

Universidad de las Ciencias Informáticas

Facultad 1



**Solución para la diseminación selectiva de
información del Sistema Integrado de Gestión
Bibliotecaria de la Universidad de las Ciencias
Informáticas.**

**Trabajo de Diploma para optar por el título de Ingeniero en
Ciencias Informáticas.**

Autor: Reinier Antonio Díaz Valdés.

Tutor: Ing. Javier Heredia Ruiz
Asesor: Ing. Adnier Roselló Carrazana.

**Ciudad de La Habana, Junio 2012.
“Año 54 de la Revolución”.**

"Milito en el bando de los impacientes, milito en el bando de los apurados, de los que siempre presionan para que las cosas se hagan y de los que muchas veces tratan de hacer más de lo que se puede."

Fidel Castro

"Si una persona es perseverante, aunque sea dura de entendimiento, se hará inteligente; y aunque sea débil se transformará en fuerte."

Leonardo Da Vinci

DECLARACIÓN DE AUTORÍA

Declaro que soy el único autor de este trabajo y autorizo a la Facultad 1 de la Universidad de las Ciencias Informáticas a hacer uso del mismo en su beneficio.

Para que así conste firmo la presente a los ____ días del mes de _____ del año _____.

Reinier Antonio Díaz Valdés

Firma del Autor.

Ing. Javier Heredia Ruiz

Firma del Tutor.

Ing. Adnier Roselló Carrazana

Firma del Asesor.

OPINIÓN DEL USUARIO DEL TRABAJO DE DIPLOMA

El Trabajo de Diploma, titulado “Solución para la diseminación selectiva de información del Sistema Integrado de Gestión Bibliotecaria de la Universidad de las Ciencias Informáticas”, fue realizado en la Universidad de las Ciencias Informáticas. Esta entidad considera que, en correspondencia con los objetivos trazados, el trabajo realizado le satisface:

- Totalmente
- Parcialmente en un _____ %

Los resultados de este Trabajo de Diploma le reportan a esta entidad los beneficios siguientes (cuantificar):

Como resultado de la implantación de este trabajo se reportará un efecto económico que asciende a <valor en MN o USD del efecto económico>

Y para que así conste, se firma la presente a los _____ días del mes de _____ del año _____.

Representante de la entidad Cargo

Firma Cuño

Agradecimientos

A la Revolución, por la confianza que depositó en mí cuando me permitió formar parte del maravilloso proyecto de la UCI e hizo realidad mis sueños. Nunca lo olvidaré.

A mis padres, a ti papá que diste todo por verme crecer y convertirme en un hombre, nunca olvidaré tu abrazo, tu amor; a ti mamá que eres la flor más hermosa que he conocido en mi vida, gracias por estar siempre ahí cuando te necesité. Los amo.

A mi hermano, por ser un verdadero ejemplo de entrega, dedicación, firmeza y honestidad, gracias por ser mi verdadero amigo, seguir tus pasos es un honor para mí. Te quiero mi Bro.

A mi maravillosa familia, a mi cuñada Merfys, a mis sobrinas, a mis abuelas, gracias a todos ustedes por el cariño y la confianza depositada en mí. Los quiero mucho.

A mi familia de la Habana, Virgilio, Mayra, Yipsy y Alessandro, con ustedes aprendí que el amor se manifiesta de muchas maneras, gracias por acogerme como un hijo, hermano y tío. Se merecen lo mejor de este mundo.

A mi novia, por estar a mi lado en los momentos malos y buenos de mi carrera, espero que estos no sean los últimos y que compartamos muchas alegrías y tristezas juntos. Siempre serás mi Yito.

A mis tutores, Javier y Adnier, también a Edisnel por ser un líder maravilloso del que aprendí mucho, gracias por ser los guías en la realización de mis sueños. Siempre los llevaré en mi corazón.

A mi amigo Jeandy Bryan, gracias por la ayuda tan grande que me brindaste en la implementación del sistema, aprendí mucho en cada línea de código. Sinceramente gracias por tu dedicación.

A mis amigos, Javier, Octavio, Lena, Adriel, Jose, Héctor, Johan, con los que compartí momentos inolvidables y a todos los que no puedo mencionar porque la lista sería infinita. Gracias por todo.

A todos, muchas gracias.

Dedicatoria

Dedico especialmente este Trabajo de Diploma a mi papá, el destino no permitió que contara con su presencia en este importante momento de la vida. Donde quiera que estés, sirvan de homenaje a tu memoria los resultados alcanzados en mi carrera y te doy las gracias por nunca dejar que claudicara ante nada en la vida.

A las personas más maravillosas que he conocido: mi mamá, mi hermano, Virgilio, Mayra y mi novia Yisel.

A mis amigos y compañeros, y a todos aquellos que de una forma u otra hicieron posible la realización de este sueño.

Resumen

Las instituciones bibliotecarias constituyen pilares en el atesoramiento del conocimiento universal del hombre. La evolución de las llamadas Tecnologías de la Informática y las Comunicaciones (TIC) ha llegado a todos los procesos con los que el ser humano interactúa a diario, las bibliotecas constituyen puntos de referencia y claros ejemplos de este vertiginoso avance. La biblioteca de la Universidad de las Ciencias Informáticas es hoy una entidad automatizada gracias a una solución informática que se conoce como Sistema Integrado de Gestión Bibliotecaria Koha. Actualmente, no cuenta con el servicio de diseminación selectiva de información, lo que constituye el objetivo principal de la presente investigación, la incorporación del servicio aumentaría las funcionalidades que están disponibles y la rapidez, eficiencia y facilidad de la experiencia del usuario con el sistema. Para la implementación de la solución se utiliza la metodología de desarrollo RUP, el lenguaje de modelado UML y la herramienta CASE Visual Paradigm. Las tecnologías que serán utilizadas son el servidor de aplicaciones web Apache, como plataforma de desarrollo el lenguaje de programación script Perl, y MySQL como gestor de base de datos relacional. Esta solución brindará la posibilidad de enviar información selectivamente a los usuarios utilizando sus propias necesidades informativas, recogidas en un perfil individual y configurable, a través del correo electrónico. Además, permitirá conocer rápidamente de nuevos registros bibliográficos adquiridos sin la necesidad de visitar física o virtualmente la institución.

Palabras clave:

Bibliotecas, Sistema Integrado de Gestión Bibliotecaria, servicio de diseminación selectiva de información, perfil.

Índice de contenidos

Resumen.....	VI
Introducción.....	1
Capítulo 1. Fundamentación teórica.....	7
1.1. Introducción al Capítulo.....	7
1.2. Sistemas Integrados de Gestión Bibliotecaria. Antecedentes.....	7
1.2.1. Conceptualización de los Sistemas Integrados de Gestión Bibliotecaria.....	8
1.2.2. Características de los Sistemas Integrados de Gestión Bibliotecaria en general.....	10
1.3. Sistema Integrado de Gestión Bibliotecaria Koha.....	11
1.3.1. Surgimiento de Koha.....	12
1.3.2. Características de Koha.....	12
1.3.3. Koha como sistema integrado, integrable y adaptable.....	13
1.3.4. Orientado a la WEB y multiplataforma.....	14
1.3.5. Experiencias de otras instituciones.....	14
1.4. Diseminación selectiva de información. Antecedentes.....	16
1.4.1. Conceptualización de diseminación selectiva de información.....	16
1.4.2. Objetivos y funciones de la diseminación selectiva de información.....	18
1.4.3. Características generales de la diseminación selectiva de información.....	20
1.4.4. La diseminación selectiva de información en la actualidad.....	20
1.4.5. La diseminación selectiva de información en Cuba.....	23
1.5. Lenguajes, tecnologías y herramientas a utilizar.....	24
1.5.1. Lenguaje de Programación Perl.....	25
1.5.2. Lenguaje de Marcas de Hipertexto: HTML.....	26
1.5.3. Lenguaje de Programación del Lado del Cliente: JavaScript.....	26
1.5.4. Hojas de Estilo en Cascada: CSS.....	27
1.5.5. Tecnología CGI.....	27
1.5.6. Módulos de Perl.....	28
1.5.6.1. Módulo para el soporte de CGI.....	28
1.5.6.2. Módulo HTML::Template.....	28
1.5.6.3. Módulo DBI.....	29
1.5.6.4. Módulo DBD.....	29
1.5.7. Sistema Gestor de Base de Datos: MySQL.....	29
1.5.8. Servidor Web Apache.....	30
1.5.9. Lenguaje UML.....	31
1.5.10. Herramienta CASE: Visual Paradigm for UML.....	31
1.5.11. Entorno de Desarrollo Integrado Eclipse.....	32
1.6. Metodología de desarrollo adoptada.....	33
1.7. Fundamentación y análisis de la propuesta de solución.....	33

1.8. Conclusiones del Capítulo.....	34
Capítulo 2. Características del sistema.....	35
2.1. Introducción al Capítulo.....	35
2.2. Entorno del sistema.....	35
2.3 Objeto de automatización.....	36
2.3.1. Flujo de eventos automatizados.....	36
2.4. Modelo de dominio.....	36
2.4.1. Glosario de términos del dominio.....	38
2.5. Requisitos funcionales del sistema.....	39
2.6. Requisitos no funcionales del sistema.....	39
2.7. Definición de los casos de uso.....	40
2.7.1. Actores del Sistema.....	41
2.7.2. Listado de casos de uso del sistema.....	41
2.7.3. Diagrama de Casos de Uso.....	41
2.7.4. Patrón de casos de uso CRUD.....	42
2.8. Descripciones de casos de uso.....	43
2.9. Conclusiones del Capítulo.....	45
Capítulo 3. Análisis y diseño de la propuesta de solución.....	46
3.1. Introducción al Capítulo.....	46
3.2. Análisis.....	46
3.2.1. Modelo de Análisis.....	46
3.2.2. Diagrama de clases del análisis.....	47
3.3. Diseño.....	47
3.3.1. Diagrama de colaboración.....	48
3.3.2. Diagrama de secuencia.....	48
3.3.3. Diagrama de clases del diseño.....	49
3.4. Modelo de Datos.....	50
3.4.1. Diagrama Entidad-Relación.....	51
3.5. Arquitectura del sistema.....	52
3.6. Concepción de la Ayuda.....	53
3.7. Conclusiones del Capítulo.....	53
Capítulo 4. Implementación y Pruebas.....	54
4.1. Introducción al Capítulo.....	54
4.2. Diagrama de despliegue.....	54
4.3. Modelo de Implementación.....	55
4.3.1. Diagrama de Paquetes.....	56
4.3.2. Diagrama de componentes.....	57
4.4. Modelo de Pruebas.....	63
4.4.1. Resultados de las pruebas.....	64
4.5. Conclusiones del Capítulo.....	69

Conclusiones.....	70
Recomendaciones.....	71
Referencias bibliográficas.....	72
Bibliografía consultada.....	76
Glosario de términos.....	78
Anexos.....	80
Anexo 1. Formulario para el establecimiento de un perfil de usuario para DSI.	80
Anexo 2. Plantilla del servicio DSI que existía en la UCI.....	82
Anexo 3. Descripciones detalladas de casos de uso.	83
Anexo 4. Guía para la configuración de una tarea programada utilizando el crontab de los sistema GNU/Linux.	90
Anexo 5. Imagen de la interfaz de usuario terminada. Módulo DSI.....	91
Anexo 6. Imagen de la notificación recibida por el usuario.....	92
Anexo 7. Ejemplo del Plan de Pruebas de Integración.	93

Índice de tablas

Tabla 1. Resumen de los sistemas analizados.....	22
Tabla 2. Actores del sistema.	41
Tabla 3. Descripción detallada del CU Gestionar autores.	45
Tabla 4. Descripción de los casos de prueba del CU Gestionar autores.	66
Tabla 5. Descripción de los casos de prueba del CU Seleccionar tipo de material.	66
Tabla 6. Descripción de los casos de prueba del CU Seleccionar estado.	67
Tabla 7. Descripción de los casos de prueba del CU Seleccionar periodicidad.	67
Tabla 8. Descripción de los casos de prueba del CU Elaborar notificación.....	68
Tabla 9. Descripción de los casos de prueba del CU Enviar notificación.	68
Tabla 10. Descripción detallada del CU Gestionar temas.	85
Tabla 11. Descripción detallada del CU Seleccionar tipo de material.	86
Tabla 12. Descripción detallada del CU Seleccionar estado.	87
Tabla 13. Descripción detallada del CU Seleccionar periodicidad.....	87
Tabla 14. Descripción detallada del CU Elaborar notificación.	88
Tabla 15. Descripción detallada del CU Enviar notificación.....	89
Tabla 16. Plan de Pruebas de Integración.	93

Índice de figuras

Figura 1. Diagrama de clases del modelo de dominio.	38
Figura 2. Diagrama de casos de uso del sistema.	42
Figura 3. Diagrama de clases del análisis del CU Gestionar autores.	47
Figura 4. Diagrama de colaboración del CU Gestionar autores.	48
Figura 5. Diagrama de secuencia del CU Gestionar autores.	49
Figura 6. Diagrama de clases del diseño del CU Gestionar autores.	50
Figura 7. Diagrama Entidad-Relación.	51
Figura 8. Descripción del Patrón MVC.	53
Figura 9. Diagrama de despliegue.	55
Figura 10. Diagrama de paquetes general.	56
Figura 11. Diagrama de paquetes del módulo DSI.	57
Figura 12. Diagrama de componentes Visión de la Arquitectura.	58
Figura 13. Diagrama de componentes del subsistema Vista.	59
Figura 14. Diagrama de componentes del subsistema Control.	60
Figura 15. Diagrama de componentes del subsistema Modelo.	61
Figura 16. Diagrama de componentes del subsistema Módulo DSI.	62
Figura 17. Gráfico de resumen de las pruebas.	65
Figura 18. Ejemplo de formulario para servicio DSI.	81
Figura 19. Muestra de la plantilla del servicio DSI que existía en la UCI.	82
Figura 20. Interfaz de usuario del módulo DSI.	91
Figura 21. Notificación recibida por el usuario.	92

Introducción

Desde tiempos remotos, el hombre garantizaba su supremacía como ser social y civilizado sobre todas las especies del planeta; esta condición fue posible gracias a la capacidad de poseer conciencia de su propia existencia, es decir, estar consciente de sí mismo y de su entorno. El ser humano incrementa y desarrolla otras importantes capacidades, las cuales son la base de nuestra civilización: vivir en comunidad y transmitir todos los conocimientos a las generaciones futuras, se convierten en tareas comunes en su devenir primitivo. La humanidad comprendió desde entonces la necesidad de encontrar la forma de preservar y transmitir su cultura, sus creencias y conocimientos tanto en el espacio como en el tiempo.

Las señales gestuales fueron la primera forma de expresar y transmitir mensajes, por lo que constituye el inicio de la transmisión de información de un individuo a otro. Las pinturas rupestres son ejemplos de cómo el hombre encontraba poco a poco un medio por el cual podría plasmar conocimientos perdurables para sus generaciones posteriores; son el más antiguo precedente de los primeros documentos impresos de los que se tiene memoria.

En búsqueda del medio idóneo para guardar sus conocimientos, el hombre desarrolló la escritura. Su invención ocurrió en varios lugares del mundo de forma independiente. Las primeras técnicas de escritura se remontan al 4000 a.C. Aparecieron en Eurasia, Oriente Medio, Egipto y China probablemente de manera independiente.

Enlazando la escritura con su portador universal, fue en China donde se fabricó el papel por primera vez, ellos transmitieron este conocimiento a los árabes, quienes a su vez lo llevaron a otras regiones del continente Europeo. En el antiguo Egipto se escribía sobre papiro (de donde proviene la palabra papel). En Europa, durante la Edad Media, se utilizó el pergamino que consistía en pieles de cabra o de carnero curtidas, preparadas para recibir la tinta, que por desgracia era bastante costoso, lo que ocasionó que a partir del siglo VIII se popularizara la infausta costumbre de borrar los textos de los pergaminos para reescribir sobre ellos, por lo que se perdió una cantidad inestimable de obras.

De este proceso surge el planteamiento que supone: por un lado, determinar la forma de garantizar la integridad intelectual del contenido de la obra y la conservación del soporte en el que fue plasmada, y por otro, encontrar el medio por el cual se mantendrá inalterada la intención o finalidad para la cual se concibió. El libro se consagra como resultado de esta idea y surge como material indispensable para plasmar el conocimiento. Su elaboración se realiza en una primera etapa completamente a mano, se

convirtió en un proceso costoso en cuanto a tiempo y recursos humanos, solo accesible en templos religiosos o en las entonces llamadas bibliotecas.

Las bibliotecas nacieron en los templos de las ciudades mesopotámicas, donde tuvieron en principio una función conservadora de registros de hechos ligados a la actividad religiosa, política, económica y administrativa, al servicio de una casta de escribas y sacerdotes. El período helenístico aportó el nacimiento de grandes bibliotecas legendarias, como la Biblioteca de Alejandría o la Biblioteca de Pérgamo, creadas con la voluntad de reunir todo el conocimiento social de su tiempo y ponerlo a disposición de los eruditos.

La invención de la imprenta en el Renacimiento y posteriormente las revoluciones francesa y norteamericana, iniciaron la extensión por Europa y América de una verdadera voluntad de hacer accesible la cultura y la educación para todos. En el mundo de las bibliotecas, esto supuso el nacimiento de una fiebre desamortizadora que se extendió por todo el continente y que transfirió a la sociedad un inmenso tesoro bibliográfico procedente de las antiguas instituciones, particularmente de la iglesia católica. Pese a todo, este deseo de acercar la cultura a toda la sociedad no consiguió hacerse realidad hasta mediados del siglo XIX, con la aparición en el mundo anglosajón de la biblioteca pública.

Las bibliotecas han jugado un papel crucial en todo el desarrollo de la sociedad, son un instrumento significativo en el proceso educativo, laboral y social. Logran satisfacer las necesidades de información, educación, investigación y ocio, así como facilitan dicha documentación a la comunidad.

Con la llegada de la “Era de la Informática” y las innovaciones tecnológicas conocidas como las Tecnologías de la Información y las Comunicaciones, las bibliotecas han adquirido nuevas funciones dentro de la sociedad de la información. El creciente poder que han alcanzado estas tecnologías ocupa un papel fundamental en la comunicación del conocimiento, y con ello, el desarrollo de diferentes medios electrónicos que ayuden a adquirir información, para que una vez recopilada y organizada pueda ser accedida por la sociedad. Gracias a estos medios se han podido informatizar diferentes esferas que posibilitan el análisis de grandes volúmenes de datos almacenados que satisfacen la gran oferta y demanda social de información.

El surgimiento de recientes soportes del entendimiento, ha transformado el papel tradicional de las bibliotecas de simples almacenes de conocimientos a puertas abiertas a la comunicación de toda la

comunidad, donde la información no solo se presentará en soporte de papel, sino que aumentará aceleradamente en formato electrónico.

Teniendo en cuenta la importancia que se le concede a nivel mundial al manejo de la información de las bibliotecas y que la acelerada automatización de las mismas se debe fundamentalmente a que el usuario demanda mayores prestaciones, se transita del simple uso de aplicaciones informáticas individuales, al uso de sistemas integrados. Estos sistemas deben permitir gestionar todas las funciones y servicios de la biblioteca de forma automatizada, de ahí que se le denomine Sistemas Integrados de Gestión Bibliotecaria.

Con el arribo de esta revolución tecnológica, las bibliotecas permiten al usuario estar informado de las novedades más recientes publicadas en su espacio. Esta necesidad de mantener alertado al usuario sobre los documentos que se ingresan, además de brindar información científica actualizada de productos en formato electrónico y que van a corresponder a los intereses específicos de acuerdo a su actividad concreta, fue tema de investigación del informático alemán Hans Peter Luhn [Gutiérrez Vallado, 2008].

Luhn, en sus investigaciones se centró principalmente en darle solución a los problemas derivados del almacenamiento y recuperación de la información en bibliotecas y centros de información. Investigó acerca de un servicio especializado dentro del quehacer bibliotecario que se va a encargar de divulgar la información de manera selectiva, prioritaria y que cubra las necesidades de los usuarios a los que va dirigida, a esta investigación la nombró diseminación selectiva de información [Gutiérrez Vallado, 2008].

En Cuba, en la Universidad de las Ciencias Informáticas, se desarrolla un Sistema Integrado de Gestión Bibliotecaria a cargo del proyecto “Biblioteca” del Centro para la Informatización Universitaria (CENIA), el cual propone soluciones para bibliotecas que pretendan implantar un sistema con estas características. Bajo licencias GPL de software libre y código abierto, la aplicación está basada en Koha, pionero dentro de los sistemas integrados para la gestión bibliotecaria libres del mundo, el cual es adaptado en dependencia de las necesidades de la institución interesada. De esta manera, el proyecto presenta una solución óptima y adecuada para el funcionamiento de la biblioteca universitaria.

Los servicios brindados por el Sistema Integrado de Gestión Bibliotecaria de la Universidad de las Ciencias Informáticas pueden ser ampliados. El incremento en el uso del sistema conlleva a la necesidad de mejorar e incrementar sus funcionalidades para garantizar que los usuarios interactúen de la manera más usable y rápida posible con la institución bibliotecaria, aprovechando las ventajas del uso de la tecnología.

El usuario tiene necesariamente que navegar en el catálogo en línea de la institución para realizar búsquedas en los registros bibliográficos existentes o para conocer de los nuevos que se agregan. La dinámica en el incremento de la información que ocurre a diario en las bibliotecas y los demás temas que pueden tratar de manera individual cada una de ellas, pueden estar al alcance del usuario utilizando el principio del servicio de diseminación selectiva de información, de esta manera circula la información de la biblioteca al usuario dinámicamente y bajo sus propios criterios de selección. El funcionamiento de este servicio no se considera el más adecuado de acuerdo al avance tecnológico actual y a las necesidades de los múltiples usuarios que lo utilizan, por lo que se considera ineficiente, lo que trae consigo que no contribuye a elevar la satisfacción de los usuarios y no contrarresta las consecuencias de la carencia de funcionalidades.

De esta forma, no se satisfacen las necesidades de información de los usuarios a través de notificaciones enviadas por medio del correo electrónico. Esta funcionalidad no está presente en el sistema que se utiliza en la biblioteca de la universidad por lo que constituye un problema a resolver, encaminado a incrementar la operatividad y autonomía del Sistema Integrado de Gestión Bibliotecaria existente.

De la problemática anterior resulta el siguiente **problema de la investigación**:

¿Cómo lograr que los usuarios obtengan la información necesaria bajo sus propios criterios de selección como parte de nuevas funcionalidades que requiere el Sistema Integrado de Gestión Bibliotecaria de la Universidad de las Ciencias Informáticas?

Para enviar notificaciones a los usuarios según el problema anteriormente planteado, es necesario estudiar los diferentes componentes que integran el Sistema Integrado de Gestión Bibliotecaria de la Universidad de las Ciencias Informáticas. Por tanto, el **objeto de estudio** del presente trabajo se enmarca en los procesos relacionados con el servicio de diseminación selectiva de información, y se define como **campo de acción** los procesos del servicio de diseminación selectiva de información en el Sistema Integrado de Gestión Bibliotecaria de la Universidad de las Ciencias Informáticas.

Como **objetivo general de la investigación** se propone desarrollar con tecnologías libres, un módulo que permita la diseminación selectiva de información para el Sistema Integrado de Gestión Bibliotecaria de la Universidad de las Ciencias Informáticas que incorpore nuevas funcionalidades que permitan el envío de notificaciones a los usuarios según sus criterios definidos.

Se plantean los siguientes **objetivos específicos**:

- Realizar el marco teórico relacionado con los conceptos de Sistema Integrado de Gestión Bibliotecaria y el servicio de diseminación selectiva de información.
- Diseñar el módulo para la diseminación selectiva de información e implementar todas las funcionalidades.
- Validar los resultados obtenidos en el desarrollo del módulo para la diseminación selectiva de información del Sistema Integrado de Gestión Bibliotecaria de la Universidad de las Ciencias Informáticas.

Como **posibles resultados** de la investigación se proyecta la obtención de un módulo capaz de ejecutar las operaciones que se definan como requisitos funcionales, los cuales deberán, entre otras funciones, incorporar el principio del servicio de la diseminación selectiva de información al Sistema Integrado de Gestión Bibliotecaria de la Universidad de las Ciencias Informáticas, lo que permitirá a los usuarios bajo sus propios criterios de selección recibir la información más relevante dentro de su rama profesional, o en el marco de sus necesidades informativas de tipo científica, intelectual o simplemente para su esparcimiento y recreación.

Métodos Científicos.

Teóricos:

- Analítico–sintético: Para analizar la situación problemática existente y obtener las conclusiones de aquellas características que debe tener la solución propuesta, así como para analizar la documentación y particularidades de las herramientas existentes y llegar a la conclusión de cuáles son los más adecuados para el desarrollo.
- Análisis histórico–lógico: Para analizar la evolución histórica de las soluciones similares, las tendencias más recientes de las aplicaciones web y, sobre la base de esos datos, complementar las características necesarias o deseables en la solución propuesta.

Empíricos:

- Observación: Para realizar el estudio de las características, tendencias y expectativas de los usuarios para con las soluciones similares. Es necesario observar dichas soluciones, las interacciones con los usuarios y las opiniones emitidas acerca de las mismas.

El presente trabajo está estructurado en 4 capítulos y anexos, que incluyen toda la investigación sobre el servicio de diseminación selectiva de información, así como el desarrollo del módulo para contar con el mismo en el Sistema Integrado de Gestión Bibliotecaria de la biblioteca de la Universidad de las Ciencias Informáticas y las pruebas realizadas al sistema para comprobar su correcto funcionamiento. A continuación se hace una breve descripción de cada uno de los capítulos.

Capítulo 1. Fundamentación Teórica. En este capítulo se hace referencia a los principales conceptos relacionados con los Sistemas Integrados de Gestión Bibliotecaria y el servicio de diseminación selectiva de información, se realiza un estudio del estado del arte del mencionado servicio dentro de los sistemas para la gestión bibliotecaria y se analiza a Koha. Son expuestas además, la metodología, las tecnologías, los lenguajes y las herramientas utilizados para desarrollar la solución propuesta.

Capítulo 2. Características del Sistema. En este capítulo se describen los principales requisitos funcionales y no funcionales del módulo así como el entorno donde será implantado.

