforma NetBeans, los desarrolladores pueden centrarse en las partes importantes de una aplicación: la lógica de negocios que hace que la aplicación sea única. El resultado es un ahorro inmenso de tiempo y esfuerzo. Algunas de las funciones de la plataforma son:

- **Administración de la interfaz de usuario:** La plataforma proporciona las ventanas, menús, barras de herramientas y otros componentes de presentación. Los desarrolladores escriben acciones y componentes que administra el sistema, lo que permite ahorrar tiempo y generar un código más limpio y con menos errores.
- **Administración de datos y presentación:** La plataforma NetBeans contiene un completo conjunto de herramientas para presentar datos al usuario y manipularlos.
- **Administración de configuración:** El almacenamiento y la restauración de configuraciones, incluso objetos de negocios complejos, es segura, simple, transparente y, en muchos casos, automática.
- **Edición gráfica:** La creación de vistas gráficas de datos arrastrando elementos es fácil y rápida con la biblioteca de elementos gráficos.
- **El editor:** Disponible como una extensión de la plataforma, las aplicaciones generadas con NetBeans pueden usar el Editor de NetBeans, un conjunto de herramientas avanzadas y extensibles para generar editores personalizados.
- **Infraestructura de asistente:** Conjunto de herramientas para generar con facilidad asistentes extensibles y de fácil uso que guían a los usuarios en las tareas más complejas.
- **Administración de almacenamiento:** Una abstracción del acceso a datos basado en archivos. Los "archivos" del sistema NetBeans pueden ser archivos locales o remotos, por ejemplo, de un servidor FTP, depósito CVS, archivo XML o base de datos. La ubicación de almacenamiento de los datos es totalmente transparente para otros módulos que los utilizan.
- **Gran cantidad de componentes adicionales:** como control de versión, editores especializados, componentes de interfaz de usuario especializados, acceso a datos remotos mediante FTP y otros sistemas, y métodos óptimos para trabajar con diferentes tecnologías Java y de Internet; que están disponibles como módulos adicionales que se pueden usar en cualquier aplicación.

- **Actualizaciones por Internet:** Las aplicaciones basadas en NetBeans pueden usar la tecnología Java Web Start para transferir conjuntos de módulos personalizados según la función de un usuario, para aplicaciones complejas. Una aplicación Web Start siempre está actualizada y combina las ventajas de la administración y ejecución centralizadas con los beneficios de un cliente completo para el usuario. Para las aplicaciones que no sean WebStart, se puede usar un componente optativo, el módulo AutoUpdate, que descarga actualizaciones o nuevas funciones a través de Internet. (NetBeans)

## POR QUE NETBEANS

**Integración de múltiples herramientas y protocolos:** Sorprende poco que las extensas posibilidades de NetBeans para el desarrollo multiplataforma estén atrayendo a numerosos programadores que habían estado empleando otras herramientas. Los acostumbrados a trabajar con herramientas basadas en la tecnología Java, como JBuilder de Borland, están descubriendo que migrar a NetBeans puede aligerar significativamente sus esfuerzos de desarrollo. (Sun)

**Facilidad de empleo a lo largo de todo el ciclo de vida del desarrollo:** La facilidad de uso de NetBeans también resulta seductora porque no hay necesidad de ir más allá del portal netbeans.org en busca de características adicionales y de añadidos, los cuales se encuentran en paquetes de características que cubren un amplio rango de requisitos, desde C/C++ a la movilidad en el Web. Una vez descargado, NetBeans es fácil de emplear gracias a su interfaz de usuario y sus características, tales como el recientemente anunciado Visual Web Pack (VWP). Con VWP, los componentes JavaServer Faces pueden ser arrastrados y soltados para establecer propiedades y crear código para los manejadores de eventos del lado del servidor. (Sun)

**Soporte al modelado mejora la productividad del desarrollador:** Mientras los desarrolladores están descubriendo como el extenso conjunto de herramientas incluido en NetBeans les facilita el diseño de aplicaciones, los que toman las decisiones corporativas han podido ver que el IDE mejora la productividad gracias a la eficiencia de las características incorporadas. Por ejemplo, la última versión de NetBeans permite el desarrollo de aplicaciones multiplataforma al incluir las características de productividad de las que anteriormente sólo se disponía en Sun Studio Enterprise. (Sun)

El soporte beta para el Unified Modeling Language (UML) es otro aspecto de NetBeans que convierte el desarrollo de aplicaciones Java para múltiples plataformas en algo rápido y eficiente

# INSTALACIÓN

## EN WINDOWS

1. Después de haber bajado el instalador, dar doble clic sobre el icono para iniciar el instalador.
2. Después aparecerá el Acuerdo de Licenciamiento (License Agreement), acéptelo y de clic en Siguiente (Next).
3. Especificar un directorio vacío donde instalar Netbeans IDE 5.5. Dar clic en Siguiente (Next).
4. Selecciones el JDK que se quiere usar junto con el NetBeans IDE de la lista de selección si hay más de uno. Dar clic en Siguiente (Next).
5. Verifique que el directorio de instalación es el correcto y que se cuenta con espacio suficiente en el sistema para la instalación.
6. De clic en Siguiente (Next) para iniciar la instalación. (Wikipedia)

## EN LINUX

1. Navegue hacia el directorio que contiene el instalador.
2. Si es necesario, cambie los permisos del instalador para hacerlo ejecutable escribiendo lo siguiente en la línea de comando (netbeans\_installer es el nombre del instalador de NetBeans que se bajó): \$ chmod +x netbeans\_installer
3. Ejecute el instalador escribiendo lo siguiente en la línea de comando (De nuevo, netbeans\_installer es el nombre del instalador de NetBeans que se bajó): \$ ./netbeans\_installer
4. Después aparecerá el Acuerdo de Licenciamiento (License Agreement), acéptelo y de clic en Siguiente (Next).
5. Especificar un directorio vacío donde instalar Netbeans IDE 5.5. Dar clic en Siguiente (Next).
6. Selecciones el JDK que se quiere usar junto con el NetBeans IDE de la lista de selección si hay más de uno. Dar clic en Siguiente (Next).
7. Verifique que el directorio de instalación es el correcto y que se cuenta con espacio suficiente en el sistema para la instalación.
8. De clic en Siguiente (Next) para iniciar la instalación. (Wikipedia)

## CONCLUSIÓN

NetBeans es la herramienta que se necesita para desarrollar el proyecto “Sistema Orientado a Objetos para Seminario Metropolitano Concepción”, por sus variadas funcionalidades y ventajas como por ejemplo la gran cantidad de componentes que incluye, la facilidad de uso y el soporte de la tecnología JAVA en su totalidad (editor, compilación, edición gráfica) además de proporcionar el soporte a JSP. En definitiva ésta herramienta es apropiada para la programación Orientada a Objetos que es lo que busca el proyecto.

## BIBLIOGRAFÍA

- NetBeans.** NetBeans. [En línea] [Citado el: 18 de Agosto de 2007.]  
[http://www.netbeans.org/index\\_es.html](http://www.netbeans.org/index_es.html).
- Sun.** Sun Microsystems. [En línea] [Citado el: 18 de Agosto de 2007.]  
[http://www.sun.com/emrkt/innercircle/newsletter/spain/0207spain\\_feature.html](http://www.sun.com/emrkt/innercircle/newsletter/spain/0207spain_feature.html).
- Wikipedia.** Wikipedia. [En línea] [Citado el: 17 de Agosto de 2007.]  
[http://es.wikipedia.org/wiki/NetBeans#NetBeans\\_IDE](http://es.wikipedia.org/wiki/NetBeans#NetBeans_IDE).