Capítulo 3. Análisis y diseño de la solución propuesta. En este capítulo se presenta la propuesta de solución mostrando elementos del diseño y la arquitectura del módulo. También se describe el modelo de datos que contiene las entidades necesarias para la persistencia de la información.

Capítulo 4. Implementación y Pruebas. En este capítulo se muestran el diagrama de despliegue y el diagrama de componentes como elementos fundamentales a modelar en la fase de implementación. Se describen los casos de prueba para cada caso de uso.

Capítulo 1. Fundamentación teórica.

1.1. Introducción al Capítulo.

El capítulo presenta de manera detallada los conceptos más importantes relacionados con los Sistemas Integrados de Gestión Bibliotecaria (SIGB) y el servicio de disseminación selectiva de información (DSI), a nivel internacional, nacional y en la Universidad de las Ciencias Informáticas (UCI). En él se tratarán los principales conceptos en los que se enmarca el dominio del problema, los SIGB y el servicio DSI, sus antecedentes, las principales definiciones, objetivos, funciones, utilidad y características generales; se expone el análisis de la solución propuesta en la investigación para la adición del servicio DSI al SIGB de la biblioteca de la UCI. Además, se especifican las herramientas, lenguajes, tecnologías y metodología a usar en el desarrollo de la solución.

1.2. Sistemas Integrados de Gestión Bibliotecaria. Antecedentes.

En la llamada “Sociedad del Conocimiento” se aprecia una gran oferta y demanda social de información. Este fenómeno fue declarado por la IFLA en la Declaración de Glasgow de 2002, de manera que la biblioteca ha de adaptarse a la nueva sociedad y actuar como intermediaria del conocimiento, pensamiento y cultura [Gómez Hernández, 2001].

Debido a este y a otros motivos económicos, se ha visto la necesidad de que la biblioteca se entienda como un sistema: un conjunto de elementos en interacción dinámica para la consecución de ciertos objetivos en un entorno. La automatización se convierte entonces en una pieza clave para que la biblioteca consiga estos objetivos. Se tiende a automatizar al sistema completo de la biblioteca, no sólo tareas de catalogación automatizada por ejemplo, en la que es especialmente necesaria [Gómez Hernández, 2001].

En los años 80, nacen los Catálogos Públicos en Línea (OPAC, por sus siglas en inglés) y los primeros sistemas de automatización como Dobis/Libis, Aleph y Sabini. Desde los 90 a la actualidad, se ha alcanzado una madurez en esta disciplina y aparecen los Sistemas Gestores de Bases de Datos (SGBD) y posteriormente se desarrollan los grandes SIGB, que amplían la automatización a todas las tareas bibliotecarias. Con la llegada de Internet, se desarrollan grandes redes cooperativas que permiten compartir recursos y ofrecer servicios bibliotecarios en línea [García Melero, García Camarero, 1999].

Si se establece la pregunta de ¿por qué la automatización?, rápidamente se responde en lo fundamental con el hecho de que el usuario demanda mayores prestaciones por las exigencias de la sociedad en la que se mueve. De esta manera, la justificación del proceso de automatización se establece por las siguientes ideas:

- El colapso del sistema manual ante el gran volumen de información actual.
- La necesidad de agilizar procesos y optimizar los recursos, detectar las carencias y subsanarlas, revisar y reorganizar los procedimientos.
- La disposición de la información actualizada del funcionamiento de la biblioteca, (por ejemplo las estadísticas) que permitan una evaluación y se puedan implementar mejoras.
- La facilitación de la cooperación bibliotecaria, que comparta recursos y permita ofrecer mejores servicios con menos costos [García Melero, García Camarero, 1999].

Para conseguir todo ello, ha sido necesaria la normalización internacional de la información, realizada en:

- La descripción bibliográfica (mediante las ISBD).
- La ordenación de sus registros (con las directrices GARE y GSARE).
- Formatos de intercambio (como el MARC).
- Estándares de comunicación (como el protocolo TCP/IP; y el Z39.50 o ZING).
- La identificación del documento (con el ISBN, ISSN, BIBLID, DOI, entre otros.) [García Melero, García Camarero, 1999].

La automatización de bibliotecas no es meramente tecnología, implica un sistema interconectado para el trabajo en grupo. Aunque la complejidad tecnológica de la biblioteca electrónica pueda asustar en un principio, las mejoras en los programas de automatización optimizarán sin duda la manejabilidad de la información y los servicios al usuario [Saorín Pérez, 2002].

1.2.1. Conceptualización de los Sistemas Integrados de Gestión Bibliotecaria.

Se entiende por Sistema Integrado de Gestión Bibliotecaria a *“un conjunto organizado de recursos humanos que utilizan dispositivos y programas informáticos, adecuados a la naturaleza de los datos que deben procesar, para realizar procesos y facilitar los servicios que permiten alcanzar los objetivos de una institución bibliotecaria: almacenar de forma organizada el conocimiento humano contenido en todo tipo de materiales bibliográficos para satisfacer las necesidades informativas, recreativas y de investigación de los usuarios ”* [Saorín Pérez, 2002].

Es importante, en cualquier escenario tecnológico, la existencia de un modelo de validez general para el desarrollo de aplicaciones que incluya las estructuras de datos, las funcionalidades y los tratamientos. En este sentido, se conoce generalmente a las aplicaciones que tratan de ofrecer una solución completa de automatización a los procesos básicos de una biblioteca como SIGB. Por lo tanto, si definimos a la biblioteca como un *“un sistema de comunicación que pone en contacto la edición mundial con la comunidad a la que atiende mediante la realización de una serie de actividades, que requieren la aplicación de unos conocimientos, códigos y normas para la ejecución de los procesos y funciones en que se descomponen.”*, debemos encontrar una aplicación de gestión automatizada que abarque todos los aspectos implicados, basada en modelos de datos eficientes [Saorín Pérez, 2002].

En el ámbito de los SIGB el término integración indica multifuncionalidad, un sistema que recoge todas las funciones (módulos) necesarias para la gestión de una biblioteca. También se caracteriza porque exista una integración a nivel de datos, de manera que la información se almacena para un uso compartido y específico de cada módulo funcional. Los años 80 fueron el momento de los sistemas integrados, completos, centrados y únicos, surgidos como evolución de los sistemas monofuncionales que se usaban hasta finales de los años 70, estos tenían por objeto resolver el problema de la gestión mecánica de aquellas funciones que suponían un mayor costo en recursos humanos en las grandes bibliotecas (préstamos, gestión de catálogos, publicaciones periódicas) [Saorín Pérez, 2002].

Es necesario entender que los SIGB, en sus versiones comerciales, son aplicaciones de propósito general dentro del mundo bibliotecario, adaptables a cualquier tipo de biblioteca, por lo que a veces el proceso de personalización resulta bastante complejo. Los SIGB se han convertido en un tipo de aplicación, en el campo del mercado de las soluciones informáticas, independiente de la empresa desarrolladora, aunque es necesario resaltar que cada desarrollo enfoca la definición y agrupación de funciones de una manera particular, por lo que nos encontramos con un panorama heterogéneo [Saorín Pérez, 2002].

Luego de consultar y exponer anteriormente los conceptos que definen desde los puntos de vistas propios de sus autores a los SIGB, se define a un Sistema Integrado de Gestión Bibliotecaria como una herramienta informática compleja, que se integra de módulos o secciones con características propias y otras compartidas, los cuales hacen uso de una base de datos centralizada donde es almacenada toda la información y se destinan a automatizar cada proceso que participa en la gestión de una biblioteca, por lo cual se logra un sistema integrado que puede crecer según lo haga la

institución en donde esté desplegado y facilita la disposición a los usuarios de todo el conocimiento e información resguardado, de manera eficaz y eficiente.

1.2.2. Características de los Sistemas Integrados de Gestión Bibliotecaria en general.

Hoy existe una oferta variada de SIGB. Son desarrollados por empresas privadas que intentan, por los beneficios que aportan, adaptarse a las necesidades de cada biblioteca. Entre sus características comunes están el trabajo sobre una arquitectura cliente-servidor y que el ajuste a normas y estándares oficiales y de mercado.

Si se compara en conjunto todas las visiones existentes para modelar a los SIGB, se encuentran las suficientes similitudes para afirmar la existencia de un modelo conceptual bastante consensuado, el cual se puede resumir en los siguientes módulos:

- **Módulo de Selección y Adquisición:** Se encarga de los pedidos y la gestión económica y presupuestaria que conlleva.
- **Módulo de Catalogación:** Permite gestionar el catálogo, incluyendo control de autoridades. Los registros se crean directamente en MARC, o se importan de sistemas cooperativos, servidores Z39.50, entre otros.
- **El OPAC o Módulo de Recuperación de Información:** Permite recuperar información bibliográfica, (por diferentes puntos de acceso) e información no bibliográfica, (como la disponibilidad de ejemplares, reservas, entre otros.) Una nueva versión es el WebOpac que mantiene los atributos del OPAC tradicional, soporta hiperenlaces, permite recuperar recursos web, entre otros.
- **Módulo de Circulación:** Realiza el préstamo, devolución, renovación y reservas. En concreto, el préstamo inter-bibliotecario se ha beneficiado mucho de la automatización en la gestión y control de las transacciones.
- **Módulo de Publicaciones Periódicas:** Permite el control y adquisición de publicaciones periódicas que por sus características es más complejo de desarrollar. Esto contribuye al pobre desarrollo que posee, teniendo escasa integración con el vaciado de revistas y con las publicaciones electrónicas a texto completo.
- **Módulo de Estadísticas:** Produce estadísticas sobre los procesos, para establecer indicadores que ayuden a la gestión y toma de decisiones.

- Módulo de Administración y Gestión: Proporciona ayuda para la organización y gestión de tareas como la gestión de recursos humanos, técnicos, económicos, de mantenimiento, entre otros [Escobedo Molina, 2009].

La implementación de un SIGB puede regirse por tres aspectos fundamentales, a continuación se relacionan y se amplían brevemente en qué consiste cada tipo:

- Centralizado: El SIGB se utiliza para la gestión centralizada de la biblioteca, desde la administración de usuarios del sistema y el establecimiento de parámetros de datos, hasta la gestión de la adquisición y los procesos técnicos.
- Descentralizado: Se utiliza el módulo de administración para parametrizar el sistema pero la gestión se delega en las diferentes bibliotecas que componen el sistema de bibliotecas. Estas tendrán acceso a los módulos de gestión.
- Distribuido: En el SIGB no existe un único módulo de administración y la parametrización se realiza en cada uno de los diferentes módulos del sistema directamente [Escobedo Molina, 2009].

En la actualidad, muchas de las instituciones más importantes para el resguardo del legado de conocimientos que a lo largo de la historia el hombre garantizó para sus futuras generaciones, se encuentran operando bajo un SIGB. Las bibliotecas nacionales de muchos países ostentan resultados satisfactorios con el uso de sistemas integrados para su gestión, esto ha dado como resultado la existencia de varias compañías que producen software para bibliotecas, y además, la intención de utilizarlos en ellas es cada vez mayor. Universidades, bibliotecas públicas, archivos nacionales, y muchos otros, constituyen ejemplos claros de las ventajas que trae el uso de un SIGB para gestionar información y brindarla a su vez de manera rápida a los usuarios con intenciones investigativas, de desarrollo intelectual, esparcimiento y recreación.

1.3. Sistema Integrado de Gestión Bibliotecaria Koha.

La UCI necesita contar con un SIGB en su biblioteca, por lo tanto ha adaptado a sus necesidades uno que constituye un pionero en su campo en el mundo: Koha, un sistema ampliamente utilizado en otras bibliotecas universitarias e institucionales en países como Francia, Estados Unidos de América, Argentina y Uruguay, donde ha sido modificado a las necesidades de cada institución. Por sus características fue escogido por el proyecto “Biblioteca”, el cual tenía a su cargo la búsqueda, modificación, adaptación e implantación, para finalmente poner en funcionamiento el SIGB de la

biblioteca de la UCI. A continuación, la investigación relaciona toda la información acerca de Koha, su surgimiento como SIGB, sus características principales y específicas, así como ejemplos concretos de utilización satisfactoria en instituciones.

1.3.1. Surgimiento de Koha.

Koha fue creado en 1999 por Katipo Communications para la Horowhenua Library Trust (HLT) en Nueva Zelanda. Por cuestiones monetarias y la antigüedad del sistema usado, la adquisición de uno nuevo sería muy difícil, así Katipo Communications asumió la tarea de crearlo bajo la licencia GPL (Licencia Pública General del inglés General Public Licence), lo que aseguraba que otras bibliotecas pudieran beneficiarse del trabajo y también cooperar en el desarrollo futuro del sistema [Escobedo Molina, 2009].

Koha entre otras funciones, permite el uso de registros y catalogación utilizando el estándar internacional MARC y Z39.50, además de estar traducido de su inglés original a francés, chino, español y árabe. Permite al bibliotecario manejar la mayoría de los procedimientos administrativos de una biblioteca y surtir a los visitantes con un catálogo público para la consulta de ejemplares y la circulación de los mismos. Es desarrollado por una comunidad de programadores y bibliotecarios de todas partes del mundo y su diseño es ajeno a cualquier intención comercial o corporativa [Escobedo Molina, 2009].

Liblime, una compañía radicada en Ohio, Estados Unidos, ha estado agregando nuevas características al SIGB. La versión más reciente es la 4.08.23, el lanzamiento fue anunciado el 30 de noviembre del 2011 y tiene un tamaño de 20,65 megabytes. Sus nuevas características incluirán un nuevo módulo para las revistas, diseño de la interfaz gráfica, funciones avanzadas de búsqueda, corrección de múltiples tareas administrativas de impresión y reportes, además de otras mejoras generales [Filippi, 2008].

1.3.2. Características de Koha.

Koha es un SIGB bien desarrollado y avanzado que contiene gran cantidad de funcionalidades. Posee módulos para circulación, catalogación, adquisiciones, publicaciones seriadas, reservas, gestión de usuarios, relaciones entre centros pertenecientes a una institución y entre varias instituciones. Es un sistema multiplataforma, integrado, integrable y adaptable, que refleja una gran cantidad de funciones y servicios complementarios que se han incluido en cada una de las nuevas versiones. Es una aplicación basada en la web, que le da una fortaleza relevante en su gestión [Marquina, 2009].

Se caracteriza por contar con una base de datos dual, lo que constituye un poderoso instrumento para lograr la organización eficaz de la información. Esta característica permite que se aprovechen las fortalezas de los dos tipos de bases de datos empleadas y asegura que el sistema sea escalable, lo cual admite que soporte el volumen de transacciones de datos de cualquier biblioteca, sin importar el tamaño de los mismos. Las bases de datos relacionales son útiles para el tratamiento de la información estructurada, como los registros bibliográficos. Por su lado las bases de datos documentales se utilizan para organizar la información no estructurada como resúmenes, documentos a texto completo, entre otros. Estas bases de datos utilizan operadores booleanos, operadores relacionales y operadores de proximidad, que son empleados en los OPAC actuales [Marquina, 2009].

1.3.3. Koha como sistema integrado, integrable y adaptable.

Koha es un sistema integrado porque agrupa en uno mismo, todas las funciones gestionadas en una biblioteca. A través de su módulo OPAC es posible consultar el catálogo desde cualquier ubicación, solo se necesita un ordenador conectado a Internet y un navegador web. En la visita podrá saber si un ejemplar está prestado o disponible, también conocerá los últimos registros añadidos.

Además, el sistema es integrable, ya que es posible la configuración para que sea capaz de comunicarse con otros sistemas informáticos similares o de otra naturaleza. Permite la búsqueda e importación de registros bibliográficos a través del protocolo Z39.50 y la integración con el sistema informático de Amazon para la visualización de imágenes de libros [Filippi, 2008].

La adaptabilidad está presente en el sistema ya que permitió ser ampliado para el uso de todo tipo de bibliotecas. Actualmente es usado por bibliotecas académicas, especializadas, especiales, escolares y públicas. Para el caso de bibliotecas muy grandes es posible la utilización de una base de datos adicional para acelerar la búsqueda en catálogos con grandes cúmulos de datos. Permite además incluir colecciones que no están en la biblioteca central dentro de un mismo sistema y gestionarlos tan eficientemente como las colecciones propias, pues el sistema desde un principio ha incluido la gestión de sucursales.

Al utilizar los estándares y protocolos empleados por las bibliotecas, se asegura la interoperabilidad de este con otros sistemas y tecnologías, lo cual acentúa aún más su adaptabilidad:

- MARC21, UNIMARC.
- Z39.50 servidor y cliente, NCIP.
- Opensearch, RSS y LDAP.

- Indizadores y analizadores de datos potentes, para optimizar la recuperación de registros.
- Herramientas de acceso a SGBD y Deep linking.
- Interfaz personalizada de administración, reporte y catalogación.
- Linux, MySQL, Zebra, Perl, Apache [Escobedo Molina, 2009].

1.3.4. Orientado a la WEB y multiplataforma.

El sistema está basado en la web, toda su gestión es a través del ordenador por medio de un navegador web. Todas sus interfaces producen salidas compatibles con la especificación XHTML 1.0 y CSS de la World Wide Web Consortium (W3C), que garantiza su operatividad a través de cientos de navegadores web, plataformas, sistemas operativos y dispositivos no convencionales. Las bibliotecas no dependerán de diseños para la distribución o el establecimiento de tecnologías específicas ni dependerán de proveedores determinados a partir de lo diseñado, lo que les permite contratar servicios donde les sea más factible la inversión del presupuesto disponible, podrán utilizar ordenadores con diferentes sistemas operativos, así como para los bibliotecarios será posible realizar los préstamos desde cualquier computadora de la biblioteca [Escobedo Molina, 2009].

La interfaz web personalizada para la administración, los reportes y el proceso de catalogación permiten la búsqueda simple y fácil para todos los usuarios. Posibilita a los usuarios gestionar de manera independiente su perfil, es decir, el sistema trabaja con grupos de usuarios definidos, donde cada uno controla su propia forma de interactuar, así como la realización y gestión de reservas de recursos en línea o físicos con la misma herramienta. Permite además, la renovación de préstamos y la creación y compartición de listas de libros [Filippi, 2008].

1.3.5. Experiencias de otras instituciones.

Koha brinda un alto nivel de experiencia en su utilización. Otras instituciones que lo utilizan hace varios años avalan su desempeño, es un sistema robusto, estable, eficaz y eficiente. Basados en estas experiencias los especialistas designados para seleccionar el SIGB para la biblioteca de la UCI no hicieron caso omiso de sus resultados; si se toman en cuenta las características antes expuestas y se suman además otras razones como los resultados obtenidos por otras instituciones, sin dudas se establece el porqué de su selección.

Se cuentan con ejemplos de instituciones que han adaptado a Koha a sus necesidades, estas muestran niveles de satisfacción elevados desde el punto de vista de las opiniones de los usuarios, de

los administradores, bibliotecarios y trabajadores en general que tienen a su cargo la operación del SIGB, las Universidades Nacional de Cuyo y Nacional de Córdoba en Argentina, la Biblioteca Central de la Universidad Mayor de San Andrés en Bolivia y más de 140 bibliotecas registradas a nivel internacional avalan el desarrollo alcanzado por Koha y la eficiencia y facilidad que aporta a todos los factores que interactúan con el sistema.

Universidad Nacional de La Plata:

Hasta el año 2003 el servicio de bibliotecas de la facultad de ingeniería de la Universidad Nacional de La Plata (UNLP) estaba compuesto por una biblioteca central y 8 departamentales. Ambas contaban con grandes volúmenes de datos y la informatización de los catálogos y los servicios presentaban algunas deficiencias, debido a la diversidad que existía [Feus Pérez, 2009].

Se necesitaba de la modernización de las bibliotecas y los centros de información, así como de la incorporación y adopción de las nuevas tendencias de gestión. Con la aplicación de sistemas y herramientas (Koha y sus dependencias) de código abierto en la biblioteca de ingeniería de la UNLP se ha logrado alcanzar la total independencia del software comercial. El sistema implantado en la UNLP ha alcanzado un importante grado de éxito, gracias a que se lograron satisfacer las necesidades requeridas por la biblioteca [Feus Pérez, 2009].

Universidad OTR de Uruguay:

ORT es una universidad autónoma y la mayor universidad privada de Uruguay. Posee dos bibliotecas especializadas, ofrece servicios a más de 6000 estudiantes, docentes, graduados e investigadores. Ambas bibliotecas son especializadas y cuentan con más de 45000 volúmenes, 200 títulos de publicaciones periódicas especializadas en soporte impreso y más de 1400 videos y DVDs.

La universidad necesitaba para mejorar el servicio de las bibliotecas unificar el catálogo ya que estaba separado en dos. Era necesario disponer de un sistema integral de gestión para instalar un catálogo, realizar reservas en línea y migrar sus datos a un formato que estuviera acorde con los estándares internacionales. Escogieron a Koha como el sistema indicado para lograr sus metas. El sistema fue puesto positivamente en marcha y actualmente la biblioteca de la ORT cuenta con un sistema integrado completo que ha cubierto las expectativas de los usuarios y especialistas [Feus Pérez, 2009].

1.4 Disseminación selectiva de información. Antecedentes.

En el año 1947 Fleming planteaba que podía haber especialistas que reunían información de fuentes dispersas sobre determinado tema para poder hacer uso de ella, lo que podía traducirse como una necesidad que se presentaba y que requería de acciones por parte de los profesionales para proveer a los usuarios periódicamente sobre diversas fuentes de documentación [Gutiérrez Vallado, 2008].

Pero no es hasta 1958 que Hans Peter Luhn, ingeniero de la IBM, expresó la necesidad de que en las unidades de información se ofreciera un servicio de DSI de carácter formal y periódico, consideraba que debía ser automatizado con la finalidad de facilitar información personalizada a usuarios con intereses específicos. Luhn fue el primero en definir el término de DSI en su ensayo "*Business Intelligence System*" (Sistema de Inteligencia de Negocio) [Berovides Hernández, Galán Domínguez, CA 2009].

El primer servicio de este tipo se basó en el modelo de Luhn y fue implementado en el año 1959 en la División de Desarrollo de Sistemas Avanzados de la compañía IBM en Nueva York. A partir de este momento se comenzó a difundir este tipo de servicio en la industria y agencias del gobierno estadounidense [Berovides Hernández, Galán Domínguez, CA 2009].

Más tarde, en el año 1967, se comienza a distinguir un mayor interés al respecto por parte de los bibliotecarios. En la reunión de la American Society for Information Science (ASIS), se presentaron cinco trabajos relativos a la DSI y ya para 1968 noventa y seis sistemas se pusieron en práctica, algunos en bibliotecas universitarias y otros en bibliotecas públicas [Berovides Hernández, Galán Domínguez, CA 2009].

1.4.1. Conceptualización de disseminación selectiva de información.

En la literatura especializada que ha sido consultada, varios autores (Gutiérrez Vallado, 2008; Saide, 2003) coinciden en que Hans Peter Luhn fue quien utilizó por primera vez el término de Disseminación Selectiva de Información (DSI) en el año 1958, y lo definió como:

...el servicio (dentro de una organización) dedicado a canalizar nuevos elementos de información (a partir de cualquier fuente) a aquellos puntos dentro de la misma organización donde la probabilidad de su utilidad en el trabajo diario o de su interés sea elevada [Luhn, 1958; citado en Sánchez Pereyra, 2009].

Por su parte, Antonio Sánchez Pereyra puntualiza a la DSI como *“un procedimiento mediante el cual se suministra periódicamente a cada usuario o grupo de usuarios las referencias de los documentos que corresponden a sus intereses cognoscitivos. Este servicio puede proporcionarse a un usuario en especial basándose en un perfil de recuperación de información individual, o bien, a un conjunto de personas que tienen intereses comunes, requiriéndose para tal caso la de un perfil colectivo, por lo que la eficacia de este servicio depende, sobre todo, de una adecuada definición del perfil del usuario mediante la selección de un conjunto de palabras clave que reflejen la temática en cuestión”* [Sánchez Pereyra, 2009].

Para Judith Holt Conner se fundamenta en *“cualquier procedimiento manual o automatizado que proporciona un servicio personalizado de alerta, seleccionando la nueva literatura de probable relevancia para cada individuo, de acuerdo a sus necesidades de investigación”* [Holt, 1969; citado en Gutiérrez Vallado, 2008].

Una vez concebido y planificado un servicio por especialistas, debe ser documentado cada paso ya que muchas veces el que concibe no es el que ejecuta y tampoco obligatoriamente el que controla y evalúa el cumplimiento de la tarea. Por tanto, es necesario documentar para controlar el correcto funcionamiento de los servicios, que posibilite además, obtener mejoras a partir de los elementos relacionados y los problemas detectados. La visualización se realiza desde una perspectiva sistémica, teniendo en cuenta que *“se deben definir una serie de procesos que interactúen de forma organizada para lograr las salidas deseadas, que es el fin de todo sistema ideado por el hombre”* [Berovides Hernández, Galán Domínguez, CA 2009].

Plantea Rojas (s.a.) que la documentación relativa al diseño de cualquier servicio responde a una serie de elementos que explica detalladamente y que se han abordado por diversos autores. Para él los más importantes son el organigrama, es decir, el nivel de subordinación del nuevo diseño, condiciones influyentes, situación problemática o necesidad, usuarios del servicio, objetivo a alcanzar, descripción general de la salida, visión o aspiraciones de la oferta, los espacios de tiempo, los recursos humanos necesarios, los materiales, los financieros, descripción detallada de los procesos y entidades relacionadas, promoción, reglamentaciones, entre otros [Berovides Hernández, Galán Domínguez, CA 2009].

Por otra parte, la entidad contenedora de la información debe tener bien identificados a sus usuarios, conocer a fondo sus necesidades e intereses y establecer un estrecho vínculo entre estos y el sistema

informativo, ya que el usuario constituye el eje central de los principales procesos que se llevan a cabo en las entidades y constituyen la razón de existir de las mismas, pues a ellos va dirigida su gestión.

El perfil de interés de los usuarios puede ser construido de tres maneras, planteado por Gutiérrez (2008):

- Por el usuario exclusivamente.
- Por el usuario y luego afinado por un intermediario.
- Por el usuario con la ayuda profesional y experta del bibliotecario.

Puntualiza Gutiérrez (2008) que la experiencia apunta a que la elaboración del perfil con la ayuda del profesional, es decir, la tercera de las formas nombradas anteriormente, es la opción más precisa; el usuario cuenta con la ayuda experta que permite definir sus necesidades y expresarlas de la manera apropiada para cotejarlas contra un archivo de documentos [Berovides Hernández, Galán Domínguez, CA 2009]. Ver **Anexo 1**, ejemplo de perfil.

En este punto, se define al servicio de diseminación selectiva de información como una funcionalidad presente en cualquier sistema contenedor de la misma, que por sus características sea de uso público para fines científicos, investigativos o recreativos, permite a los usuarios que interactúan con él, recibir notificaciones de elementos que les resulten más relevante para sí mismos, a partir de criterios predefinidos y establecidos por una especie de encuesta estudiada y elaborada para conocer sus necesidades informativas.

1.4.2. Objetivos y funciones de la diseminación selectiva de información.

Si se toman en cuenta los conceptos expuestos anteriormente, se puede plantear que el objetivo de cualquier servicio de DSI es proporcionar a los usuarios información actualizada de manera selectiva y tomar sus propias necesidades e intereses como base de selección.

Los autores Perea de Ogas y Sánchez Pereyra, profundizan en el tema y lo relacionan con el aspecto tecnológico. Ellos señalan la presencia de dos tipos de objetivos en un servicio DSI, un objetivo general y otros específicos:

Objetivo General dado por Perea de Ogas y Sánchez Pereyra [Gutiérrez Vallado, 2008]:

La creación de un servicio especializado y personalizado dirigido a investigadores, técnicos, docentes y alumnos, el cual permitirá mantenerlos informados acerca de las últimas noticias, cursos, seminarios,

capacitaciones, direcciones electrónicas, textos completos y otros de interés en su área de especialización.

Objetivos específicos planteados por Perea de Ogas y Sánchez Pereyra [Gutiérrez Vallado, 2008]:

- Determinar las necesidades documentales para brindar la información rápida y oportunamente.
- Diseñar un sistema de comunicación efectivo y oportuno que brinde la información dentro de los parámetros de calidad, confidencialidad y seguridad.
- Propiciar el acceso a la información y utilizar los distintos medios que faciliten la tecnología de la información.

Enumera Cruz las funciones del servicio DSI en cinco aspectos básicos [Gutiérrez Vallado, 2008]:

1. Elaboración de los perfiles de interés: La selección de descriptores con vocabulario controlado, en el que se expresan los temas de interés. A estos descriptores se les pueden asignar diferentes valores para determinar su grado de importancia en el momento de recuperar la información. El perfil de necesidades de información puede elaborarse para usuarios individuales o en grupos con intereses comunes, con la asesoría directa del especialista en el manejo de la información.
2. Búsqueda de información: Este paso se hace en forma periódica, a través de medios manuales o automatizados. En el primer caso, los medios de recuperación podrían ser los catálogos públicos, índices, resúmenes, entre otros, de las unidades de información. En el segundo caso se explotarían las bases de datos internas y externas.
3. Distribución de resultados (referenciales): La cual se hace a través de listas que pueden incluir citas bibliográficas, citas completas o citas con resúmenes.
4. Distribución de la información (textual): Esto se puede realizar a través del servicio de préstamo de documentos, fotocopiado, fax y correo electrónico, entre otros.
5. Evaluación del servicio: Este aspecto resulta sumamente importante, toda vez que es imprescindible realizar evaluaciones periódicas del servicio, con la finalidad de verificar que se cumplan con los objetivos establecidos y corregir las posibles desviaciones. Los factores más importantes a considerar en esta instancia son los efectos directos que el servicio ha tenido en la productividad de los usuarios. Dicha evaluación se podría realizar con la aplicación de cuestionarios o entrevistas personales a los usuarios.

La evaluación es también uno de los aspectos a considerar en el diseño de servicios de información pues debe establecerse una comparación entre lo que se había propuesto en el diseño (que debe encontrarse debidamente documentado) y lo que realmente se ha logrado y, teniendo en cuenta que toda obra humana es perfectible, se pueden detectar errores que deben ser subsanados para lograr las mejoras en el servicio [Berovides Hernández, Galán Domínguez, CA 2009].

1.4.3. Características generales de la diseminación selectiva de información.

Guinchet, Menou y Sutter consideran que el servicio DSI es de difusión activa o dinámica, porque se dirige hacia el usuario, a diferencia de los que ellos clasifican como pasivos o estáticos, que son los que se encuentran a disposición del usuario en la biblioteca, como por ejemplo el servicio de préstamo, el acceso a bases de datos, servicio de fotocopias y otros [Vieira, 2003].

El servicio de DSI facilita al usuario la información que necesita sin necesidad de acudir a la entidad contenedora de la información que la proporciona, y su periodicidad depende de las necesidades de los usuarios, pueden definirse y ser semanales, mensuales, semestrales, entre otros.

Actualmente es posible efectuarla completamente de manera automática, donde cada usuario decide qué quiere recibir, con qué frecuencia y si desea darse de alta del servicio en cualquier momento que estime conveniente. Para implementar un servicio de este tipo se requiere de algún programa informático que integre en un único concepto el acceso y la búsqueda en la base de datos, la agenda del sistema operativo y el correo electrónico. Las consultas a las bases de datos funcionan automáticamente al igual que las respuestas que se envían a cada usuario registrado en la lista de correos asociada al servicio. El mensaje de correo que se envía contiene un documento anexo con las referencias de aquellos elementos que se ajusten al perfil de información que ha solicitado o los textos íntegros de los documentos [Berovides Hernández, Galán Domínguez, CA 2009].

1.4.4. La diseminación selectiva de información en la actualidad.

En este epígrafe, la investigación relaciona los conceptos antes tratados, con el objetivo de lograr que el SIGB implantado en la biblioteca de la UCI pueda contar con el servicio de diseminación selectiva de información para sus usuarios. Si bien se expuso a Koha como la base del sistema que utiliza la antes mencionada biblioteca, el mismo no posee implementado el servicio al que se hace referencia, existen otros a nivel mundial que sí poseen las características de Koha e implementan diseminación de información selectivamente, de estos se extraen las características a incorporar en el diseño de la solución.

Aleph 500

Desarrollado por la empresa israelí Ex Libris, bajo licencias privativas, mantiene liderazgo en el mercado de automatización de bibliotecas; constituye una completa solución, avanzada y basada en estándares industriales, bibliotecológicos e informáticos. Más de 1800 sistemas Aleph están actualmente instalados en bibliotecas y consorcios de 63 países, tales como Harvard University, MIT, University of Chicago, University of California, Michigan University, Universidad Nacional Autónoma de México, Biblioteca Nacional de Chile, Pontificia Universidad Católica de Chile, entre otras. El sistema brinda a los usuarios la posibilidad de realizar una variedad de tareas con facilidad, entre ellas destaca la definición de múltiples perfiles para diseminación selectiva de información, el cual notifica acerca de la disponibilidad de nuevas publicaciones de interés para ellos.

Unicorn

Distribuido por la empresa canadiense SIRSI, es un sistema de código cerrado para la gestión de los recursos de las bibliotecas. Es un software encargado de gestionar todos los procesos que abarcan el día a día en un sistema bibliotecológico, gestión de catálogos, reservas, acceso a datos, y muchos otros. Desde hace 25 años, se utiliza en bibliotecas de todo el mundo, desde entidades públicas a privadas, que alcanzan el millar de instalaciones alrededor de todo el mundo y alberga a millones de títulos en sus sistemas. El programa proporciona la diseminación selectiva de información. El usuario interesado puede ingresar su perfil en la base de datos y periódicamente, de manera automática, el sistema le enviará alertas sobre los temas de su interés y el material relacionado que ingresa a la biblioteca.

PhpMybibli

Conocido como PMB, SARL PMB Services es la empresa encargada de su creación y mantenimiento. Se trata de un SIGB open source (código abierto), que por sus características se convierte en un software robusto, potente y altamente configurable. Además de los clásicos módulos de circulación, catalogación, administración y OPAC, PMB incorpora la capacidad de gestionar publicaciones periódicas, ediciones, adquisiciones, tesoro, control de autoridades y un módulo especial de diseminación selectiva de información utilizando fuentes RSS (sindicación realmente simple, del inglés Really Simple Syndication).

Sistema	Presencia del servicio DSI	Características del servicio DSI
Aleph 500.	Si	Permite definir múltiples perfiles, notifica las nuevas publicaciones de interés.
Unicorn.	Si	Alerta automática sobre temas de interés ingresados en el perfil.
PhpMybibli.	Si	Utiliza fuentes RSS.
Koha.	No	
Otros sistemas.	Si	Se elaboran los perfiles y el alcance del servicio a partir de sus propias necesidades. Puede ser automático o manual.

Tabla 1. Resumen de los sistemas analizados.

Se concluye con la observación de una tendencia a la incorporación del servicio de disseminación selectiva de información tanto en sistemas propietarios como de código abierto, son utilizados como referencias algunos de los más usados mundialmente de una u otra política de distribución de software. Se destaca además, la presencia del servicio en innumerables instituciones científicas o docentes que a partir de su página web, brindan acceso a perfiles de disseminación, los cuales se implementaron bajo sus propios requisitos, de esa manera obtienen un servicio adaptado a sus necesidades con el esfuerzo propio de cada entidad. Esta idea se pretende utilizar para implementar la solución propuesta, es decir, adaptar las características de otros sistemas al SIGB de la biblioteca de la UCI.

Resultaría más engorroso implantar un SIGB con el servicio DSI incorporado, la experiencia adquirida de varios años de trabajo por los bibliotecarios y la institución en general se vería afectada. Además, el ser humano es por naturaleza resistente al cambio, y la implantación de un SIGB propietario no está contemplada como vía de solución, debido a que se pretende buscar la alternativa más económica posible y usar software bajo licencias libres que permitan el acceso y modificación del código fuente, a todo esto se le añade que PhpMybibli, solución homóloga bajo licencia de libre distribución, no reúne las características completas deseadas por parte de la biblioteca universitaria para el servicio DSI, lo que suscitaría igualmente una modificación de su código, algo más fácil de hacer en una aplicación

conocida por parte del grupo de desarrollo y se evitaría tener que estudiar un sistema completamente nuevo.

Todo lo anterior conlleva a definir que el servicio no presenta un esquema conceptual fijo e invariable, por el cual se establezca una arquitectura para su implementación, sino que dado el concepto es aplicable en la medida definida por la institución que pretenda brindarlo. La investigación realizada recoge de cada sistema las características homólogas a la solución propuesta para la biblioteca de la UCI, teniendo en cuenta la capacidad automática y la cantidad finita de elementos permitidos a incorporar en los perfiles, el servicio permitirá el establecimiento de diez autores e igual cantidad de temas preferidos. Además, permitirá establecer los tipos de materiales de preferencia, la periodicidad deseada para que se envíe la notificación que podrá ser diaria, semanal o mensual y por último el estado del servicio, es decir si está activado o no (Ver [Anexos 5 y 6](#)).

Todo lo antes expuesto define el perfil establecido para la solución, se destaca que no debe ser muy extenso el campo de preferencias pues resulta ilógico que un usuario pueda responder a una amplia necesidad informativa a la escala de una biblioteca. Este comportamiento es similar y aún más restrictivo en los servicios presentes en otras instituciones, ya que se trata de un servicio informativo atendiendo a una necesidad establecida y seleccionada, que incluso en muchas ocasiones se realiza de forma manual.

1.4.5. La diseminación selectiva de información en Cuba.

Durante años, dentro y fuera del país, diferentes instituciones implementaron variantes adaptadas a sus necesidades. Desde finales de los años 60, por ejemplo, se practicaba el servicio con el envío a sus usuarios de fotocopias de artículos y copias de traducciones, e incorporó luego el envío de referencias bibliográficas. En la década de los 80 se desarrollaron muchos perfiles de diseminación selectiva para asegurar el apoyo a problemas de investigación de algunos sectores productivos, lo que constituyó un trabajo conjunto del Instituto de Información Científica y Tecnológica (IDICT) y el Ministerio del Azúcar (MINAZ), entre otros [[Ponjuan Cance, 2007](#)].

Cabe destacar que el Centro Latinoamericano y del Caribe de Información en Ciencias de la Salud (BIREME), asumió la creación y desarrollo de la Biblioteca Virtual de Salud (BVS). La misma dispone actualmente entre sus productos y servicios de información el de DSI, como respuesta a las necesidades de comunidades específicas de usuarios, lo cual posibilita a su vez, que innumerables servicios se realicen y operen de modo regional y descentralizado. El servicio antes mencionado se

centra hoy en la entrega de noticias y diversos contenidos de interés por medio de la red de forma automática y sin necesidad de navegar de una página a otra [Ponjuan Cance, 2007].

La UCI también aporta antecedentes a la investigación con el uso de un servicio de disseminación selectiva de información en el pasado. Este se estableció por el grupo de Desarrollo de Productos y Servicios, fue brindado en una primera etapa a profesores y jefes de proyecto. Consistía en una plantilla elaborada en una herramienta informática con una disposición armónica y funcional para plasmar la información que fuera colectada y de relevancia para los usuarios. Las principales categorías de información incorporadas en este servicio fueron las relacionadas con las noticias, artículos científicos, reseñas de libros, nuevas adquisiciones y de información comercial. Según las investigaciones realizadas, el servicio estaba en funcionamiento en 2006, además se pudo obtener una plantilla (**Anexo 2**) marcada como la número 6, con fecha 5 de junio del propio año. Es destacable que esta recopilación iba dirigida a un conjunto de personas que compartían intereses comunes, pero que no especificaba preferencias de información almacenadas en un perfil.

Cuba ha identificado desde muy temprano la conveniencia y necesidad de dominar e introducir en la práctica social las Tecnologías de la Información y las Comunicaciones, para lograr una cultura digital como una de las características imprescindibles del hombre nuevo, lo que facilitaría a nuestra sociedad acercarse más hacia el objetivo del desarrollo sostenible. El acceso a grandes bases de conocimientos en universidades y bibliotecas, la enseñanza a distancia y la colaboración desinteresada entre centros de investigación, son a su vez, otros de los tantos logros identificables dentro del infinito universo de posibilidades que pueden brindar estas tecnologías que enaltecen la condición humana.

1.5. Lenguajes, tecnologías y herramientas a utilizar.

Tomando como punto de partida el proyecto encargado en la UCI de la implementación del SIGB para su biblioteca, se propone continuar utilizando los lenguajes, tecnologías y herramientas que se establecieron en un principio por el grupo de analistas, ya que para desarrollar cualquier sistema informático es vital la correcta selección de estos elementos base, de esta manera, se garantiza el óptimo desempeño del sistema. Además, se mencionan las versiones utilizadas de cada elemento en el desarrollo de la solución.

La selección fue motivo de una profunda investigación, luego de analizar varios SIGB, sus características, licencias en las que se distribuyen y su capacidad de dar cumplimiento a los requisitos de la biblioteca de la UCI, Koha fue el seleccionado. Unido a toda la información que se expuso

anteriormente sobre el SIGB en cuestión, se agregan como argumento de su implantación por parte del proyecto encargado, los siguientes elementos:

- Sistema desarrollado y distribuido bajo licencias de código abierto GNU/GPL.
- Es multiplataforma, puede funcionar bajo los sistemas operativos Linux y Windows.
- Los lenguajes y tecnologías que lo soportan son ampliamente utilizadas y de código abierto.
- Posee una comunidad de usuarios y programadores que cooperan constantemente para el mejoramiento de sus funcionalidades.
- Cuenta con la cooperación además de la Universidad de La Plata donde está instaurado.
- Posee una interfaz web traducida a varios idiomas (incluye el español.)
- Capacidad real de adaptación a los requerimientos de la biblioteca de la UCI (aporta la mayor cantidad de posibilidades, aunque ninguno es óptimo sin antes adaptarlo) [Porras Herrera, Veloz Morales, 2008].

Por último, se debe tener en cuenta además, el establecimiento de parámetros para no fracasar en la selección del SIGB más conveniente; se tuvo en consideración la participación de los bibliotecarios de la UCI, estos fueron:

- Licencias.
- Necesidades de la biblioteca.
- Tecnologías más usadas en la UCI.
- Herramientas para aumentar la eficiencia en las búsquedas.
- Mantenimiento y soporte internacional [Porras Herrera, Veloz Morales, 2008].

1.5.1 Lenguaje de Programación Perl.

Perl, Practical Extraction and Report Language (Lenguaje Práctico para la Extracción e Informe) es un lenguaje de programación diseñado por Larry Wall y creado en 1987. Perl fue originalmente desarrollado para la manipulación de texto. En la actualidad es ampliamente utilizado en la administración de sistemas y en el desarrollo web. Perl toma características del lenguaje C y del lenguaje interpretado Bash. La estructura completa de Perl deriva del lenguaje C, es un lenguaje imperativo, con variables, expresiones, asignaciones y bloques de códigos delimitados por llaves. Perl tiene una gran potencia en la manipulación de textos debido a que incluye expresiones regulares que facilitan el trabajo con ellos [Wall, 2008]. La versión utilizada de este lenguaje fue la 5.12.4.

1.5.2. Lenguaje de Marcas de Hipertexto: HTML.

El HTML, Hyper Text Markup Language (Lenguaje de Marcas de Hipertexto) es un lenguaje muy sencillo que permite describir hipertexto, es decir, texto presentado de forma estructurada, agradable, con enlaces que conducen a otros documentos o fuentes de información relacionadas y con inserciones multimedia; es el lenguaje que se utiliza para presentar información en la World Wide Web (WWW) [Álvarez, 2010]. Se utilizó la versión 4.0 para el desarrollo de la solución.

La descripción se basa en especificar en el texto la estructura lógica del contenido (títulos, párrafos de texto normal, enumeraciones, definiciones y citas), así como los diferentes efectos que se quieren proporcionar (cursiva, negrita, o un gráfico determinado), para que la presentación final la realice un programa especializado como los navegadores Internet Explorer o Mozilla Firefox [Feus Pérez, 2009].

Este lenguaje indica al navegador dónde colocar el texto, las imágenes y los videos, atendiendo a la disposición deseada para colocarlos en la página. No es más que una serie de etiquetas que se utilizan para definir la forma o estilo que se quiera aplicar al documento.

1.5.3. Lenguaje de Programación del Lado del Cliente: JavaScript.

Se trata de un lenguaje de programación del lado del cliente, porque es el navegador el que soporta la carga del procesamiento. Gracias a su compatibilidad con la mayoría de los navegadores modernos, es el lenguaje de programación del lado del cliente más utilizado [Concepto de JavaScript, 2008].

Es un lenguaje de programación interpretado por el navegador que se utiliza para controlar su apariencia y manipular los eventos que ocurran en su ventana. A través del JavaScript se pueden conseguir interesantes efectos en las páginas web, validar formularios, abrir y cerrar ventanas, cambiar dinámicamente el aspecto y los contenidos de una página, realizar cálculos matemáticos sencillos, entre otros [Concepto de JavaScript, 2008].

Es un lenguaje de programación bastante sencillo, pensado para hacer las cosas con rapidez, y a veces, con ligereza. Incluso, las personas que no tengan una experiencia previa en la programación, podrán aprender este lenguaje con facilidad y utilizarlo en toda su potencia con sólo un poco de práctica. Entre las acciones típicas que se pueden realizar con este lenguaje se tienen dos vertientes: por un lado, los efectos especiales sobre las páginas web, para crear contenidos dinámicos y elementos que tengan movimiento, cambien de color o cualquier otra actividad, y por el otro, JavaScript permite ejecutar instrucciones como respuesta a las acciones del usuario, con lo que se

puede crear páginas interactivas con programas como calculadoras, agendas, o tablas de cálculo [Concepto de JavaScript, 2008].

Finalmente, JavaScript posee muchas posibilidades, permite la programación de pequeños scripts, pero también de programas más grandes, orientados a objetos, con funciones y estructuras de datos complejas. Además, pone a disposición del programador todos los elementos que forman la página web, para que éste pueda acceder a ellos y modificarlos dinámicamente. Fue utilizada la versión 1.5 en el desarrollo de la solución.

1.5.4. Hojas de Estilo en Cascada: CSS.

CSS son las siglas que nombran a las hojas de estilo en cascada, el nombre proviene del inglés Cascading Style Sheets, por eso se conoce comúnmente por sus siglas. Es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML). El World Wide Web Consortium (W3C) es el encargado de formular la especificación de las hojas de estilo que servirán de estándar para los agentes de usuario o navegadores. La idea que se encuentra detrás del desarrollo de CSS es separar la estructura de un documento de su presentación [Feus Pérez, 2009].

La información de estilo puede ser adjuntada en un documento separado o en el mismo documento HTML. En este último caso podrían definirse estilos generales en la cabecera del documento o en cada etiqueta particular mediante el atributo "<style>" [Feus Pérez, 2009]. La versión utilizada fue la 2.1.

1.5.5. Tecnología CGI.

Interfaz común de pasarela (en inglés Common Gateway Interface, abreviado CGI) es una importante tecnología de la World Wide Web que permite a un cliente (explorador web) solicitar datos de un programa ejecutado en un servidor web. CGI especifica un estándar para transferir datos entre el cliente y el programa [Van Der Henst, 2001].

Las aplicaciones CGI fueron una de las primeras maneras prácticas de crear contenido dinámico para las páginas web. En una aplicación CGI, el servidor web pasa las solicitudes del cliente a un programa externo. Este programa puede estar hecho en cualquier lenguaje que soporte el servidor, aunque por razones de portabilidad se suelen usar lenguajes de script. La salida de dicho programa es enviada al cliente en lugar del archivo estático tradicional [Feus Pérez, 2009].

CGI ha hecho posible la implementación de funciones nuevas y variadas en las páginas web, de tal manera que esta interfaz rápidamente se volvió un estándar y fue implementada en todo tipo de servidores web.

1.5.6. Módulos de Perl.

Un módulo es un conjunto de operaciones que pueden ser utilizadas en cualquier script en Perl, simplemente se llama al módulo al que se hace referencia y las operaciones que de él nos interesa utilizar. Es evidente que dentro de un script se pueden llamar diversos módulos según lo que efectivamente le sirva a quien esté programando. En el sistema se utilizan varios módulos de Perl, unos escritos por la comunidad que soporta al lenguaje certificados internacionalmente, y otros escritos por la comunidad de desarrollo de Koha [[Feus Pérez, 2009](#)].

1.5.6.1. Módulo para el soporte de CGI.

Este módulo utiliza objetos de Perl para que sea más fácil llenar los formularios web y analizar su contenido. Este paquete define objetos CGI, entidades que contienen la cadena de consulta actual y otras variables de estado. Si se utilizan los métodos de un objeto CGI, se pueden examinar las palabras claves y los parámetros pasados a un script y crear formularios, cuyos valores iniciales son tomados de la consulta actual, lo que da la preservación de la información de estado. El módulo ofrece funciones de acceso directo que producen lenguaje HTML, reduciendo la mecanografía y los errores de codificación. También proporciona algunas de las funcionalidades más avanzadas de los CGI e incluye soporte a la carga de archivos, cookies, hojas de estilo en cascada y marcos. Existen dos formas de programación con el módulo CGI: uno es el estilo orientado a objetos y el otro es el estilo funcional u orientado a funciones [[Van Der Henst, 2001](#)].

1.5.6.2. Módulo HTML::Template.

Permite hacer el uso de plantillas HTML de forma simple y natural. Se extiende el HTML con algunas etiquetas nuevas como: `<TMPL_VAR>`, `<TMPL_LOOP>`, `<TMPL_INCLUDE>`, `<TMPL_IF>`, `<TMPL_ELSE>` y `<TMPL_UNLESS>`. Un archivo escrito en HTML y con estas etiquetas es llamado plantilla (en inglés *template*). Generalmente se guardan separados de los scripts, que pueden ser creados por otra persona. El uso de este módulo permite rellenar los valores de las variables, los ciclos y las ramas declaradas en las plantillas. Este módulo es uno de los más importantes porque permite separar el diseño (el HTML) de los datos generados por los scripts escritos en Perl. Además, brinda

una gran posibilidad de unirlo con el módulo CGI y facilitar en gran medida la programación web. El módulo es licenciado bajo los términos de la GPL [Feus Pérez, 2009].

1.5.6.3. Módulo DBI.

DBI (Interfaz de bases de datos independiente) es un módulo de acceso a bases de datos para el lenguaje de programación Perl. Este define un conjunto de funciones, variables y convenciones que ofrecen una interfaz de base de datos consistente e independiente de la base de datos que se esté utilizando, es decir, DBI es la interfaz independiente para base de datos de Perl, lo cual no significa que no haya otros, pero normalmente todo lo que se puede hacer con un módulo que no use DBI para acceso a bases de datos, se puede hacer con este, de forma más fácil y portable [Feus Pérez, 2009].

Es importante destacar que DBI es una capa de “pegamento” entre una aplicación y uno o más módulos controladores de base de datos. Es el módulo controlador el que hace mayor parte de la verdadera labor.

1.5.6.4. Módulo DBD.

DBD (Driver de Base de Datos, del inglés Data Base Driver) es el driver de la base de datos, es decir, se encarga de llevar a cabo lo que se pide que se haga en nuestro script Perl (usando DBI) en una base de datos específica. Existe un módulo DBD para cada tipo de base de datos, dicho módulo se encarga de pasar las peticiones que realizamos en DBI a peticiones a la base de datos sobre la que se esté trabajando. Para trabajar con una base de datos determinada nos hace falta tener controlado el módulo DBD correspondiente, por ejemplo, para trabajar con la base de datos MySQL hace falta el módulo DBD-MySQL [Feus Pérez, 2009].

1.5.7. Sistema Gestor de Base de Datos: MySQL.

MySQL es un sistema gestor de bases de datos relacionales, licenciado bajo la GPL de GNU. Su diseño multihilo le permite soportar una gran carga de forma muy eficiente. Este gestor de bases de datos es, probablemente, el más usado en el mundo del software libre, debido a su gran rapidez y facilidad de uso. Esta gran aceptación es debida, en parte, a que existen infinidad de librerías y otras herramientas que permiten su uso a través de gran cantidad de lenguajes de programación, además de su fácil instalación y configuración. Se trabajó con la versión 5.1.58 de este sistema.

Las principales características de este gestor de bases de datos son las siguientes:

- Aprovecha la potencia de sistemas multiprocesador, gracias a su implementación multihilo.
- Soporta gran cantidad de tipos de datos para las columnas.
- Dispone de API's en gran cantidad de lenguajes (C, C++, Java, PHP, entre otros).
- Gran portabilidad entre sistemas.
- Gestión de usuarios y contraseñas, manteniendo un alto nivel de seguridad en los datos.

Principales ventajas:

- Velocidad al realizar las operaciones, lo que le hace uno de los gestores con mejor rendimiento.
- Bajo costo en requerimientos para la elaboración de bases de datos, ya que debido a su bajo consumo puede ser ejecutado en una máquina con escasos recursos sin ningún problema.
- Facilidad de configuración e instalación.
- Soporta gran variedad de Sistemas Operativos.
- Baja probabilidad de corromper datos, incluso si los errores no se producen en el propio gestor, sino en el sistema en el que está.
- Conectividad y seguridad.

Principales desventajas:

- Un gran porcentaje de las utilidades de MySQL no están documentadas.
- No es intuitivo, como otros programas, por ejemplo, Microsoft Access.

1.5.8. Servidor Web Apache.

Apache es el servidor web hecho por excelencia, su fácil configuración, robustez y estabilidad hacen que cada vez millones de servidores reiteren su confianza en este programa. La licencia Apache es una descendiente de la licencias BSD (Berkeley Software Distribution), no es GPL (General Public License). Esta licencia permite hacer lo que se desee con el código fuente, incluso productos propietarios. Este servidor se utilizó en su versión 2.2.20.

Algunas razones por las que este software libre es enormemente reconocido en muchos ámbitos empresariales y tecnológicos son:

- Corre en una multitud de Sistemas Operativos, lo que lo hace prácticamente universal.
- Apache es una tecnología gratuita de código fuente libre. El hecho de ser gratuita es importante, pero no tanto como que sea de código fuente libre. Esto le da una transparencia al

software de manera que si se quiere ver qué es lo que se está instalando como servidor, se puede saber sin ningún secreto ni ninguna puerta trasera.

- Es un servidor altamente configurable y de diseño modular. Actualmente existen muchos módulos para Apache que son adaptables a este y están disponibles para que sean instalados cuando sea necesario.
- Trabaja en gran medida con Perl, PHP, Java Script Page y otros lenguajes de script. Posee todo el soporte que se necesita para tener páginas web dinámicas.
- Apache permite personalizar la respuesta ante los posibles errores que se puedan dar en el servidor. Es posible configurar Apache para que ejecute un determinado script cuando ocurra un error en concreto.
- Tiene una alta configurabilidad en la creación y gestión de logs (ficheros de registros de actividades). Permite la creación de ficheros de logs para el administrador y de este modo se puede tener un mayor control sobre lo que sucede en el servidor.

1.5.9. Lenguaje UML.

La programación orientada a objetos constituye la base de la reutilización de código por medio de componentes. UML (Lenguaje de Modelación Unificado o Unified Modeling Language, en inglés) es el lenguaje estándar adoptado por el OMG (Object Management Group) y mundialmente aceptado para la descripción de los "planos" de software. UML es un lenguaje gráfico para visualizar, especificar, construir y documentar cada una de las partes que comprende el desarrollo de software. Posee formas de modelar conceptos como los procesos de negocio y las funciones del sistema, además de aspectos concretos como lo son: escribir clases en un lenguaje determinado, esquemas de base de datos y componentes de software reusables [García, 2005]. Se modeló utilizando la versión 2.1 del lenguaje UML.

1.5.10. Herramienta CASE: Visual Paradigm for UML.

Visual Paradigm for UML es una herramienta de ingeniería de software asistida por computación (Computer Aided Software Engineering, CASE). La misma propicia un conjunto de ayudas para el desarrollo de programas informáticos, desde la planificación, pasando por el análisis y el diseño, hasta la generación del código fuente de los programas y la documentación. Ha sido concebida para soportar el ciclo de vida completo del proceso de desarrollo del software a través de la representación de todo tipo de diagramas. Constituye una herramienta de software libre de probada utilidad para el analista.

Además, está diseñada para una amplia gama de usuarios interesados en la construcción de sistemas de software de forma fiable, utilizando un enfoque orientado a objetos. La versión utilizada fue la 8.0.

Se caracteriza por:

- Software libre.
- Disponibilidad en múltiples plataformas (Windows, Linux).
- Diseño centrado en casos de uso y enfocado al negocio que generan software con mayor calidad.
- Capacidades de ingeniería directa e inversa.
- Soporta aplicaciones web.
- Soporte de UML versión 2.1.
- Modelo y código que permanece sincronizado en todo el ciclo de desarrollo.
- Diagramas de flujo de datos.

1.5.11. Entorno de Desarrollo Integrado Eclipse.

La herramienta Eclipse es un entorno de desarrollo integrado de código abierto y multiplataforma para desarrollar lo que el proyecto llama "*aplicaciones de cliente enriquecido*", opuesto a las aplicaciones "*cliente liviano*" basadas en navegadores. Esta plataforma, típicamente ha sido usada para desarrollar entornos de desarrollo integrados (del inglés IDE), como el IDE de Java llamado Java Development Kit (JDK) y el compilador que se entrega como parte de Eclipse, que son usados también para desarrollarlo a él mismo.

Eclipse fue desarrollado originalmente por IBM como el sucesor de su familia de herramientas para VisualAge. Eclipse es ahora desarrollado por la Fundación Eclipse, una organización independiente sin ánimo de lucro que fomenta una comunidad de código abierto y un conjunto de productos complementarios, capacidades y servicios. Está licenciado actualmente bajo la Eclipse Public License, con orientación al código abierto. La implementación se realizó con la versión 3.7.0.

El IDE Eclipse posee mil 468 líneas de códigos destinadas al lenguaje Perl, lo que representa el 0,07% de sus 2 millones 63 mil 83 líneas de código en total, gracias a los módulos o plugins (software adicional) proporciona toda su funcionalidad al frente de la plataforma de cliente enriquecido, a diferencia de otros entornos monolíticos donde las funcionalidades están todas incluidas, las necesite el usuario o no. Este mecanismo de módulos es una plataforma ligera para componentes de software.

La definición que da el proyecto Eclipse acerca de su software es: "*una especie de herramienta universal, un IDE abierto y extensible para todo y nada en particular*".

1.6. Metodología de desarrollo adoptada.

Cuando se realizan proyectos de elaboración de software, es necesario hacerlo mediante una metodología de desarrollo que ayude a organizar y planificar todo el proceso para poder obtener un producto de óptima calidad y los clientes queden satisfechos con el resultado. Entre estas metodologías se encuentran las llamadas "*ágiles*", las cuales permiten desarrollar y obtener rápidamente aplicaciones de menor envergadura y tiempo de desarrollo donde el cliente participa en todo momento a lo largo de todo el proceso, algunos ejemplos son: Scrum y XP; mientras las "*tradicionales*" son utilizadas para dirigir procesos más grandes y con un tiempo mayor de duración. A inicios de la creación del proyecto encargado de desarrollar el SIGB de la biblioteca de la UCI, se estableció como metodología a RUP, por lo que la solución continúa el uso de la misma para elaborar la documentación en correspondencia con la ya existente, sin dejar de destacar que una metodología ágil hubiera sido más conveniente ya que el grado de complejidad y alcance es perfectamente aplicable a este tipo de técnicas metodológicas.

La metodología RUP (Rational Unified Process), es un proceso de desarrollo de software que junto al Lenguaje Unificado de Modelado (UML), constituyen el estándar más utilizado para el análisis, implementación y documentación de sistemas orientados a objetos. El RUP divide el proceso de desarrollo en ciclos, donde se obtiene un producto al final de cada uno, los cuales se dividen en fases que finalizan con un hito en el que se debe tomar una decisión importante. Se caracteriza además, por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. Incluye a los artefactos, que son los productos tangibles del proceso, y a los roles, que definen el papel que desempeña una persona en un determinado momento. Una persona puede desempeñar distintos roles a lo largo del proceso [Feus Pérez, 2009].

1.7. Fundamentación y análisis de la propuesta de solución.

El SIGB de la biblioteca de la UCI necesita contar con el servicio de disseminación selectiva de información. El mismo ofrecería múltiples ventajas a sus usuarios, por lo que si se tiene en cuenta que Koha, la base del sistema, no lo posee hasta el momento, puede integrársele una solución que siga la arquitectura y el patrón de diseño establecido para su implementación, es decir, agregarle al SIGB de la UCI las funcionalidades que le permitan contar con el servicio para la disseminación selectiva de

información como si fuera un módulo más integrado. La incorporación de este elemento fortalecería más el desempeño desde el punto de vista autónomo del SIGB para con sus usuarios y se lograría que los mismos reciban información en correspondencia con los criterios personales de selección establecidos en los perfiles individuales de cada uno, todo esto automáticamente y sin la intervención de un encargado, a través del correo electrónico por las facilidades de comunicación que brinda este difundido servicio internacional, presente de manera inherente en la universidad.

La investigación se aproxima a los conceptos analizados y los relaciona. Los SIGB que cuentan con el servicio de DSI han obtenido muy buenos resultados en su gestión. Es de interés contar con esas funcionalidades en la biblioteca de la UCI, lo que garantizaría el establecimiento y explotación de tecnologías de alcance mundial y de última concepción, para lograr la igualdad con los avances en el desarrollo del campo de la informática y las comunicaciones, los cuales han automatizado casi todos los procesos con los que el hombre se relaciona a diario. Esto constituye además, un ahorro para el país por concepto de adquisición de un servicio similar que no está presente en el sistema ya establecido y en explotación, se elimina la necesidad de adquirir otro equivalente y aumenta el uso y funcionalidad del existente. Constituirá también un elemento exportable cuando finalice su implementación. Además, se consolidará el sistema como propio y adaptado a nuestras instituciones, listo para ser desplegado en toda la red de bibliotecas de la nación.

1.8. Conclusiones del Capítulo.

El capítulo ofrece los aspectos teóricos a tener en cuenta para la implementación del módulo para el servicio de diseminación selectiva de información del SIGB de la UCI. Además, se pudo comprobar la importancia del estudio de los conceptos y las soluciones homólogas para finalmente contar con los elementos necesarios que soporten cualquier decisión que se tome, esta necesita ser defendida con criterios claros y bien justificados. También fue posible establecer los lenguajes, tecnologías y herramientas a utilizar para la realización de la solución propuesta, así como, la metodología de desarrollo que controle y documente el avance del proceso de implementación. Después de la lectura del capítulo, el lector tendrá a su alcance la razón de ser y el eje central de la presente investigación.

Capítulo 2. Características del sistema.

2.1. Introducción al Capítulo.

En el presente capítulo se describe la propuesta de solución para el desarrollo del módulo que constituye el objetivo general de la investigación. También se caracteriza el entorno del sistema donde se definen nuevos procesos automatizables y se establece el objeto de automatización de la solución. Además, se detallan los dos primeros flujos de trabajo de la metodología propuesta (RUP) en el capítulo anterior. Estos flujos de trabajos, incluyen de forma general la elaboración del modelo de dominio, la identificación y levantamiento de los requisitos funcionales y no funcionales que requiere el sistema, los casos de uso asociados a estos, las descripciones y los diagramas representativos de los elementos mencionados.

2.2. Entorno del sistema.

El SIGB de la biblioteca de la UCI está desplegado con el objetivo de automatizar los procesos que allí se realizan. El uso de aplicaciones informáticas especializadas para bibliotecas facilita el trabajo de los bibliotecarios y brinda servicios más eficientes y rápidos a los usuarios que utilizan las redes de computadoras para acceder a catálogos en línea de instituciones bibliotecarias, así se acortan las distancias y el tiempo. El local de la biblioteca es como cualquier otro, un lugar para la lectura, la investigación o el estudio y los usuarios son atendidos por el personal capacitado, todo lo antes constituye el entorno físico del sistema.

Después de analizar el entorno físico, se describen las características que posee el entorno virtual. Este está comprendido por una interfaz de usuario amigable, donde el primer paso es la autenticación y luego se brindan una serie de opciones configurables o de uso para el usuario. Estas se pueden establecer como:

- Opciones de búsqueda simple y avanzada.
- Exploración por temas.
- Detalles personales
- Préstamos en vigor.
- Sanciones impuestas.
- Historial de lecturas.
- Sugerencias de adquisición para la biblioteca.
- Control de listas privadas y públicas de ejemplares.

Cabe destacar que las búsquedas simples y avanzadas están disponibles aun cuando el usuario no está autenticado o no es un usuario del sistema, es decir, si solo es un visitante virtual.

2.3 Objeto de automatización.

Teniendo en cuenta los procesos que no han sido automatizados, la solución plantea que la carencia de nuevas funcionalidades constituye un problema existente en el sistema, en el cual existen procesos automatizables actualmente. Por tanto, el objeto a automatizar se define como la incorporación de un servicio de diseminación selectiva de información en el SIGB de la biblioteca de la UCI, el cual envíe notificaciones a los usuarios según los criterios de selección establecidas en un perfil que contenga las necesidades informativas de cada uno de ellos.

2.3.1. Flujo de eventos automatizados.

Inicialmente el sistema impacta sobre el usuario con una interfaz sencilla y con pocas opciones, basada en una vista simple y de fácil recorrido visual para una primera interacción hombre-máquina. El sistema brinda las posibilidades de realizar búsquedas y la de autenticación para acceder a las opciones completas disponibles para los interesados, donde se encuentran las descritas en el entorno del sistema, y además, se agregará la opción de configurar el servicio de diseminación selectiva de información.

Una vez dentro de la interfaz del servicio, se dispondrá de un cuadro de texto para la entrada de criterios los cuales serán buscados en el sistema mientras se teclee, los resultados se mostrarán en una lista de auto-completamiento donde el carácter introducido se resaltará en azul, de esta manera procede igual al establecer los autores y los temas favoritos. Toda selección podrá ser guardada siempre y cuando el cuadro de chequeo a la izquierda este marcado en el momento de presionar el botón para guardar. Los tipos de materiales preferidos se podrán actualizar, así mismo el período de notificación y el estado del servicio, estos últimos establecen el envío de correos diaria, semanal o mensualmente y el sistema comprobará si está activado o no el servicio antes de enviar las notificaciones, terminando de esta manera el flujo de eventos.

2.4. Modelo de dominio.

El modelo de dominio captura los tipos más importantes de objetos en el contexto del negocio, representando las “cosas” que existen o los eventos que se involucran en el ambiente del negocio. Además, brinda un marco de trabajo conceptual de los elementos pertenecientes al dominio del

problema, ayuda a pensar enfocándose en la semántica, proporciona un glosario de términos basado en sustantivos, fundamenta el modelamiento de casos de uso y se basa en la estructura definida posibilitando describir el estado del dominio del problema en cualquier momento [García, 2005].

El modelo de dominio se debe concebir como un diccionario visual de abstracciones que será utilizado en fases posteriores y cuya función principal es ayudar a comprender el problema a tratar, es una de las vistas del modelo de requerimientos. Las clases en el modelo del dominio son las abstracciones principales del sistema y el modelo se representa como un diagrama de clases con las abstracciones identificadas durante el análisis. Si se toma la definición antes expuesta, es posible establecer los objetos del dominio de la solución como representaciones de las "cosas" que existen y los eventos que ocurren en el entorno en el que trabaja el sistema.

La solución a desarrollar no posee un modelo de negocio bien definido, ya que el objetivo es ofrecer nuevas funcionalidades para resolver la problemática existente, caracterizada por la carencia de funciones que incrementen la usabilidad y eficiencia del SIGB de la biblioteca de la UCI que ya está implantado y en funcionamiento. Por ese motivo, mediante un modelo de dominio se logra la comprensión de los conceptos, los cuales son tratados como objetos que pueden ser clases del dominio o eventos, estructurados en un glosario de términos que permite fácilmente a los involucrados (miembros del proyecto y bibliotecarios) comprender los procesos que se llevarán a cabo luego de la implementación.

La figura a continuación representa el modelo de dominio de la solución para la diseminación selectiva de información del SIGB de la biblioteca de la UCI. La clase "Usuario" representa a los usuarios que acceden al sistema, estos poseen un perfil que solo les pertenece a ellos y está simbolizado por la clase "Perfil", todos los perfiles pertenecen al SIGB representado por la clase "SIGB". La clase "Notificación" es generada por el SIGB siguiendo la lógica del negocio, esta indica que se genere una notificación cuando se adquieran nuevos ejemplares que cumplan con los criterios personales de los usuarios establecidos en sus perfiles de diseminación selectiva de información, y sea enviada por medio del correo electrónico a modo de aviso.

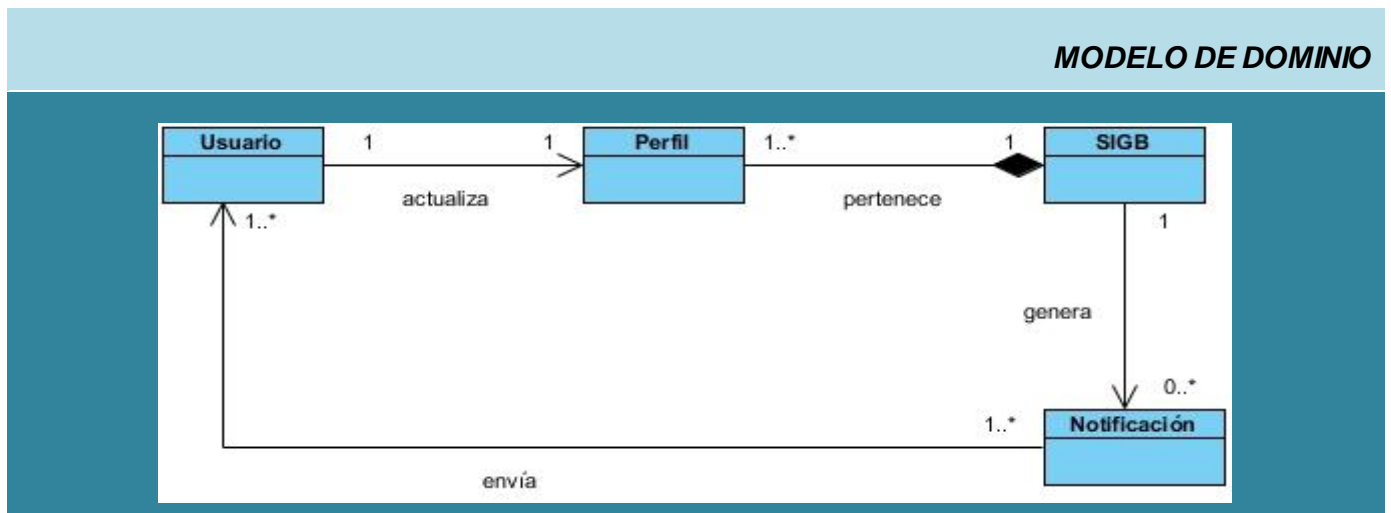


Figura 1. Diagrama de clases del modelo de dominio.

2.4.1. Glosario de términos del dominio.

Usuario.

Es el usuario que necesita hacer uso del servicio DSI, para ello actualiza su perfil donde establece los criterios por los cuales el sistema debe generar las notificaciones.

Perfil.

Se encarga de mostrar y almacenar las preferencias del usuario para la definición de sus criterios personales y la configuración general del servicio, el mismo está contenido en el SIGB.

SIGB.

Es el sistema integrado donde se gestiona todo el contenido y la información disponible para utilidad del usuario, el volumen de datos introducidos puede aumentar vertiginosamente por lo que se crea el servicio DSI para filtrar lo más relevante para un usuario según su necesidad informativa.

Notificación.

Es generada por un script que se ubica en el SIGB, su función es contener los datos que serán enviados al usuario para alertarle sobre nuevos ejemplares adquiridos que respondan a sus necesidades informativas, recogidas a través del perfil.

2.5. Requisitos funcionales del sistema.

Los requisitos funcionales son capacidades o condiciones que el sistema debe cumplir. Estos deben ser comprensibles por clientes, usuarios y desarrolladores, deben tener una sola interpretación y estar definidos en forma medible y verificable. A continuación se muestra el listado de requisitos [Jacobson, 2004]:

- RF_1 Buscar autor.
- RF_2 Agregar autores a la lista de autores preferidos.
- RF_3 Mostrar autores preferidos.
- RF_4 Eliminar autores de la lista de autores preferidos.
- RF_5 Buscar tema.
- RF_6 Agregar temas a la lista de temas preferidos.
- RF_7 Mostrar temas preferidos.
- RF_8 Eliminar temas de la lista de temas preferidos.
- RF_9 Seleccionar tipo de material deseado.
- RF_10 Seleccionar estado del servicio.
- RF_11 Seleccionar periodicidad.
- RF_12 Elaborar notificación.
- RF_13 Enviar notificación.

2.6. Requisitos no funcionales del sistema.

Los requerimientos no funcionales son propiedades o cualidades que el producto debe tener. Representan las características que hacen al producto atractivo, usable, rápido o confiable. Son fundamentales en el éxito del producto y normalmente están vinculados a requisitos funcionales [Jacobson, 2004].

Usabilidad.

La usabilidad se refiere a la facilidad de uso por parte del usuario, a la capacidad del sistema de ser comprendido, aprendido y usado por este. En este caso, la aplicación web debe ser flexible y de fácil aprendizaje, pues se trata en todo lo posible de mantener un estándar de operatividad que logre que las interacciones del usuario con el sistema sean predecibles y familiares. Será portador de múltiples vías por las cuales el usuario puede realizar una tarea y debe brindar la posibilidad de diálogos, por lo

que el sistema puede ser usado por cualquier persona que posea conocimientos básicos en el manejo de la computadora y en ambientes web.

Rendimiento.

Las pantallas deben estar poco cargadas de imágenes para garantizar que la ejecución de los hipervínculos, las adiciones, modificaciones y búsquedas no tenga un tiempo de espera muy prolongado y lograr de esta manera una respuesta rápida del sistema.

Portabilidad.

La portabilidad se refiere a la capacidad que tienen los programas de ejecutarse en diferentes sistemas operativos con mínimas modificaciones. El sistema debe ser capaz de ejecutarse sobre plataforma Linux, Windows, Unix, etc.

Seguridad.

El sistema tendrá acceso diferenciado con contraseña para cada usuario, con permiso de acceso o negación por parte de la biblioteca. El sistema deberá estar protegido contra accesos no autorizados y las modificaciones de información.

Interfaz.

La interfaz de usuario debe ser sencilla, con colores acordes con los recomendados por la institución, amigable, intuitiva y de fácil navegación por el usuario, con el objetivo de evitar la resistencia humana al uso del nuevo sistema, ya que el factor humano determina en gran medida el éxito o el fracaso del mismo.

Licencia.

El sistema está respaldado totalmente por la Licencia Pública General (GNU, por sus siglas en inglés).

Legales, de Derecho de Autor y otros.

El diseño del módulo para el servicio de diseminación selectiva de información, así como el Sistema Integrado de Gestión Bibliotecaria del cual forma parte el mismo pertenece al proyecto Biblioteca UCI.

2.7. Definición de los casos de uso.

Los casos de uso son artefactos narrativos que describen, bajo la forma de acciones y reacciones, el comportamiento del sistema desde el punto de vista del usuario. Por lo tanto, establece un acuerdo entre clientes y desarrolladores sobre las condiciones y posibilidades (requisitos) que debe cumplir el

sistema. Constituyen una secuencia de interacciones entre el sistema y alguien o algo que usa alguno de sus servicios. Un caso de uso es iniciado por un actor, quien desde ese momento junto con otros actores, intercambia datos, controla el sistema y participa en la interacción con otros casos de uso.

2.7.1. Actores del Sistema.

Los actores del sistema representan entidades externas que interactúan directamente con el sistema (personas, máquinas u otros sistemas) [Jacobson, 2004].

Actor	Objetivo
Usuario	Establecer sus criterios de selección para hacer uso del servicio de DSI del SIGB de la biblioteca, de esta manera podrá recibir notificaciones a su correo electrónico automáticamente.
Reloj	Actuar como agente de ejecución automática de scripts bajo parámetros de tiempo definidos, garantiza el envío de notificaciones al usuario.

Tabla 2. Actores del sistema.

2.7.2. Listado de casos de uso del sistema.

Los requisitos funcionales identificados son reunidos en los siguientes casos de uso:

- CU_1 Gestionar autores.
- CU_2 Gestionar temas.
- CU_3 Seleccionar tipo de material.
- CU_4 Seleccionar estado.
- CU_5 Seleccionar periodicidad.
- CU_6 Elaborar notificación.
- CU_7 Enviar notificación.

2.7.3. Diagrama de Casos de Uso.

El diagrama de casos de uso es un modelo del sistema que contiene actores, casos de uso y sus relaciones. El diagrama de casos de uso describe lo que hace el sistema para cada tipo de usuario y

permite que los desarrolladores del sistema y los clientes lleguen a un acuerdo sobre los requisitos, es decir, sobre las condiciones y posibilidades que debe cumplir el sistema.

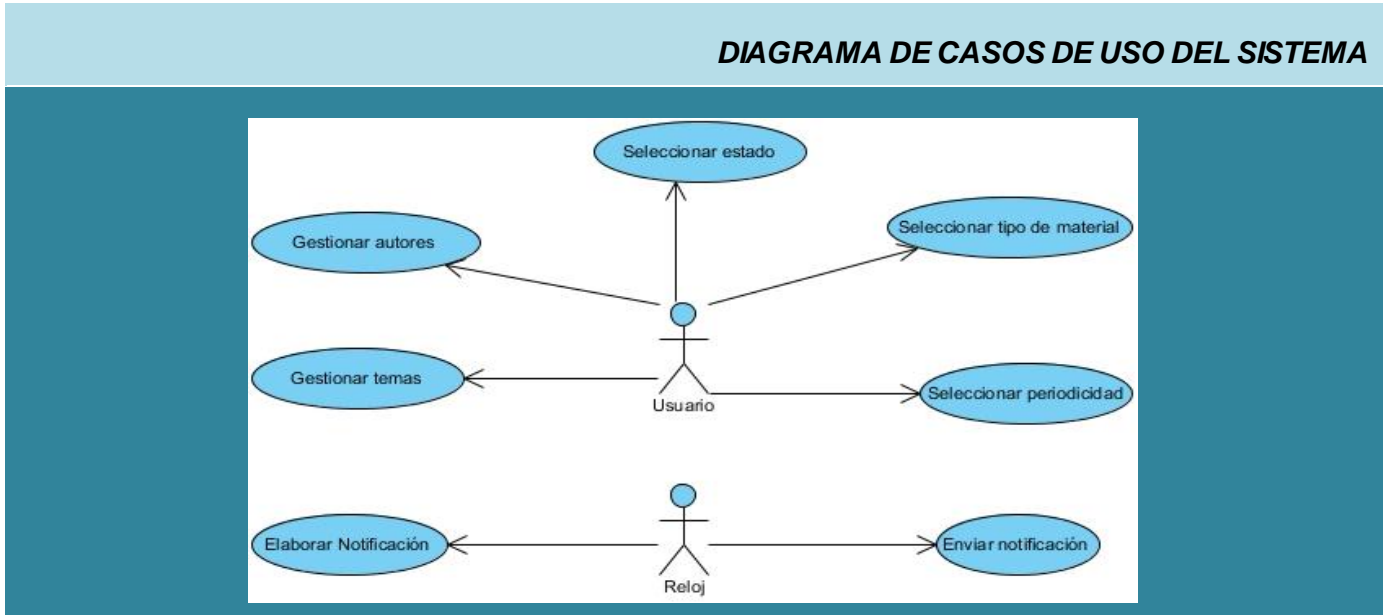


Figura 2. Diagrama de casos de uso del sistema.

2.7.4. Patrón de casos de uso CRUD.

La experiencia en la utilización de casos de uso ha evolucionado en un conjunto de patrones que permiten con más precisión reflejar los requisitos reales, haciendo más fácil el trabajo con los sistemas y mucho más simple su mantenimiento. Dado un contexto y un problema a resolver, estas técnicas han mostrado ser la solución adoptada en la comunidad del desarrollo de software. Se presentan a modo de herramientas que permiten resolver los problemas que se les planteen a los desarrolladores de una forma ágil y sistemática. Estos patrones se enfocan hacia el diseño y las técnicas utilizadas en modelos de alta calidad y no en cómo modelar usos específicos. Utilizando estos patrones, arquitectos, analistas, ingenieros, y gerentes pueden lograr mejores resultados de forma más rápida.

Este patrón está presente en los casos donde se quiere realizar inserciones, eliminaciones, modificaciones y consultas a alguna entidad del sistema. Su nombre es un acrónimo de las palabras en inglés Create, Read, Update, Delete. El patrón CRUD Completo consiste en un caso de uso para administrar la información, nos permite modelar las diferentes operaciones para administrar una entidad de información, tales como crear, leer, cambiar y eliminar o dar de baja.

2.8. Descripciones de casos de uso.

Las descripciones de casos de uso son reseñas textuales del caso de uso, explican los procesos o actividades que tienen lugar en el caso de uso. A continuación se muestra la descripción textual de un caso de uso tomado como ejemplo, para ver las descripciones detalladas de todos los casos de uso ver **Anexo 3**.

CU_1 Gestionar autores.

Objetivo	El usuario busca, agrega y elimina autores de su lista de autores preferidos, el sistema muestra en todo momento los autores que estén definidos como preferidos con el fin de gestionarlos para recibir notificaciones de nuevos materiales bajo esas autorías.	
Actores	Usuario: (Inicia).	
Resumen	El caso de uso se inicia cuando el usuario busca, agrega o elimina autores de su lista de autores preferidos, finaliza cuando el usuario termina la gestión de autores.	
Complejidad	Media.	
Prioridad	Crítico	
Precondiciones	El usuario ha sido autenticado en el sistema.	
Postcondiciones	El usuario debe haber entrado a gestionar sus preferencias de diseminación selectiva de información.	
Flujo de eventos		
Flujo básico Gestionar autores.		
	Actor	Sistema
1.	Busca, agrega, elimina o visualiza autores de su lista de autores preferidos.	1.1 Permite la ejecución de las siguientes opciones: <ul style="list-style-type: none"> a) Buscar autores. b) Agregar a preferidos. c) Mostrar preferidos. d) Eliminar autores preferidos.

Sección 1: “Buscar autores”		
Flujo básico Buscar autor		
	Actor	Sistema
1.	Introduce un criterio para buscar por autor en un campo de texto cualquiera de los diez disponibles.	1.1 Muestra los resultados de la búsqueda que correspondan al criterio entrado por el actor, auto-completando de la lista de autores mientras se introducen los caracteres hasta que coincida con el criterio del usuario.
Flujos alternos		
1. Fallo de conexión.		
	Actor	Sistema
1.		1.1 Si ocurre un fallo de conexión el sistema mostrara el mensaje “Error de conexión.”
Sección 2: “Agregar a preferidos”		
Flujo básico Agregar autores a la lista de autores preferidos.		
	Actor	Sistema
1.	Marca los cuadros de chequeo correspondientes a los autores y presiona el botón Guardar para establecerlos como preferidos.	1.1 Agrega los elementos que se encuentren en cada uno de los campos de texto que estén marcados por su cuadro de chequeo correspondiente a la lista de autores preferidos. 1.2 Cambia la etiqueta del botón Guardar a Guardando... mientras se realiza la acción.
Flujos alternos		
1. Fallo de conexión.		
	Actor	Sistema
1.		1.1 Si ocurre un fallo de conexión el sistema mostrara el mensaje “Error de conexión.”
Sección 3: “Mostrar preferidos”		
Flujo básico Mostrar autores preferidos.		
	Actor	Sistema
1.		1.1 Muestra los autores preferidos en cada uno

		de los campos de texto en los cuales fueron establecidos con su correspondiente marca en el cuadro de chequeo, sucede en todo momento, después que son guardados.
Sección 4: “Eliminar autores preferidos”		
Flujo básico Eliminar autores de la lista de autores preferidos.		
	Actor	Sistema
1.	Desmarca los cuadros de chequeo de los autores que desea eliminar y presiona el botón Guardar.	1.1 Elimina los elementos que corresponden a los que fueron desmarcados de los cuadros de chequeo. 1.2 Cambia la etiqueta del botón Guardar a Guardando... mientras se realiza la acción.
Flujos alternos		
1. Fallo de conexión.		
	Actor	Sistema
1.		1.1 Si ocurre un fallo de conexión el sistema mostrara el mensaje “Error de conexión.”

Tabla 3. Descripción detallada del CU Gestionar autores.

2.9. Conclusiones del Capítulo.

En este capítulo se analizó de forma general el entorno físico y virtual del sistema, el objeto de automatización y los principales flujos a los que va dirigido para comprender el negocio existente y definir las características a incorporar. La representación del modelo de dominio ayudó a comprender el entorno y la forma en que opera el sistema visto desde elementos del mundo real y como estos interactúan mediante eventos. Por medio del planteamiento de los requerimientos funcionales y no funcionales con los que debe contar el sistema, es posible comprender las funcionalidades que requiere el módulo y de qué manera serán presentadas al usuario. Mediante las descripciones de los casos de uso fue posible distinguir la interacción del usuario con el sistema y la respuesta del mismo a cada requisito funcional, utilizando un lenguaje comprensible.

Capítulo 3. Análisis y diseño de la propuesta de solución.

3.1. Introducción al Capítulo.

En el presente capítulo se abordará el flujo de trabajo de Análisis y Diseño. Se expondrá a través de un grupo de artefactos cómo será llevada la solución del sistema que se propone hasta el diseño. Para su modelado se utilizan los diagramas de clases del análisis, de interacción, el diseño de clases y el esquema de la base de datos que se necesita para el almacenamiento de la información a persistir. Por último, se expone la arquitectura base del SIGB de la biblioteca de la UCI y por ende la utilizada para desarrollar la propuesta de solución planteada en la investigación.

3.2. Análisis.

El análisis del sistema constituye uno de los flujos de trabajos realizados durante el proceso de desarrollo de software. Este se desarrolla fundamentalmente dentro de la fase de elaboración, aunque se tiene en cuenta a lo largo del ciclo de vida del proyecto. El análisis se basa en la obtención de una visión del sistema que se preocupa de ver qué hace el mismo, de modo que sólo se interesa por los requisitos funcionales levantados. El objetivo de este flujo de trabajo es traducir los requisitos a una especificación que describe cómo implementar el sistema [Jacobson, 2004].

3.2.1. Modelo de Análisis.

EL modelo de análisis ofrece una especificación más precisa de los requisitos que la obtenida en la captura de requisitos. La misma es estructurada de un modo que sea más fácil de comprender, preparar y modificar. En él, se identifican las clases que describen la realización de los casos de uso, los atributos y las relaciones entre ellas. Constituye además, una aproximación al modelo de diseño, sin embargo no contempla el lenguaje de programación que se va a utilizar para el desarrollo de la aplicación, debido a que su objetivo principal es comprender con exactitud los requisitos del software y no explicar su solución.

Para una mayor comprensión se relacionan las clases que son utilizadas para la elaboración del modelo de análisis:

- **CI_<Nombre de la clase>**: Modelan la interacción entre los usuarios y el sistema, es decir, ventanas, formularios, dispositivos, sistemas externos, entre otros.
- **CC_<Nombre de la clase>**: Encapsulan el comportamiento de cada caso de uso y coordinan el trabajo de las clases interfaz y entidad.

- **CE_<Nombre de la clase>**: Modelan toda la información del sistema que posee una vida larga y que puede ser persistente.

3.2.2. Diagrama de clases del análisis.

El diagrama de clases del análisis es un artefacto en el que se representan los conceptos en un dominio del problema. Representa las cosas del mundo real, no aquellas de la implementación automatizada. El caso de uso Gestionar autores se ha tomado como ejemplo para representar el análisis realizado.

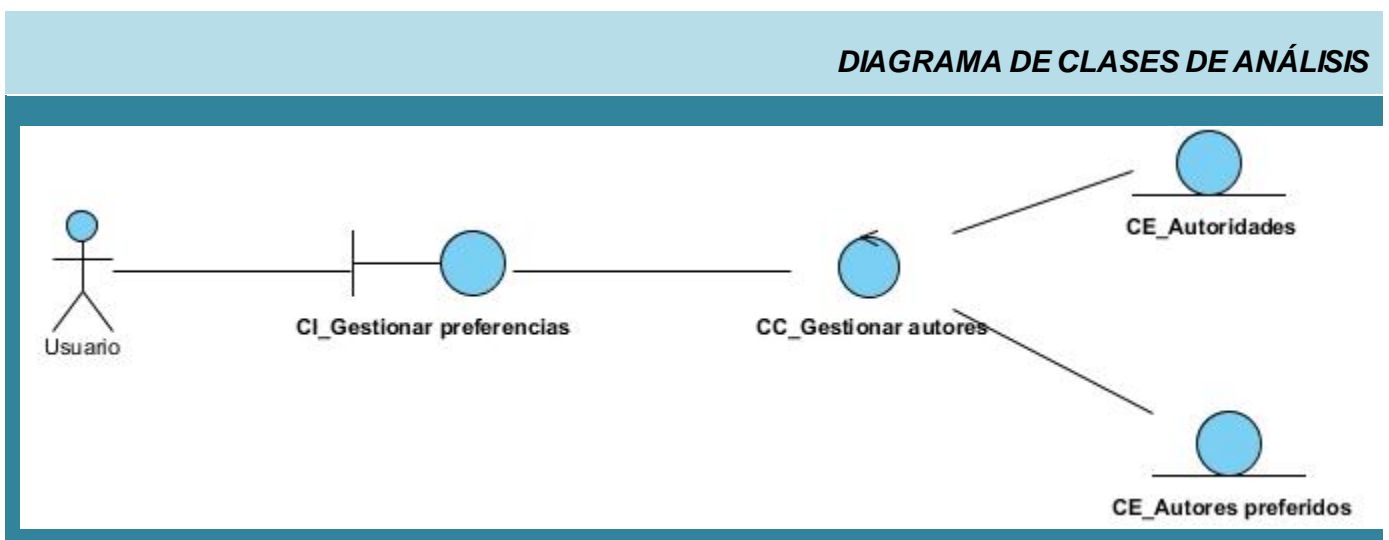


Figura 3. Diagrama de clases del análisis del CU Gestionar autores.

3.3. Diseño.

El diseño es un refinamiento del análisis, este tiene en cuenta los requisitos no funcionales que en definitiva definen el cómo cumple el sistema sus objetivos. El diseño debe ser suficiente para que el sistema pueda ser implementado sin ambigüedades. De hecho, cuando la precisión del diseño es muy grande, la implementación puede ser hecha por un generador automático de código. El flujo de trabajo está compuesto por los diagramas de interacción, que pueden ser de colaboración o secuencia y los diagramas de clases del diseño. El caso de uso Gestionar autores se ha tomado como ejemplo para representar los diagramas del diseño [Jacobson, 2004].

3.3.1. Diagrama de colaboración.

Se utilizan para ilustrar la realización de un caso de uso. Muestran como los objetos interactúan para lograr el comportamiento de un caso de uso o parte de este, de esta forma definen los roles de los mismos. A diferencia de los diagramas de secuencia su principal objetivo es mostrar la relación entre dichos objetos. Estos diagramas tienen una mayor utilidad cuando se utilizan en interacciones entre un número no muy grande de objetos, pues en caso contrario el número de mensajes entre estos crece y el diagrama se hace difícil de entender; en estos casos los diagramas de secuencia son una mejor elección [Jacobson; Rumbaugh; Booch, 2000].

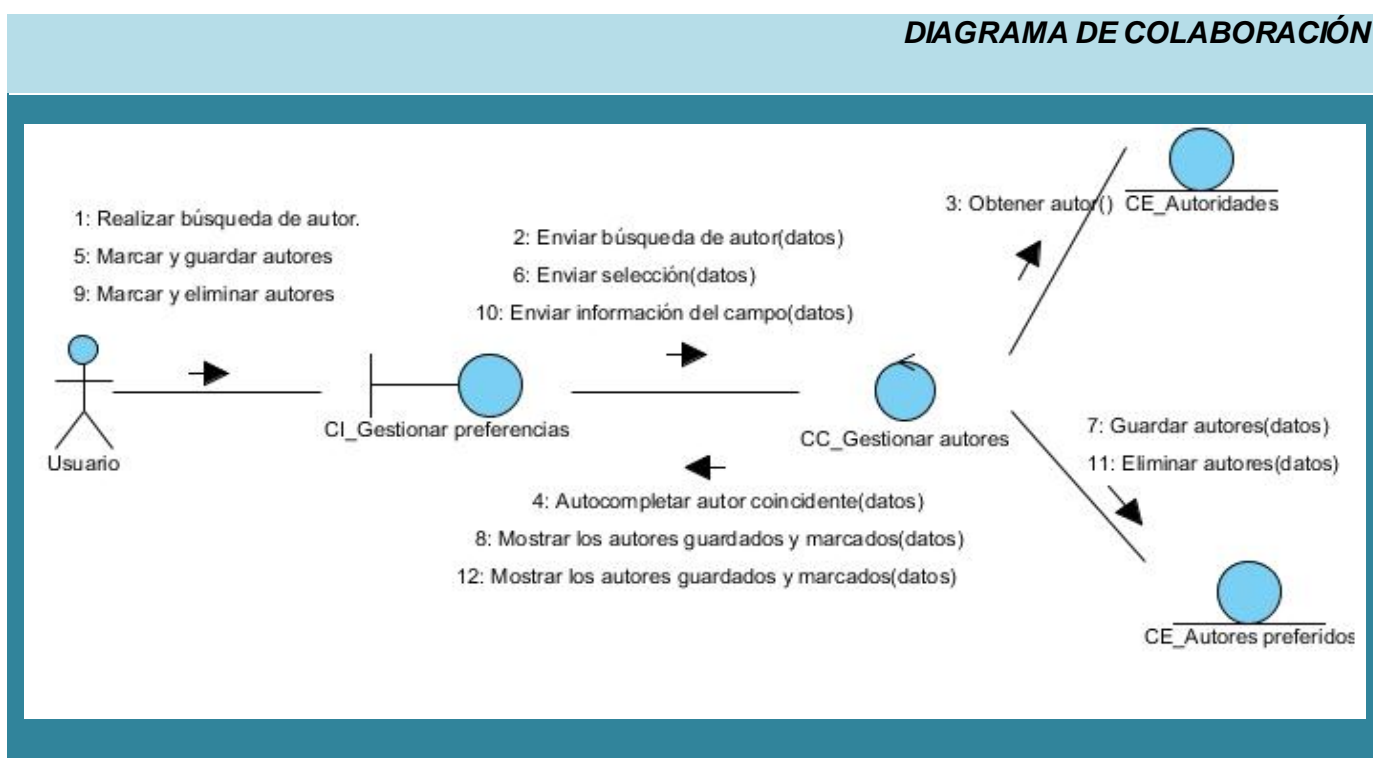


Figura 4. Diagrama de colaboración del CU Gestionar autores.

3.3.2. Diagrama de secuencia.

También se utilizan para ilustrar la realización de un caso de uso. Son particularmente importantes para los diseñadores pues aclaran los roles jugados por los objetos en un flujo, lo cual le proporciona un gran valor para la determinación de las responsabilidades de las clases. A diferencia del diagrama de colaboración este incluye la secuencia cronológica de los mensajes y no la relación entre los objetos, por lo que es mejor su utilización cuando el orden en el tiempo de los mensajes es de importancia [Jacobson; Rumbaugh; Booch, 2000].

DIAGRAMA DE SECUENCIA

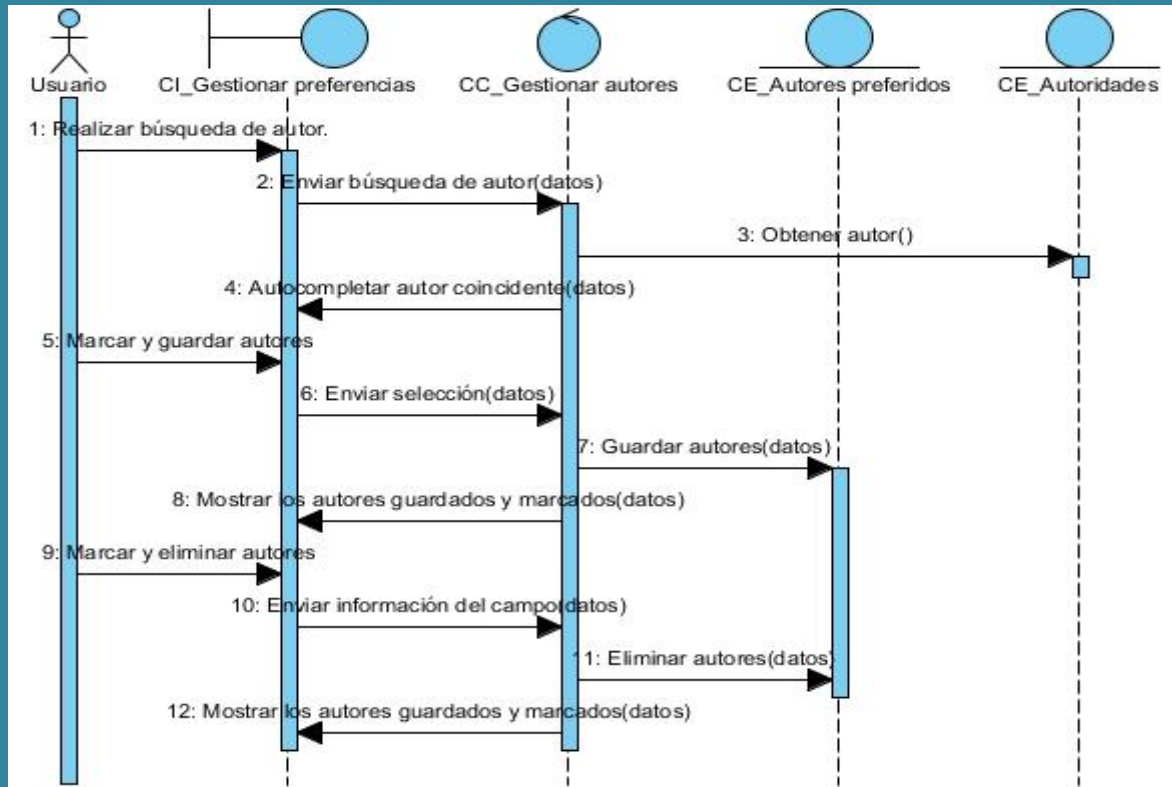


Figura 5. Diagrama de secuencia del CU Gestionar autores.

3.3.3. Diagrama de clases del diseño.

Los diagramas de clases del diseño son los más utilizados en el modelado de sistemas orientados a objetos. Un diagrama de clases del diseño muestra un conjunto de clases, interfaces y colaboraciones así como sus relaciones. Se utilizan para modelar la vista de diseño estática de un sistema, esto incluye modelar el vocabulario del sistema, modelar las colaboraciones o modelar esquemas.

En dependencia del nivel en el que se modele (análisis, diseño, implementación), los distintos elementos que se tienen en un modelo pueden “marcarse” para indicar que cumplen con un objetivo determinado. Estas marcas pueden ser aplicadas a paquetes, clases, atributos de una clase, operaciones de una clase, relaciones entre clases y son muy útiles para tener una idea a simple vista del propósito que tiene un elemento del modelo en un diagrama. Dichas marcas son conocidas como estereotipos, y se definen de esta manera: <<link>>, así se ejemplifica el estereotipo de una relación

entre clases donde existe una interconexión entre ellas y se puede estereotipar siguiendo los términos afines a una tecnología o lenguaje específico, en este caso aquellos que se relacionan con la Web.

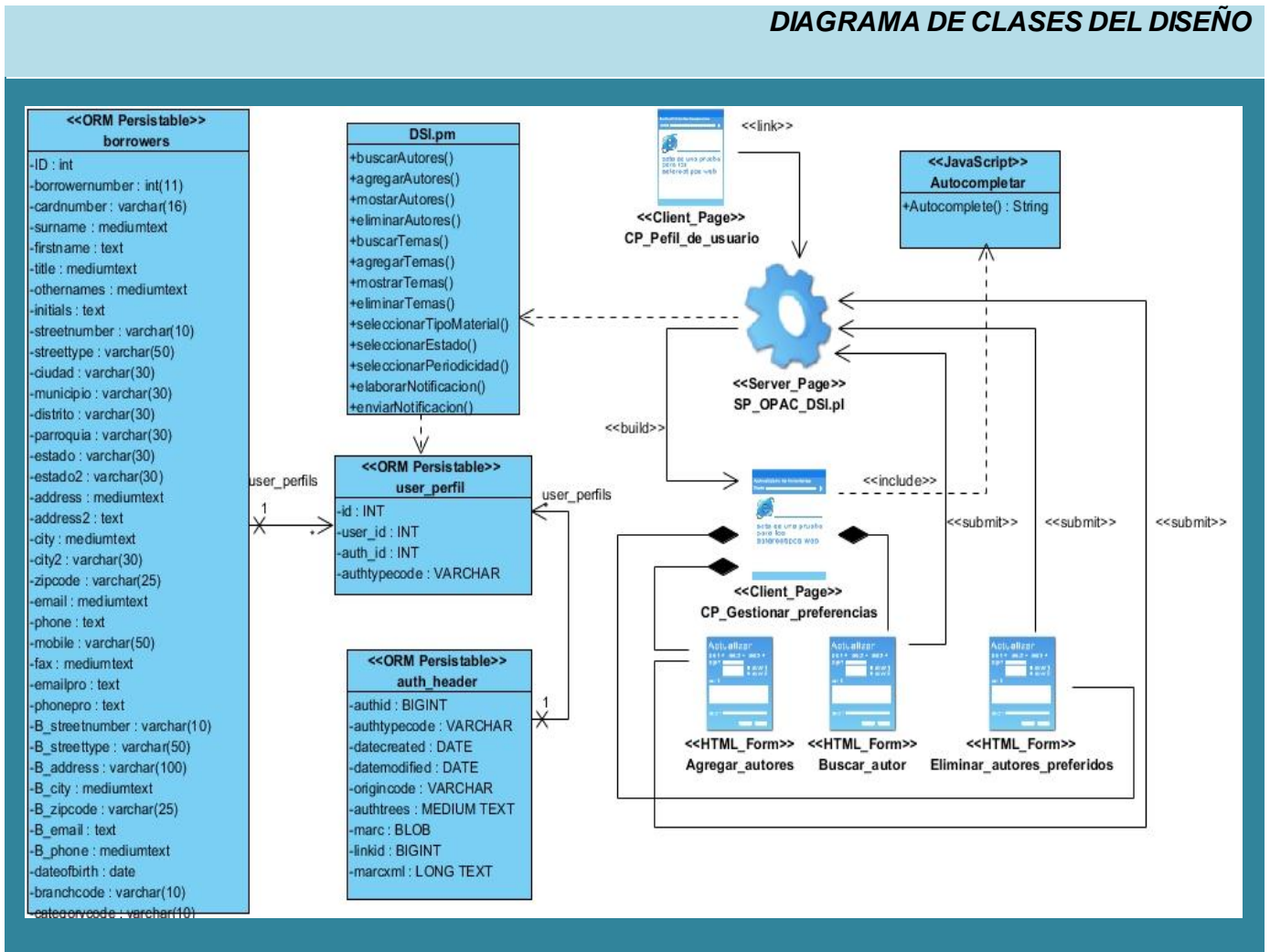


Figura 6. Diagrama de clases del diseño del CU Gestionar autores.

3.4. Modelo de Datos.

El modelo de datos es un lenguaje orientado a describir una base de datos. Típicamente permite describir las estructuras de datos, las restricciones de integridad y las operaciones de manipulación de los datos. Otro enfoque es pensar que un modelo de datos permite describir los elementos de la realidad que intervienen en un problema dado y la forma en que se relacionan esos elementos entre sí. No hay que perder de vista que una base de datos siempre está orientada a resolver un problema

determinado, por lo que los dos enfoques propuestos son necesarios en cualquier desarrollo de software.

3.4.1. Diagrama Entidad-Relación.

Un diagrama o modelo entidad-relación (a veces denominado por su siglas en inglés ER, Entity Relationship, o DER, Diagrama de Entidad-Relación) es una herramienta para el modelado de datos de un sistema de información. Estos modelos expresan entidades relevantes así como sus interrelaciones y propiedades dentro de un sistema para el almacenamiento de datos.

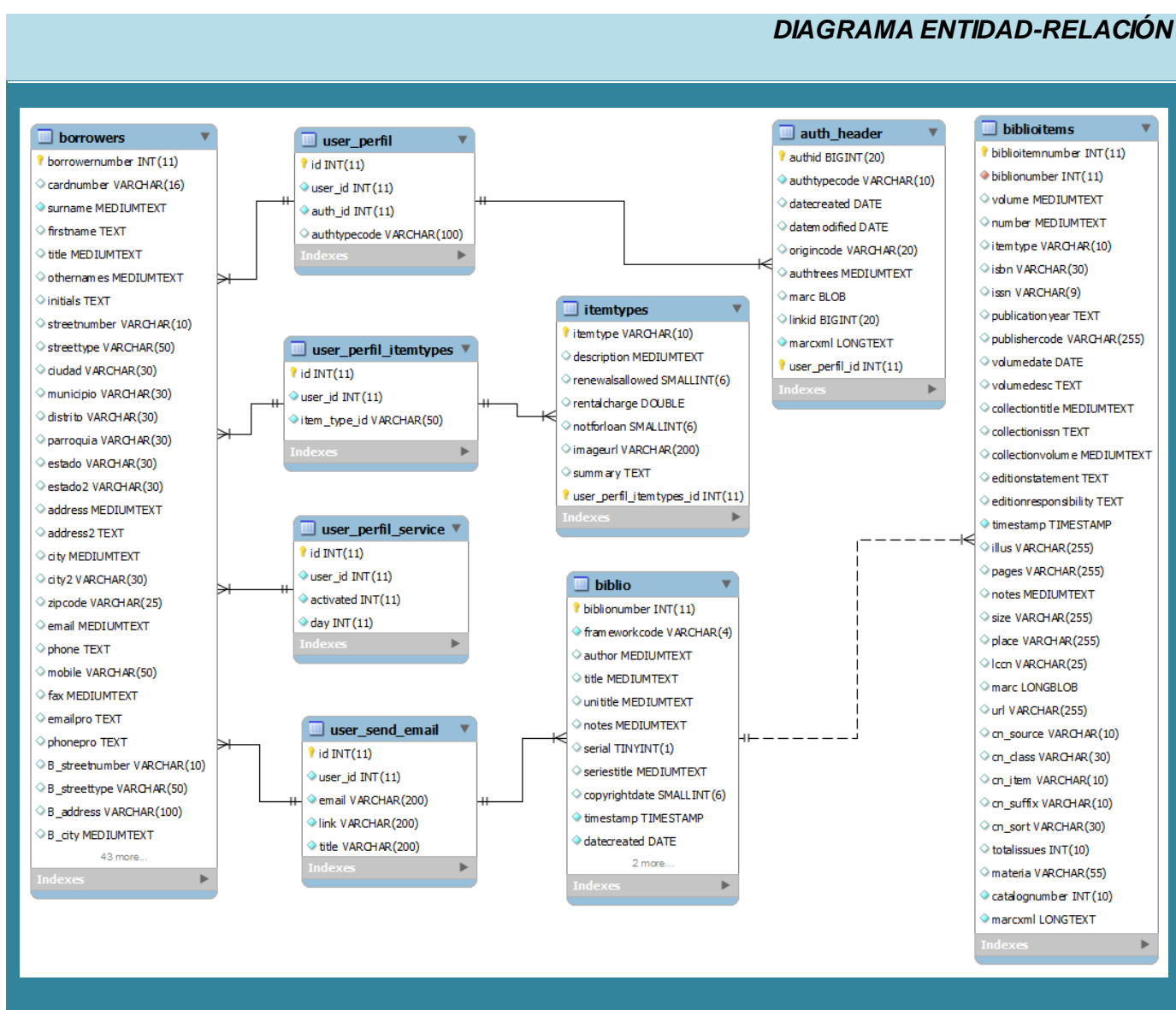


Figura 7. Diagrama Entidad-Relación.

3.5. Arquitectura del sistema.

Las técnicas metodológicas desarrolladas con el fin de facilitar la programación se engloban dentro de la llamada Arquitectura de Software o Arquitectura Lógica. Se refiere a un grupo de abstracciones y patrones que brindan un esquema de referencia útil para la guía en el desarrollo de software dentro de un sistema informático. Así, los programadores, diseñadores, ingenieros y analistas pueden trabajar bajo una línea común que les posibilite la compatibilidad necesaria para lograr el objetivo deseado [Guglielmetti, 2010].

Algunos tipos de arquitectura son más recomendables que otras para ciertas tecnologías, en este caso se utiliza el patrón de diseño Modelo Vista Controlador (MVC), el mismo es capaz de separar los datos de una aplicación, la interfaz de usuario y la lógica de control en tres componentes distintos.

Definición de las partes.

Modelo: Componente encargado del acceso a datos. Modela los datos y el comportamiento detrás de los procesos de negocio. A través de este componente se realizan las consultas a la base de datos, se ejecutan los cálculos de los procesos de negocio, se procesan las diferentes órdenes y se encapsulan los datos y el comportamiento que son independientes de la presentación.

Vista: Se despliega la información de acuerdo con el tipo de cliente, así como los resultados de la lógica de negocios (modelo). Una vista puede ser una página web o una parte de una página.

Controlador: Sirve como una conexión lógica entre la interacción del usuario y los servicios del negocio disponibles. En otras palabras, el controlador actúa como intermediario entre el modelo, la vista y cualquier otro recurso necesario para generar una página [Carrodeguas Oliva; León Piñero, 2009].

Procesamiento del patrón MVC

El procesamiento se lleva a cabo entre sus tres componentes de la siguiente manera: el controlador recibe una orden y decide quién la lleva a cabo en el modelo. Una vez que el modelo (la lógica de negocio) termina sus operaciones devuelve el flujo al controlador y este envía el resultado a la vista [Carrodeguas Oliva; León Piñero, 2009].

PATRÓN MODELO VISTA CONTROLADOR

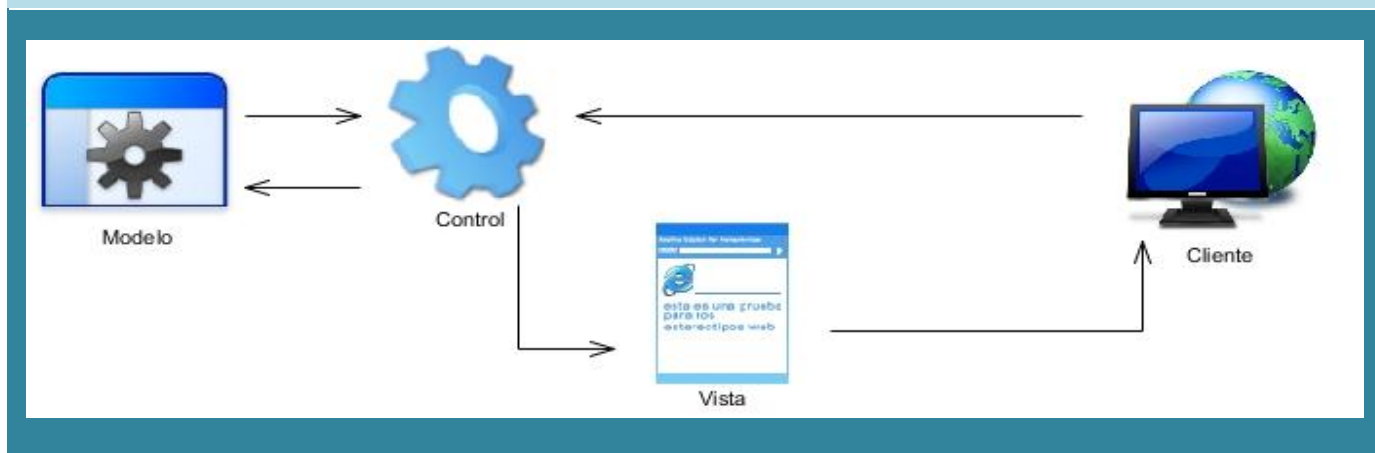


Figura 8. Descripción del Patrón MVC.

3.6. Concepción de la Ayuda.

En el momento que se hace uso del sistema, es necesario que el mismo cuente con diferentes temas de ayuda que le permitan al usuario entender cómo hacer uso de las funcionalidades del mismo con el fin de facilitar el trabajo y de esta forma evitar errores. La ayuda debe estar accesible como parte del menú en la página del módulo del servicio DSI, con el objetivo de que el usuario acceda a la información que necesita en ese momento. Esta cuenta con informaciones generales del módulo, así como explicaciones de las diferentes funcionalidades del subsistema. Cada página de la ayuda muestra cómo realizar aquellas operaciones que estén relacionadas con la posición donde se encuentre el usuario en dicho momento.

3.7. Conclusiones del Capítulo.

En el capítulo se detallaron las principales actividades desarrolladas en el flujo de análisis y diseño, el mismo brinda una visión general del sistema y constituye uno de los más importantes dentro de la etapa de elaboración del software. Para ello, se representó la lógica del negocio mediante los diagramas de análisis del sistema, diagramas de colaboración y de clases del diseño. Se ha representado el modelo de datos a utilizar para la persistencia de la información, la arquitectura base de la solución propuesta y la concepción de la ayuda para el usuario. De esta forma, se crean las condiciones necesarias para el comienzo de la etapa de implementación del sistema.

Capítulo 4. Implementación y Pruebas.

4.1. Introducción al Capítulo.

Este capítulo expone lo referente a los flujos de trabajo Implementación y Pruebas, los cuales son determinantes en el proceso de desarrollo de software. Para ello, se modela el diagrama de componentes, haciendo una representación de la implementación de las clases de diseño en términos de componente y cómo estos se organizan de acuerdo con los nodos específicos en el modelo de despliegue. Además, se realiza un análisis de los casos de prueba, teniendo en cuenta los datos de entrada, resultados esperados y condiciones que deben cumplirse mientras se ejecuta el caso de prueba, con el objetivo de comprobar los errores que pueda tener el sistema, corregirlos y obtener un óptimo funcionamiento.

4.2. Diagrama de despliegue.

El diagrama de despliegue se utiliza para modelar la configuración de los elementos de procesado en tiempo de ejecución y de los componentes, procesos y objetos de software que viven en ellos. Se modelan los nodos físicos y las asociaciones de comunicación que existe entre ellos. En este modelo se tienen en cuenta los protocolos de comunicación entre los nodos, los puertos de acceso, o sea la representación de la configuración que debe tener la red para que sea posible usar todas las funcionalidades que tiene el sistema [Laurencio Pérez; Diéguez García, 2009].

Nodo PC Cliente.

Representa las computadoras que utilizarán los usuarios para interactuar con la aplicación. Establece comunicación con el servidor de aplicaciones a través del protocolo HTTP.

Nodo Impresora.

Representa las impresoras, pues se requiere del uso de las mismas con el objetivo de imprimir los diferentes documentos que se necesiten como reportes, búsquedas, datos de interés para los usuarios y cualquier información disponible.

Nodo Servidor de Aplicaciones.

En este nodo se encuentran los scripts de la aplicación.

Nodo Servidor de Base Datos.

En este nodo se encuentra el Servidor de Base de datos del SIGB de la biblioteca de la UCI.

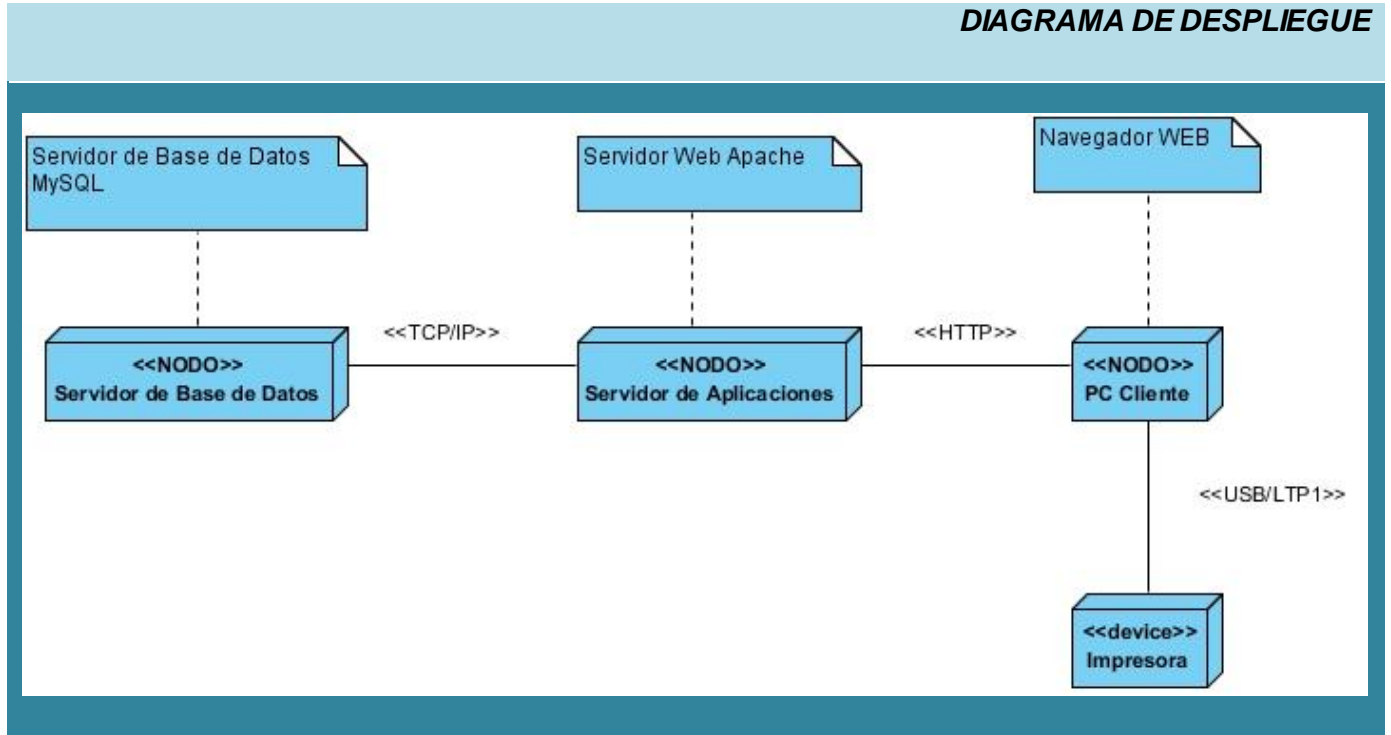


Figura 9. Diagrama de despliegue.

4.3. Modelo de Implementación.

En el flujo de trabajo implementación, a partir del resultado del diseño, se implementa el sistema en términos de componentes, es decir, ficheros de código fuente, scripts y ejecutables, para lo cual se realiza el modelo de implementación. Se incluyen los componentes (que representan al código fuente) y la correspondencia de las clases con estos.

Para lograr un mejor entendimiento del modelo de implementación se hace necesario dividir el sistema en subsistemas, que son una colección de componentes y otros subsistemas usados para estructurar el modelo de implementación. Estos están divididos en paquetes que pueden ser implementados y probados de forma separada. Mediante un diagrama de paquetes es posible lograr la comprensión del modelo de implementación, en él se representan los módulos del sistema como paquetes y las relaciones de dependencia que existen entre ellos. Utilizando un diagrama de componentes es posible reflejar los elementos que representan los ficheros de código fuente, scripts y ejecutables dentro de los paquetes.

4.3.1. Diagrama de Paquetes.

A continuación se representa el diagrama de paquetes general del sistema, donde se denota en rojo el paquete que representa al subsistema del módulo para el servicio DSI:

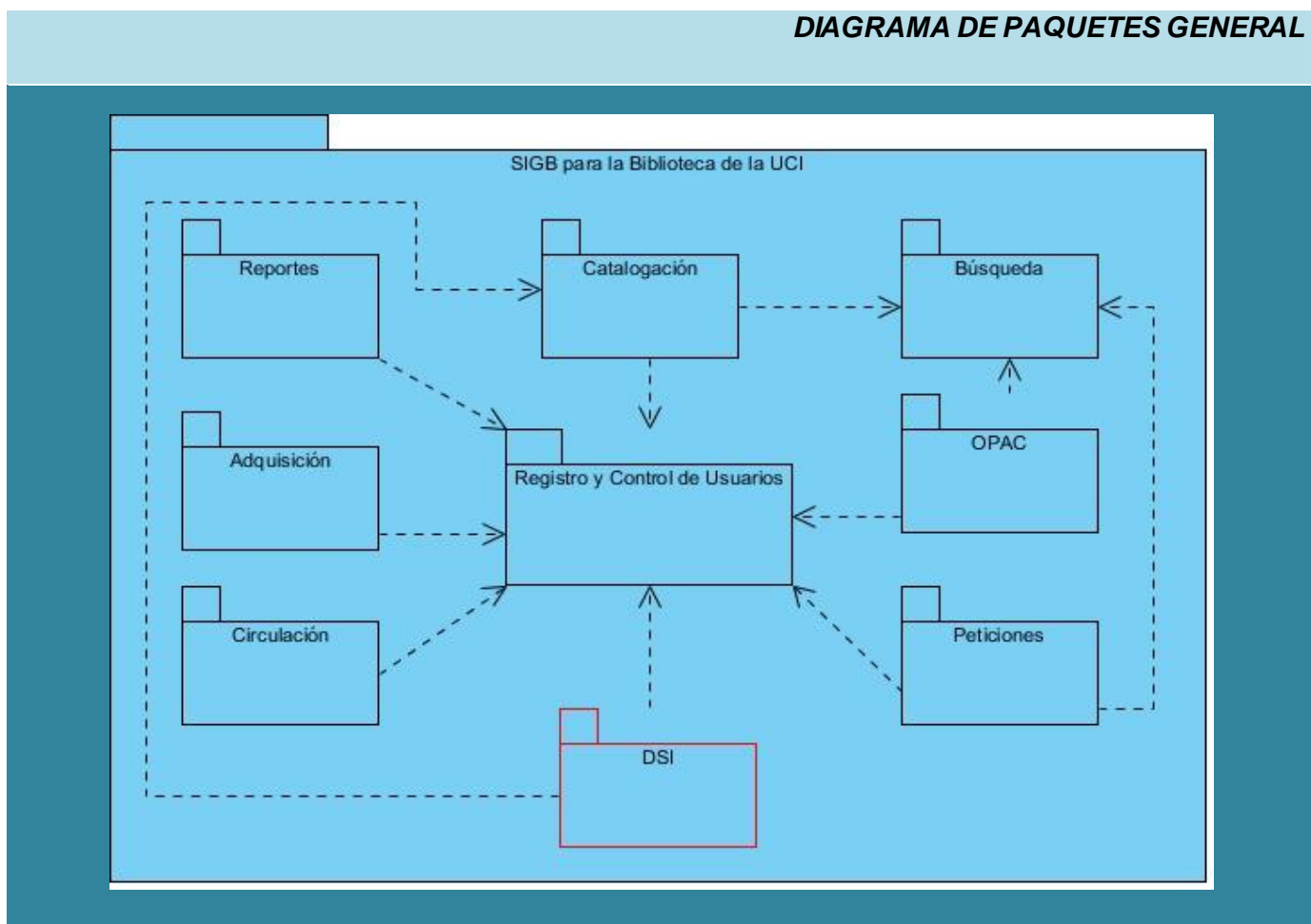


Figura 10. Diagrama de paquetes general.

En la siguiente figura se representa el paquete del módulo DSI como subsistema del SIGB, el cual se comporta como un nuevo sistema. Se denotan los paquetes propios del módulo y las relaciones de dependencia que existen entre sus estructuras básicas:

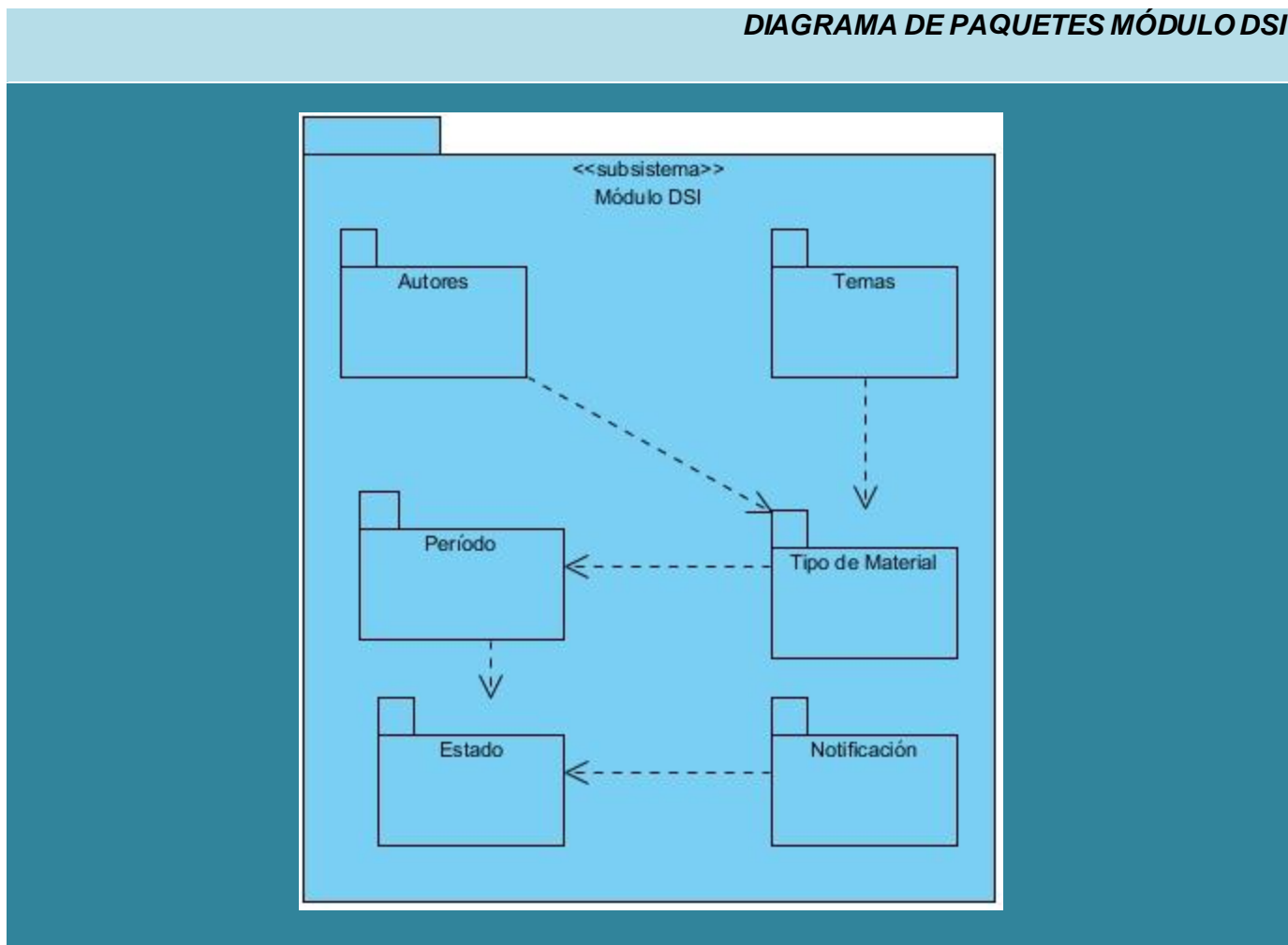


Figura 11. Diagrama de paquetes del módulo DSI.

4.3.2. Diagrama de componentes.

Los diagramas de componentes describen los elementos físicos del sistema y su relación, además muestran las dependencias lógicas entre componentes de software. El diagrama de componente forma parte de la vista física de un sistema, el cual modela la estructura de implementación de la aplicación, su organización en componentes y su despliegue en nodos de ejecución. La vista de implementación se representa con los diagramas de componentes [Laurencio Pérez; Diéguez García, 2009].

El subsistema del módulo DSI está dividido en tres subsistemas de implementación fundamentales: el subsistema Vista, el subsistema Control y el subsistema Modelo, estructurados de forma tal que se agrupan los scripts de acuerdo con el rol que desempeñan dentro del patrón arquitectónico Modelo Vista Controlador. En la siguiente figura se presenta de forma global la agrupación de componentes teniendo en cuenta el patrón y la visión de la arquitectura:

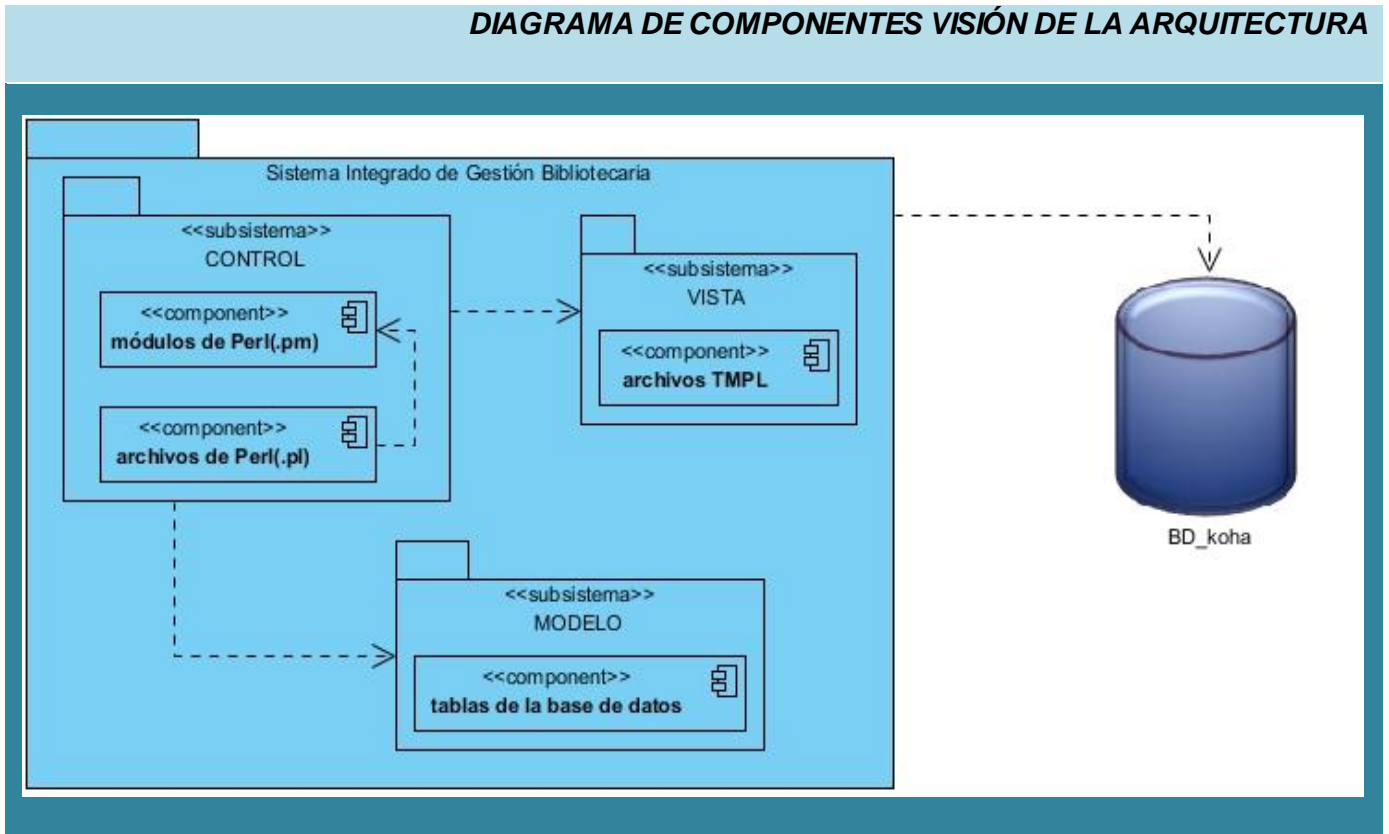


Figura 12. Diagrama de componentes Visión de la Arquitectura.

Con el objetivo de separar los subsistemas del módulo DSI a implementar, se procede a describir cada uno de ellos por separado y se establece la dependencia de los componentes dentro de los subsistemas del módulo, independientes y en conjunto.

El subsistema Vista contiene los componentes necesarios para la interacción del usuario con el sistema, los cuales son manejados por el subsistema Control. A continuación se representan de forma detallada los componentes de dicho subsistema:

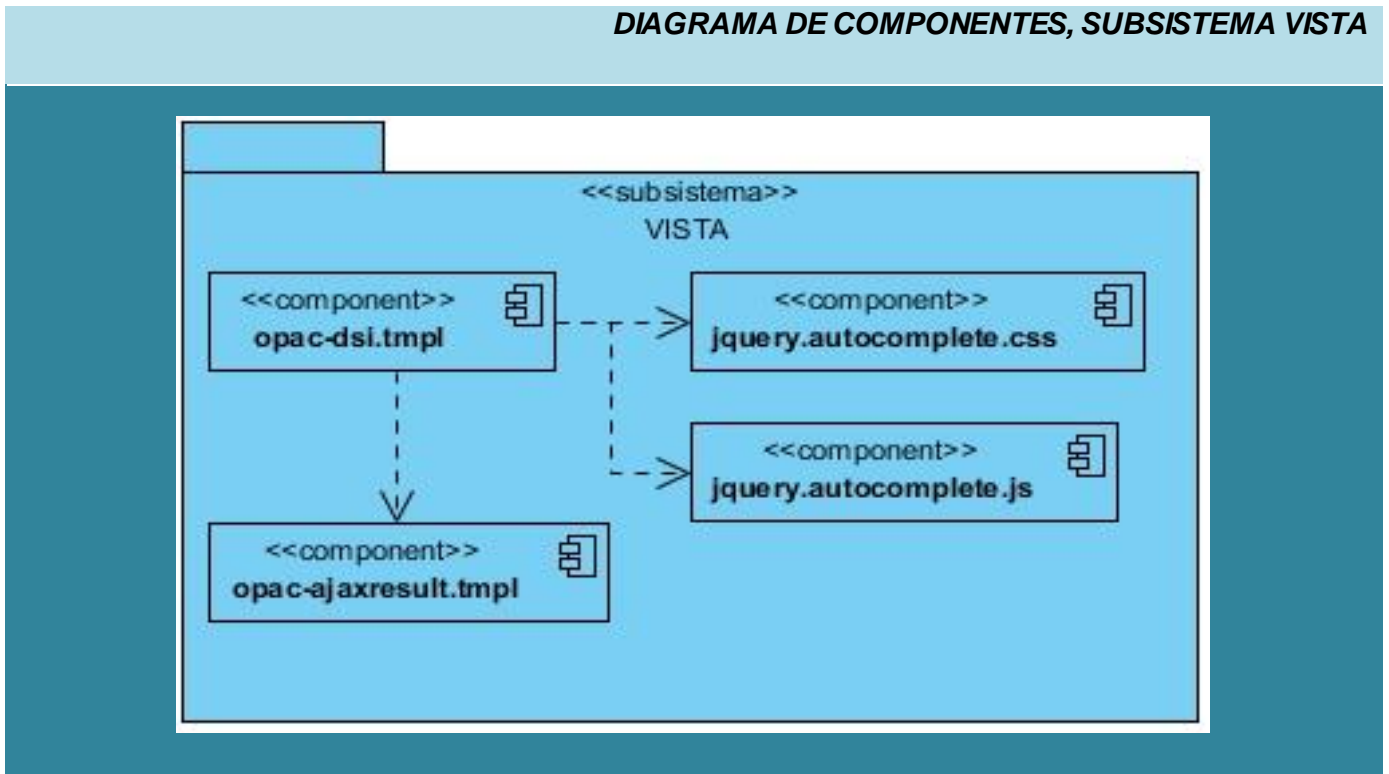


Figura 13. Diagrama de componentes del subsistema Vista.

El subsistema Control, es el rector de las actividades de la aplicación, este contiene los ficheros de código fuente que interactúan con los demás subsistemas coordinando las acciones del software.

DIAGRAMA DE COMPONENTES, SUBSISTEMA CONTROL

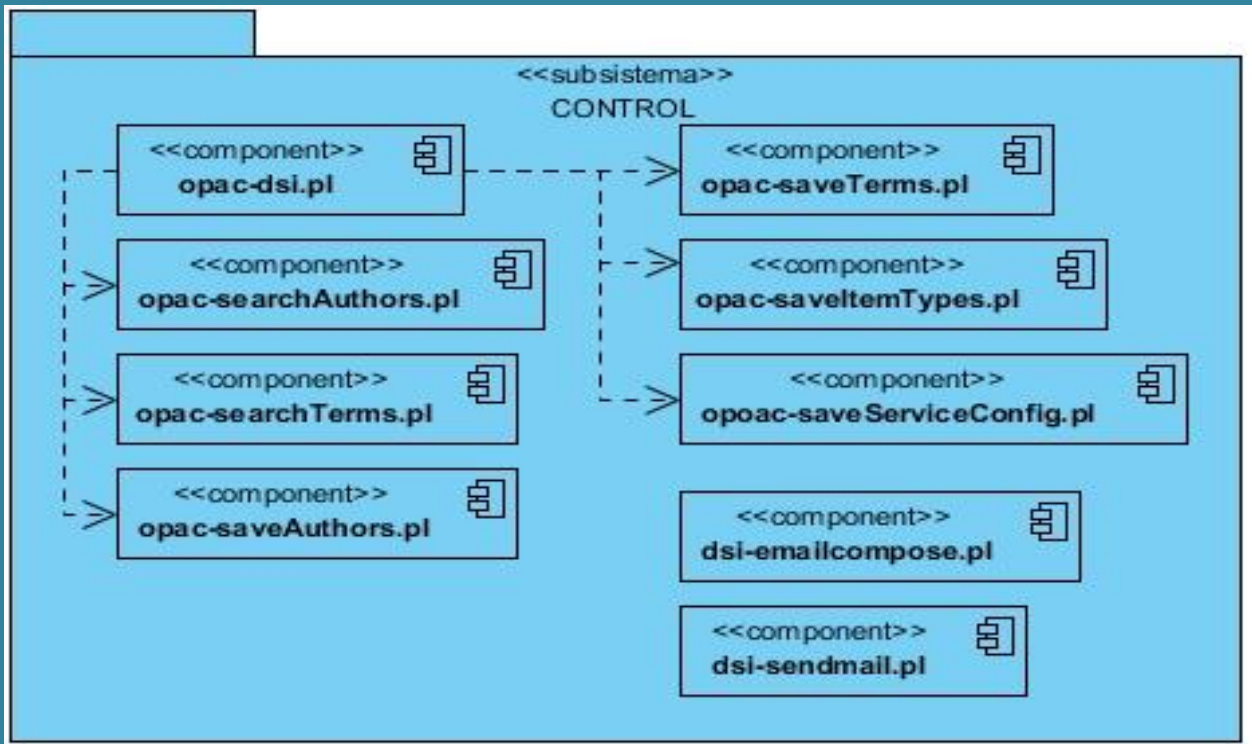


Figura 14. Diagrama de componentes del subsistema Control.

El subsistema Modelo es el encargado de la interacción con la base de datos y contiene la lógica del negocio, para de esta forma gestionar la información con la que trabaja el sistema:

DIAGRAMA DE COMPONENTES, SUBSISTEMA MODELO

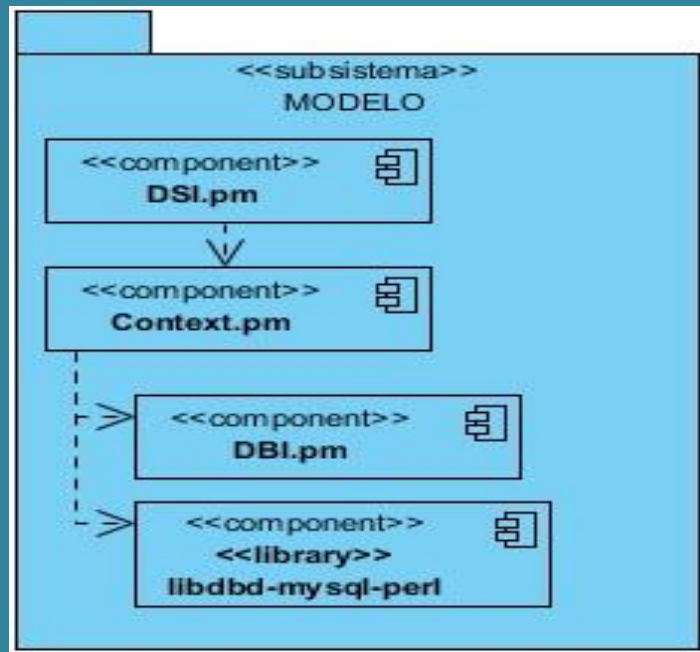


Figura 15. Diagrama de componentes del subsistema Modelo.

A continuación se representa el diagrama de componentes del subsistema o módulo del servicio de DSI, el mismo muestra las dependencias y relaciones entre los componentes de sus subsistemas vistos como paquetes:

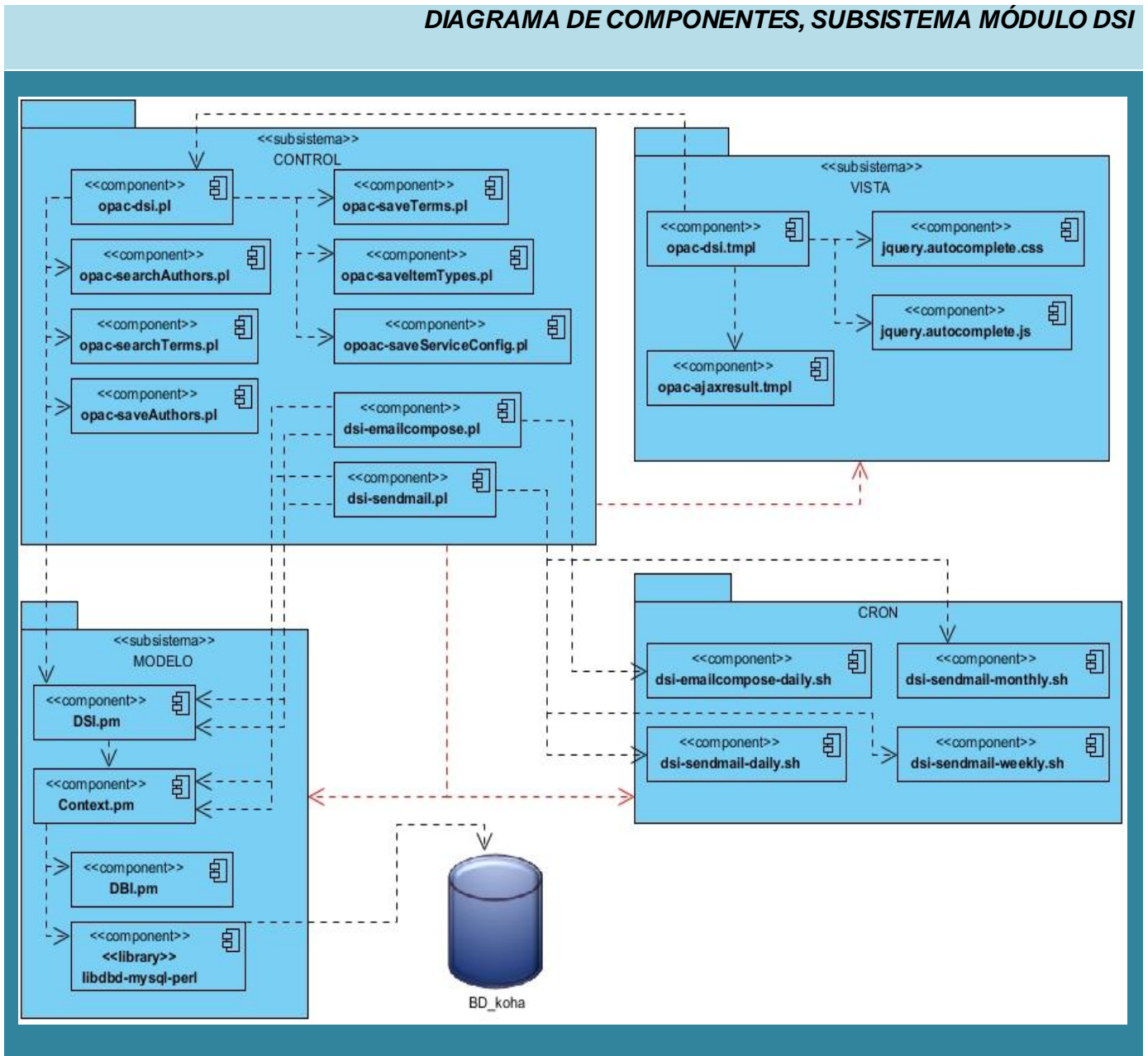


Figura 16. Diagrama de componentes del subsistema Módulo DSI.

Por último, es necesario describir el papel que juega el paquete CRON representado en el diagrama de componentes del módulo DSI. Este representa en los sistemas operativos GNU/Linux basados en UNIX, un administrador regular de procesos en segundo plano como el equivalente a Tareas Programadas de Windows, el cual puede ejecutar scripts a intervalos regulares definidos (cada hora, día, semana o mes). Constituye la vía esencial para que el módulo DSI logre el envío de notificaciones atendiendo al período establecido por los usuarios en sus perfiles. Los procesos que deben ejecutarse y la hora en la que deben hacerlo se especifican en el fichero crontab.

Como parte de los procesos que necesita ejecutar el sistema en determinado período de tiempo, los componentes del paquete CRON se encargan de su realización. Este paquete no forma parte de la arquitectura que plantea el patrón MVC, sin embargo, es necesaria su representación dentro del diagrama de componentes del módulo DSI, pues en su estructura se ubican los scripts que son importantes para realizar las tareas esenciales del módulo. (Ver **Anexo 4**).

4.4. Modelo de Pruebas.

Desarrollar sistemas con calidad debe ser una premisa para todo grupo de desarrollo de software, en este sentido, se hace necesario realizar pruebas para evaluar en qué grado el sistema implementado cumple con las expectativas del cliente. La prueba de software es un elemento crítico para la garantía de la calidad del software y representa una revisión final de las especificaciones del diseño y de la codificación.

Existen dos formas de pruebas fundamentales: el procedimiento de Caja Negra y el de Caja Blanca. El primero se refiere a las pruebas que se llevan a cabo sobre la interfaz del software, por lo que los casos de prueba pretenden demostrar que las funciones del software son operativas, que la entrada se acepta de forma adecuada y que se produce una salida correcta, así como que la integridad de la información externa se mantiene. Esta prueba examina algunos aspectos del modelo fundamentalmente del sistema sin tener mucho en cuenta la estructura interna del software [Ortega Retureta; Roselló Carrazana, 2010].

La técnica de Partición Equivalente es una de las más efectivas pues permite examinar los valores válidos e inválidos de las entradas existentes en el software, descubre de forma inmediata una clase de errores que, de otro modo, requerirían la ejecución de muchos casos antes de detectar el error genérico. La partición equivalente se dirige a la definición de casos de pruebas que descubran clases

de errores, reduciendo así el número de clases de prueba que hay que desarrollar [Ortega Retureta; Roselló Carrazana, 2010].

Las pruebas de integración se realizan para comprobar que un sistema trabaja correctamente en operaciones conjuntas y es posible aplicarlas en la actual solución para verificar que el nuevo subsistema se integra con el sistema correctamente. Estas pruebas identifican problemas de interfaces entre unidades, falta de coherencia entre lo que se espera de una unidad y lo que se ofrece y hacen énfasis en la interacción y no en el funcionamiento individual. Para aplicar este tipo de pruebas fue necesario obtener los distintos valores que se envían desde la interfaz del módulo al sistema, verificar si eran correctos y comprobar la extracción satisfactoria de los datos que se manejan y que son vitales para el trabajo del nuevo subsistema.

La estrategia ascendente dentro de las pruebas de integración fue la utilizada. Esta permite ver el sistema como un árbol, se prueba desde las hojas hacia la raíz, donde las hojas son las funcionalidades básicas que se ejecutan y se requieren sus salidas para que trabaje la raíz que constituye al sistema. De la forma en que el módulo trabaja es posible realizar pruebas de integración fácilmente, los datos mostrados en la interfaz del usuario son extraídos de la base de datos donde ya fueron guardados por otros subsistemas, por lo tanto, solo se muestra información que coincide con la entrada por el usuario, no se recogen datos nuevos para persistirlos.

4.4.1. Resultados de las pruebas.

Para la realización de las pruebas, se estableció como estrategia de trabajo la ejecución de las mismas en dos iteraciones. La primera iteración se centró en los casos de uso que describen las funcionalidades que interactúan directamente con el usuario, estos son:

- CU_1 Gestionar autores.
- CU_2 Gestionar temas.
- CU_3 Seleccionar tipo de material.
- CU_4 Seleccionar estado.
- CU_5 Seleccionar periodicidad.

En la segunda iteración se realizaron las pruebas de integración y se culminó con los casos de prueba para los casos de uso:

- CU_6 Elaborar notificación.
- CU_7 Enviar notificación.

Fueron encontradas cuatro no conformidades al finalizar la primera iteración y tres en la segunda, todas se resolvieron satisfactoriamente. Las pruebas de integración no arrojaron problemas entre la interrelación del sistema en general y el nuevo subsistema o módulo para el servicio de DSI. (Ver Anexo 7).

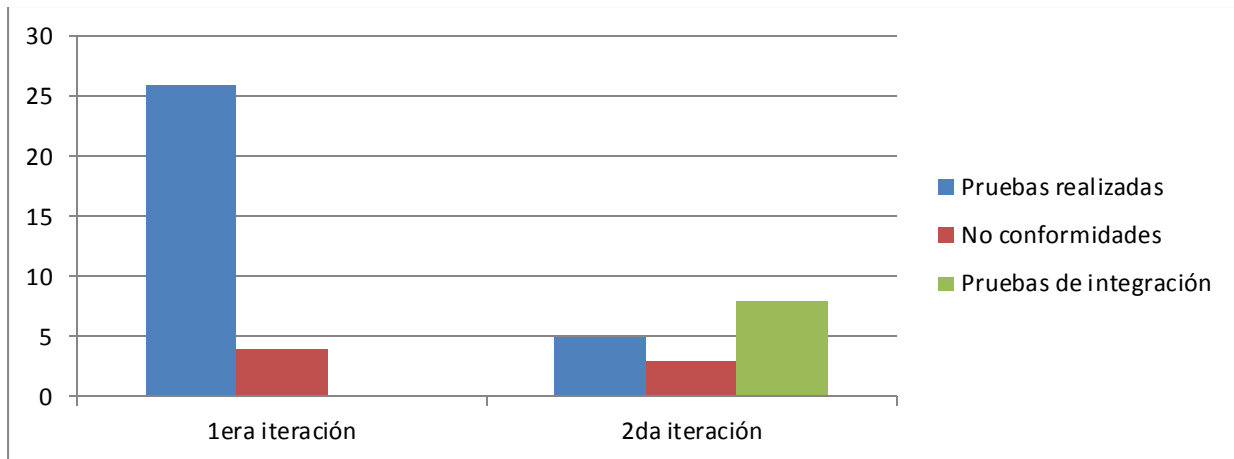


Figura 17. Gráfico de resumen de las pruebas.

Se presenta a continuación el resumen de los casos de prueba de cada caso de uso, donde se describen los valores de entrada y los resultados observados aplicando el método de prueba de Caja Negra y utilizando la Técnica de Partición de Equivalencia. El CU_1 coincide en todas las pruebas y resultados con el CU_2 debido a su similitud, por lo que no es necesario presentarlo aquí.

CU_1 Gestionar autores.		
Entrada	Resultados	Condiciones
El usuario introduce menos de tres caracteres para la búsqueda de autores en un cuadro de texto.	El sistema no muestra la lista de resultados con auto-completamiento.	Se repite la operación hasta que el usuario entre como mínimo tres caracteres para la búsqueda de autores.
El usuario selecciona un resultado de la búsqueda con auto-completamiento y no presiona el botón Guardar.	El sistema no guarda el autor seleccionado.	Se repite la operación hasta que el usuario presione el botón Guardar.
El usuario selecciona un resultado de la búsqueda con	El sistema guarda el autor exitosamente.	No estaba almacenado con anterioridad el autor.

auto-completamiento y presiona el botón Guardar.		
El usuario retira la selección de un autor guardado para eliminarlo y no presiona el botón Guardar.	El sistema no elimina el autor seleccionado.	Se repite la operación hasta que el usuario presione el botón Guardar.
El usuario desea visualizar todos los autores preferidos guardados.	El sistema muestra los autores preferidos guardados.	El usuario accedió a la interfaz del servicio.

Tabla 4. Descripción de los casos de prueba del CU Gestionar autores.

CU_3 Seleccionar tipo de material.		
Entrada	Resultados	Condiciones
El usuario selecciona los tipos de materiales y no presiona el botón Guardar.	El sistema no guarda la selección realizada.	Se repite la operación hasta que el usuario presione el botón Guardar.
El usuario desea retirar la selección de un tipo de material deseado.	El sistema no retira la selección del tipo de material especificado.	Se repite la operación hasta que el usuario presione el botón Guardar.
El usuario desea visualizar los tipos de materiales deseados.	El sistema muestra los tipos de materiales.	El usuario accedió a la interfaz del servicio.

Tabla 5. Descripción de los casos de prueba del CU Seleccionar tipo de material.

CU_4 Seleccionar estado.		
Entrada	Resultados	Condiciones
El usuario selecciona el estado deseado ya sea activado o desactivado y no presiona el botón Guardar.	El sistema no guarda la selección realizada.	Se repite la operación hasta que el usuario presione el botón Guardar.

El usuario selecciona el estado deseado ya sea activado o desactivado y presiona el botón Guardar.	El sistema guarda la selección realizada.	El usuario cambió la preferencia del estado.
El usuario desea cambiar la selección del estado.	El sistema no cambia la selección del período.	Se repite la operación hasta que el usuario presione el botón Guardar.

Tabla 6. Descripción de los casos de prueba del CU Seleccionar estado.

CU_5 Seleccionar periodicidad.		
Entrada	Resultados	Condiciones
El usuario selecciona el período deseado ya sea diario, semanal o mensual.	El sistema guarda la selección realizada.	El usuario cambió la preferencia del período.
El usuario selecciona el período deseado.	El sistema no guarda la selección realizada.	Se repite la operación hasta que el usuario presione el botón Guardar.
El usuario desea cambiar la selección del período.	El sistema no cambia la selección del período.	Se repite la operación hasta que el usuario presione el botón Guardar.

Tabla 7. Descripción de los casos de prueba del CU Seleccionar periodicidad.

CU_6 Elaborar notificación.		
Entrada	Resultados	Condiciones
Se ejecuta el script de elaboración de notificaciones.	El sistema guarda las notificaciones elaboradas.	Existen nuevos ejemplares con características que coinciden con algún criterio de selección en los perfiles de los usuarios.
Se ejecuta el script de	El sistema no guarda ninguna	El sistema no cuenta con

elaboración de notificaciones.	notificación para enviar.	nuevos ejemplares con fecha actual a la ejecución, o no coinciden las características de los existentes con ningún criterio de selección en los perfiles de los usuarios.
Se ejecuta el script de elaboración de notificaciones.	El sistema no guarda ninguna notificación para enviar.	Se perdió la conexión con la base de datos.

Tabla 8. Descripción de los casos de prueba del CU Elaborar notificación.

CU_7 Enviar notificación.		
Entrada	Resultados	Condiciones
Se ejecuta el script de envío de notificaciones.	Se envían las notificaciones por correo.	Existen notificaciones guardadas y que corresponden con el período establecido en el perfil de los usuarios para su envío. El servidor de correo está funcionando y accesible.
Se ejecuta el script de envío de notificaciones.	No se envían las notificaciones por correo.	El sistema no pudo encontrar el servidor de correo debido a que no está disponible o se ha perdido la conexión.
Se ejecuta el script de envío de notificaciones.	No se envía ninguna notificación por correo.	Se perdió la conexión con la base de datos.
Se ejecuta el script de envío de notificaciones.	No se envía ninguna notificación por correo.	No existían notificaciones guardadas, o no corresponde con el período establecido en el perfil de los usuarios para su envío.

Tabla 9. Descripción de los casos de prueba del CU Enviar notificación.

4.5. Conclusiones del Capítulo.

En este capítulo se abordaron los aspectos fundamentales de los flujos de trabajo Implementación y Pruebas, donde se realizó la modelación del diagrama de despliegue que especifica la distribución física por nodos del sistema en tiempo de ejecución y sus características. Además, se modeló el diagrama de componentes y se representó la relación entre los mismos lo que permitió conocer de qué manera se ubicarán los ficheros que contendrán el código fuente. Por último, en el flujo de trabajo Pruebas, se realizó la descripción de los casos de prueba pertenecientes a los casos de uso documentados en el trabajo, se pudo verificar la calidad de la solución e identificar las no conformidades referentes al módulo.

Conclusiones

La presente investigación se centró en el desarrollo del módulo para el servicio de disseminación selectiva de información del SIGB de la biblioteca de la UCI, con la finalidad de facilitar la interacción del usuario con el sistema, y recibir periódicamente notificaciones acerca de nuevos registros bibliográficos adquiridos que respondan a sus propias necesidades informativas, recogidas mediante un perfil individual disponible en las preferencias del catálogo en línea de la institución. Durante el avance del mismo se cumplieron los objetivos propuestos y se arribó a conclusiones tales como:

- Conceptualizar a los Sistemas Integrados de Gestión Bibliotecaria y al servicio de disseminación selectiva de información permitió establecer la relación existente entre ambos y conocer sus características principales.
- Se realizó un estudio de las soluciones similares que permitió extraer las funcionalidades necesarias a incorporar, donde se demostró la factibilidad de implementar sobre el SIGB existente en la UCI y no optar por la incorporación de otro.
- El estudio realizado como parte de la investigación sirvió de apoyo en la toma de decisiones con vista a un desarrollo eficiente del módulo.
- Se diseñó e implementó el conjunto de clases que dan cumplimiento a los requisitos funcionales del módulo, se probó la integración del mismo al SIGB de la UCI y se aplicaron buenas prácticas de diseño e implementación para responder a la problemática existente, incorporando nuevas funcionalidades al sistema desplegado y en funcionamiento.
- Se demostró a partir de las diferentes iteraciones de pruebas practicadas al software, que este satisface los requisitos funcionales y no funcionales del módulo.

Recomendaciones

Teniendo en cuenta que la constancia y atención hacen que las obras brinden mejores servicios, y en aras de lograr un mejor acabado al servicio de disseminación selectiva de información en la biblioteca de la UCI se ofrecen las siguientes recomendaciones:

- Propiciar en el futuro que el servicio no solo sea automático, sino que participen especialistas que diseminen información con un nivel de selección más alto.
- Ampliar el sistema para que consulte otras fuentes de información externas, asegurando un flujo de disseminación de información selectiva más amplio.
- Elaborar un manual de ayuda que le permita al usuario comprender mejor el funcionamiento del módulo.
- Continuar preparando al equipo actual de trabajo en los diferentes componentes y características del sistema.
- Implantar el SIGB con el nuevo módulo en bibliotecas del país con características similares a la biblioteca de la UCI.
- Poner esta investigación a disposición de todos los interesados.

Referencias bibliográficas.

Álvarez, Miguel Ángel (2010): “*Qué es HTML.*” [En línea] Enero de 2001. [Disponible en: <http://www.desarrolloweb.com/articulos/que-es-html.html>] [Citado el: 5 de abril de 2012].

Berovides Hernández, Anmaris; Galán Domínguez, Esther; Larrazabal Capaz, Daysell; Pardo Suárez, Delis R.; Pérez Aismarez, Yacselys (2009): “*Diseño del servicio de Diseminación Selectiva de la Información para la biblioteca del CEDEM.*” Revista Especializada en Temas de Población, Año 5 Número 10, 2009 ISSN: 1817- 4078. [Disponible en: <http://www.cedem.uh.cu/sites/default/files/5.pdf>] [Citado el: 6 de mayo de 2012].

Carrodegas Oliva, Rogelio; León Piñero, Roberlay (2009): “*Automatización del módulo de Referencia de la Biblioteca de la Universidad de las Ciencias Informáticas.*” Trabajo de Diploma. Universidad de las Ciencias Informáticas, 2009. [Disponible en: http://repositorio_institucional.uci.cu/jspui/bitstream/ident/TD_2381_09/1/TD_2381_09.pdf].

Concepto de JavaScript (2008): [En línea] Noviembre de 2011. [Disponible en: https://developer.mozilla.org/index.php?title=Es/Gu%C3%ADa_JavaScript_1.5/Concepto_de_JavaScript] [Citado el: 25 de marzo de 2012].

Escobedo Molina, Armando (2009): “*Sistemas Integrados de Bibliotecas de Código Abierto: una Descripción.*” Tesina. Escuela Nacional de Biblioteconomía y Archivonomía. México, D. F. 2009. [Disponible en: <http://eprints.rclis.org/bitstream/10760/15872/1/Tesina%20final.pdf>] [Citado el: 2 de abril del 2012].

Feus Pérez, Yusdanis (2009): “*Adaptación del Módulo de Circulación del Sistema Integrado de Gestión Bibliotecaria Koha a la Biblioteca Nacional de Cuba José Martí.*” Trabajo de Diploma. Universidad de las Ciencias Informáticas, 2009. [Disponible en: http://repositorio_institucional.uci.cu/jspui/bitstream/ident/TD_2380_09/1/TD_2380_09.pdf].

Filippi, Ailé (2008): “*Análisis del Sistema Integrado de Gestión Bibliotecaria Koha. Modelo de desarrollo, valor para la biblioteca, componentes tecnológicos y oportunidades de migración.*” Presentación. [En línea] Abril de 2008. [Disponible en: <http://es.scribd.com/doc/25185158/Analisis-del-Sistema-Integrado-de-Gestion-Bibliotecaria-Koha>] [Citado el: 20 de abril de 2012].

García, Joaquín (2005): “UML: Diagramas UML. ¿Qué es UML? Análisis y Diseño. Ingeniería del Software”. [En línea] Julio de 2005. [Disponible en: <http://www.ingenierossoftware.com/analisisydiseno/uml.php>] [Citado el: 10 de marzo de 2012].

García Melero, L.A.; García Camarero, E. (1999): “Automatización de bibliotecas”. Madrid: Arco/Libros, 1999. [Disponible en: <http://www.slideshare.net/Carminis/sigb-4790288>]. [Citado el: 2 de abril de 2012].

Gómez Hernández, J.A. (2001): “Qué son y de dónde vienen los portales de internet: comparativa de algunos portales de ocio” En: *Educación y Biblioteca*, nº 123 (Dossier: Ocio y tecnología en las bibliotecas públicas), mayo-junio, 2001 pp. 73-79. [Disponible en: <http://dialnet.unirioja.es/servlet/articulo?codigo=127331>] [Citado el: 6 de marzo de 2012]. ISSN 0214-7491.

Guglielmetti, Marcos (2010): “Definición de Arquitectura Software - Significado y definición de Arquitectura Software”. [En línea] 2004. [Disponible en: <http://www.mastermagazine.info/termino/3916.php>] [Citado el: 9 de marzo de 2012].

Gutiérrez Vallado, O. (2008): “Diseminación Selectiva de Información en Bibliotecas Universitarias”. Informe académico. Universidad Nacional Autónoma de México, 2008. [Disponible en: <http://www.filos.unam.mx/LICENCIATURA/bibliotecologia/textos-apoyo-docencia/gutierrez-vallado-olga.pdf>]. [Citado el: 4 de marzo de 2012].

Jacobson, Ivar (2004): “El proceso Unificado de Desarrollo de Software”. CUJAE, La Habana, 2004. 438 p. [Disponible en: <http://monografias.umcc.cu/monos/2007/informatica/m07245.pdf>] [Citado el: 4 de abril de 2012].

Jacobson, Ivar; Rumbaugh, James; Booch, Grady (2000): “El proceso unificado de desarrollo”. Addison Wesley, 2000. Capítulo 9, Páginas 205-254.

Laurencio Pérez, Daniuska; Diéguez García, Yunieski (2009): “Sistema de Gestión de la Trayectoria Productiva de cada estudiante en la facultad 1”. Trabajo de Diploma. Universidad de las Ciencias Informáticas, 2009. [Disponible en: http://repositorio_institucional.uci.cu/jspui/bitstream/ident/TD_1903_09/1/TD_1903_09.pdf].

Marquina, Julián (2009): “Curso Introducción al sistema de gestión bibliotecaria KOHA”. [En línea] Febrero de 2009. [Disponible en: <http://www.recbib.es/formacion/cursos-introduccion-al-sistema-de-gestion-bibliotecaria-koha>] [Citado el: 6 de mayo de 2012].

Ortega Retureta, Laura S.; Roselló Carrazana, Adnier (2010): “Desarrollo del módulo Adquisición del Sistema Integrado de Gestión Bibliotecaria Koha para la Biblioteca Nacional José Martí”. Trabajo de Diploma. Universidad de las Ciencias Informáticas, 2010. [Disponible en: http://repositorio_institucional.uci.cu/jspui/bitstream/ident/TD_03466_10/1/TD_03466_10.pdf].

Ponjuan Cance, G (2007): “Surgimiento de la Diseminación Selectiva de la información en Cuba”. [Carta]. Ciudad de la Habana: CNICM, 2007.

Porras Herrera, Lityuen A; Veloz Morales, Dargel (2008): “Implantación de un SIGB en la biblioteca de la UCI”. Trabajo de Diploma. Universidad de las Ciencias Informáticas, 2008. [Disponible en: http://repositorio_institucional.uci.cu/jspui/bitstream/ident/TD_1257_08/1/TD_1257_08.pdf].

Saide, R. (2003): “Diseminación selectiva de información (DSI): implementación para el Departamento de Ecología Humana del CINVESTAV-Mérida”. Biblioteca Universitaria, Nueva Época. Vol. 6, No.1, 40-46pp, 2003. [Disponible en: <http://www.cedem.uh.cu/sites/default/files/5.pdf>]. [Citado el: 2 de abril de 2012].

Sánchez Pereyra, Antonio (2009): “La diseminación selectiva de información a través de internet: propuesta de boletín electrónico de información bibliográfica especializada en economía, utilizando la base de datos CLASE”. Universidad Nacional Autónoma de México, 2009. [En línea]. Disponible en: <http://www.dgbiblio.unam.mx/servicios/dgb/publicdgb/bole/fulltext/voll1/clase.html> [Citado el: 3 de abril de 2012].

Saorín Pérez, Tomás (2002): “Modelo Conceptual para la Automatización de Bibliotecas en el contexto digital”. Tesis doctoral. Facultad de Ciencias de la Documentación Departamento de Información y Documentación. Universidad de Murcia, 2002. [Disponible en: <http://www.digitum.um.es/jspui/bitstream/10201/84/1/SaorinPerez.pdf?sequence=1>]. [Citado el: 2 de abril de 2012].

Van Der Henst, Christian S. (2001): “¿Qué es el CGI?”, [En línea] 2001. [Disponible en: <http://www.maestrosdelweb.com/editorial/cgiintro/>] [Citado el: 10 de marzo de 2012].

Vieira, M. (2003): “*Reflexiones sobre la diseminación de informaciones*”. Ciencias de la Información. Vol. 34, No. 1, 25-32pp, 2003.

Wall, Larry (2008): “*The Perl Programming Language - www.perl.org*”. [En línea] 2008. [Disponible en: <http://www.perl.org/>] [Citado el: 4 de abril de 2012].

Bibliografía consultada.

Álvarez, Miguel Ángel (2010): “Qué es Perl”. [En línea] Sep. 2001. [Disponible en: <http://www.desarrolloweb.com/articulos/541.php>] [Citado el: 5 de abril de 2012].

Álvarez, Miguel Ángel (2010): “Qué es HTML.” [Disponible en: <http://www.desarrolloweb.com/articulos/que-es-html.html>] [Citado el: 5 de abril de 2012].

Arriola Navarrete, Óscar; Butrón Yáñez, Katya. “Sistemas integrales para la automatización de bibliotecas basados en software libre”. **ACIMED.** *Revista cubana de los profesionales de la información y la comunicación en salud.*, Vol.18 No.6, Dic. 2008, pp. 9-10. [Disponible en: <http://www.uh.cu/static/documents/TD/Sistemas%20integrales%20automatizacion.pdf>] [Citado el: 5 de abril de 2012].

Berovides Hernández, Anmaris; Galán Domínguez, Esther; Larrazabal Capaz, Daysell; Pardo Suárez, Delis R.; Pérez Aismarez, Yacselys (2009): “Diseño del servicio de Diseminación Selectiva de la Información para la biblioteca del CEDEM”. *Revista Especializada en Temas de Población*, Año 5 Número 10, 2009 ISSN: 1817- 4078. [En línea] [Disponible en: <http://www.cedem.uh.cu/Revista/portada.html>] [Citado el: 6 de mayo de 2012].

“Concepto de JavaScript - MDC,” [En línea] 2008. [Disponible en: https://developer.mozilla.org/index.php?title=Es/Gu%C3%ADa_JavaScript_1.5/Concepto_de_JavaScript] [Citado el: 6 de abril de 2012].

Eclipse Perl Integration (2012): [En Línea] [Disponible en: <http://www.epic-ide.org>] [Citado el: 11 de mayo de 2012].

Escobedo Molina, Armando (2009): “Sistemas Integrados de Bibliotecas de Código Abierto: una Descripción”. Tesina. Escuela Nacional de Biblioteconomía y Archivonomía. México, D. F. 2009. [Disponible en: <http://eprints.rclis.org/bitstream/10760/15872/1/Tesina%20final.pdf>] [Citado el: 2 de abril del 2012].

Feus Pérez, Yusdanis (2009): “Adaptación del Módulo de Circulación del Sistema Integrado de Gestión Bibliotecaria Koha a la Biblioteca Nacional de Cuba José Martí”. Trabajo de Diploma. Universidad de las Ciencias Informáticas, 2009. [Disponible en: http://repositorio_institucional.uci.cu/jspui/bitstream/ident/TD_2380_09/1/TD_2380_09.pdf].

Guglielmetti, Marcos (2010): “Definición de Arquitectura Software - Significado y definición de Arquitectura Software”. [Disponible en: <http://www.mastermagazine.info/termino/3916.php>] [Citado el 9 de marzo de 2012].

Jacobson, Ivar (2004): “El proceso Unificado de Desarrollo de Software”. La Habana, 2004. 438 p.

Jacobson, Ivar; Rumbaugh, James; Booch, Grady (2000): “El proceso unificado de desarrollo”. Addison Wesley, 2000. Capítulo 9, Páginas 205-254.

MySQL [En línea] [Disponible en: http://www.mysql.com/?bydis_dis_index=1] [Citado el 10 de Febrero de 2012].

Saorín Pérez, Tomás (2002): “Modelo Conceptual para la Automatización de Bibliotecas en el contexto digital”. Tesis doctoral. Facultad de Ciencias de la Documentación Departamento de Información y Documentación. Universidad de Murcia, 2002.

Van Der Henst, Christian S. (2001): “¿Qué es el CGI?”, [En línea] 2001. [Disponible en: <http://www.maestrosdelweb.com/editorial/cgiintro/>] [Citado el: 10 de marzo de 2012].

Wall, Larry (2008): “The Perl Programming Language - www.perl.org”. [En línea] 2008. [Disponible en: <http://www.perl.org/> [Citado el: 4 de abril de 2012].

Glosario de términos

Biblioteca: Local donde se tiene considerable número de libros ordenados para la lectura.

Sistema: Conjunto de cosas que ordenadamente relacionadas entre sí contribuyen a determinado objeto.

Servicio: Organización y personal destinados a cuidar intereses o satisfacer necesidades del público o de alguna entidad oficial o privada.

Diseminación: Esparcir, divulgar, publicar, extender una noticia.

Perfil: Conjunto de rasgos peculiares que caracterizan a una persona o cosa. En informática se le denota así al lugar personal que tiene un usuario dentro de un sistema.

Aplicación: es el programa que el usuario activa para trabajar en el ordenador. Existen muchos programas de ordenador que pueden clasificarse como aplicación, generalmente se les conoce como Software.

Información: Conjunto de datos interrelacionados, debidamente estructurados y asociados a una experiencia.

Proceso: es un conjunto de actividades o eventos que se realizan o suceden con un determinado fin.

Metodología: es el conjunto de métodos por los cuales se regirá una investigación.

Internet: red mundial de ordenadores interconectados basada en el protocolo TCP/IP. Ofrece distintos servicios: e-mail, foros, FTP, chat, compartir ficheros, videoconferencia, etc.

Dominio: un dominio es un área funcional diferenciable con requisitos y rasgos similares que puede ser soportada por algún tipo de sistema de software.

Usuario: es la persona que utiliza o trabaja con algún objeto o que es destinataria de algún servicio público, privado, empresarial o profesional.

Herramienta: las herramientas se diseñan para cumplir uno o más propósitos específicos, por lo que son artefactos o requisitos con una función técnica.

Ingeniería: Se denomina con el nombre de ingeniería a aquella disciplina que se ocupa del estudio y de la aplicación de los conocimientos que de este y de la experiencia resultan, para que a través de diseños, técnicas y problemas puedan ser resueltos los diferentes problemas que afectan a la humanidad.

Versión: En el desarrollo de software se usa para identificar un cambio de un estado a otro superior, determina el aumento en la terminación de una aplicación, mejoras, modificaciones, añadidas, correcciones, denota lo alcanzado hasta un punto determinado.

Anexos

Anexo 1. Formulario para el establecimiento de un perfil de usuario para DSI.

[REGRESAR](#)

Perfil del Usuario	
PERFIL DE USUARIOS PARA DSI	
	
<p>La Diseminación Selectiva de Información (DSI) es un servicio que brinda la Biblioteca EPM a los funcionarios de Empresas Públicas de Medellín y a las Pymes, con el fin de que estén informados y actualizados en los temas de su interés. Si está interesado en beneficiarse con este servicio, diligencie el siguiente formulario:</p>	
Nombres y apellidos	
Correo electrónico	_____
Dependencia	_____
Cargo	_____
Dirección de la oficina y teléfono	
Si participa en algún proyecto corporativo especifique el nombre	
Defina los temas de su interés	1. _____
en una frase corta, teniendo en cuenta las especialidades de la Biblioteca: Ciencia, Industria, Medio Ambiente y Tecnología.	2. _____
	3. _____
Use otros términos en español e inglés para aclarar el tema	

Limite la búsqueda por	Idioma
	Español Inglés Francés Portugués Otro. ¿Cuál? _____ _____
	Tipo de material
	Libro impreso Libro electrónico (E-book) Revista impresa Revista electrónica Norma _____ _____
La Biblioteca EPM te acerca al conocimiento... › Ciencia › Industria › Medio ambiente › Tecnología	

Figura 18. Ejemplo de formulario para servicio DSI.

Anexo 2. Plantilla del servicio DSI que existía en la UCI.[REGRESAR](#)

DSI *Servicio de Diseminación Selectiva de Información*

Programación y Bases de Datos Dirección de Información
Universidad de las Ciencias Informáticas

No. 6 Viernes, 5 de junio de 2006

Noticias

- [Oracle Database 10g Express Edition Gana el Premio LinuxWorld Product Excellence por la Mejor Solución de Base de Datos.](#)
- [Microsoft Visual Studio Express será gratuito en forma permanente.](#)
- [Google confirma un nuevo 'Summer of Code'.](#)
- [Mayores inversiones en marketing online y bases de datos.](#)
- [Restricciones para los administradores de bases de datos.](#)
- [CA ofrece una herramienta gratuita de gestión de bases de datos multiplataforma.](#)

Artículos científicos

[Language-independent standards.](#)

Reseñas de libros

[Cross-Platform .NET development: using Mono, Portable .NET, and Microsoft .NET.](#)

Nuevas adquisiciones

[Mono a developer's notebook](#)

Eventos

[Encuentro de estudiantes e ingenieros industriales y afines \(ESTIIN 2006\)](#)
[I Congreso Iberoamericano de Ciencia, Tecnología, Sociedad e Innovación CTS+I](#)

Figura 19. Muestra de la plantilla del servicio DSI que existía en la UCI.

Anexo 3. Descripciones detalladas de casos de uso.

[REGRESAR](#)

CU_2 Gestionar temas.

Objetivo	El usuario busca, agrega y elimina temas de su lista de temas preferidos, el sistema muestra en todo momento los temas que estén definidos como preferidos con el fin de gestionarlos para recibir notificaciones de nuevos materiales.	
Actores	Usuario: (Inicia).	
Resumen	El caso de uso se inicia cuando el usuario busca, agrega o elimina temas de su lista de temas preferidos, finaliza cuando el usuario termina la gestión de temas.	
Complejidad	Media	
Prioridad	Crítico	
Precondiciones	El usuario ha sido autenticado en el sistema.	
Postcondiciones	El usuario debe haber entrado a gestionar sus preferencias de diseminación selectiva de información.	
Flujo de eventos		
Flujo básico Gestionar temas.		
	Actor	Sistema
1.	Busca, agrega, elimina o visualiza temas de su lista de temas preferidos.	1.2 Permite la ejecución de las siguientes opciones: <ul style="list-style-type: none"> a) Buscar temas. b) Agregar a preferidos. c) Mostrar preferidos. d) Eliminar temas preferidos.
Sección 1: “Buscar temas”		
Flujo básico Buscar tema.		
	Actor	Sistema
1.	Introduce un criterio para buscar por tema en un campo de texto cualquiera de los diez disponibles.	1.1 Muestra los resultados de la búsqueda que correspondan al criterio entrado por el actor, auto-completando de la lista de temas mientras

		se introducen los caracteres hasta que coincida con el criterio del usuario.
Flujos alternos		
1. Fallo de conexión.		
	Actor	Sistema
1.		1.1 Si ocurre un fallo de conexión el sistema mostrara el mensaje "Error de conexión."
Sección 2: "Agregar a preferidos"		
Flujo básico Agregar temas a la lista de temas preferidos.		
	Actor	Sistema
1.	Marca los cuadros de chequeo correspondientes a los temas y presiona el botón Guardar para establecerlos como preferidos.	1.1 Agrega los elementos que se encuentren en cada uno de los campos de texto que estén marcados por su cuadro de chequeo correspondiente a la lista de temas preferidos. 1.2 Cambia la etiqueta del botón Guardar a Guardando... mientras se realiza la acción.
Flujos alternos		
1. Fallo de conexión.		
	Actor	Sistema
1.		1.1 Si ocurre un fallo de conexión el sistema mostrara el mensaje "Error de conexión."
Sección 3: "Mostrar preferidos"		
Flujo básico Mostrar temas preferidos.		
	Actor	Sistema
1.		1.1 Muestra los temas preferidos en cada uno de los campos de texto en los cuales fueron establecidos con su correspondiente marca en el cuadro de chequeo, sucede en todo momento, después que son guardados.
Sección 4: "Eliminar temas preferidos"		
Flujo básico Eliminar temas de la lista de temas preferidos.		

	Actor	Sistema
1.	Desmarca los cuadros de chequeo de los temas que desea eliminar y presiona el botón Guardar.	1.1 Elimina los elementos que corresponden a los que fueron desmarcados de los cuadros de chequeo. 1.2 Cambia la etiqueta del botón Guardar a Guardando... mientras se realiza la acción.
Flujos alternos		
1. Fallo de conexión.		
	Actor	Sistema
1.		1.1 Si ocurre un fallo de conexión el sistema mostrara el mensaje "Error de conexión."

Tabla 10. Descripción detallada del CU Gestionar temas.

CU_3 Seleccionar tipo de material.

Objetivo	El usuario define los tipos de materiales para los cuales desea que se filtren los ejemplares que coincidan con los autores o temas preferidos. Afecta a autores y a temas, pero estos son a su vez independientes.	
Actores	Usuario: (Inicia) Selecciona los tipos de materiales.	
Resumen	El caso de uso se inicia cuando el usuario decide seleccionar los tipos de materiales que prefiere en su perfil el servicio de DSI y finaliza cuando termina la actualización por parte del usuario.	
Complejidad	Media	
Prioridad	Crítico	
Precondiciones	El usuario ha sido autenticado en el sistema.	
Postcondiciones	El usuario debe haber entrado a gestionar sus preferencias de diseminación selectiva de información.	
Flujo de eventos		
Flujo básico Seleccionar estado.		
	Actor	Sistema
1.		1.1 Inicialmente los tipos de materiales no están marcados, significa que se prefieren todos.

2.	Selecciona los tipos de material deseados y presiona el botón Guardar.	2.1 Guarda los elementos seleccionados. 2.2 Deja marcado el elemento que se seleccionó hasta que sea desmarcado y guardada nuevamente la sección. 2.3 Cambia la etiqueta del botón Guardar a Guardando... mientras se realiza la acción.
Flujos alternos		
1. Fallo de conexión.		
1.		1.1 Si ocurre un fallo de conexión el sistema mostrara el mensaje "Error de conexión."

Tabla 11. Descripción detallada del CU Seleccionar tipo de material.

CU_4 Seleccionar estado.

Objetivo	El usuario define si el estado del servicio DSI está activado o desactivado en dependencia de las necesidades que posea en el momento.	
Actores	Usuario: (Inicia) Selecciona el estado del servicio.	
Resumen	El caso de uso se inicia cuando el usuario decide activar o desactivar en su perfil el servicio de DSI y finaliza cuando termina la actualización por parte del usuario.	
Complejidad	Media	
Prioridad	Crítico	
Precondiciones	El usuario ha sido autenticado en el sistema.	
Postcondiciones	El usuario debe haber entrado a gestionar sus preferencias de diseminación selectiva de información.	
Flujo de eventos		
Flujo básico Seleccionar estado.		
	Actor	Sistema
1.		1.1 Inicialmente el estado del servicio está marcado en Desactivado.
2.	Establece el estado del servicio en la modalidad deseada en activado o desactivado y selecciona Actualizar.	2.1 Guarda la preferencia establecida. 2.2 Marca el elemento correspondiente al estado guardado.

Tabla 12. Descripción detallada del CU Seleccionar estado.

CU_5 Seleccionar periodicidad.

Objetivo	El usuario define el período de tiempo para el cual quiere que el servicio DSI le envíe las notificaciones en dependencia de sus necesidades.	
Actores	Usuario: (Inicia) Selecciona la periodicidad.	
Resumen	El caso de uso se inicia cuando el usuario decide establecer el período de tiempo para que el servicio de DSI le envíe las notificaciones y finaliza cuando termina la actualización por parte del usuario.	
Complejidad	Media	
Prioridad	Crítico	
Precondiciones	El usuario ha sido autenticado en el sistema.	
Postcondiciones	El usuario debe haber entrado a gestionar sus preferencias de diseminación selectiva de información.	
Flujo de eventos		
Flujo básico Seleccionar estado.		
	Actor	Sistema
1.		1.1 Inicialmente el período del servicio está marcado en Mensual.
2.	Establece el período del servicio en la modalidad deseada, diario semanal o mensual y selecciona Actualizar.	2.1 Guarda la preferencia establecida. 2.2 Marca el elemento correspondiente al estado guardado.

Tabla 13. Descripción detallada del CU Seleccionar periodicidad.

CU_6 Elaborar notificación.

Objetivo	Se encarga de realizar el análisis de los ejemplares nuevos en el sistema y las preferencias de los usuarios para elaborar las notificaciones que relacionen las correspondencias encontradas entre materiales y criterios.
Actores	Reloj: (Inicia) Elabora las notificaciones.
Resumen	El caso de uso se inicia cuando el Reloj, agente de ejecución automática corre el script diario para la elaboración de notificaciones como parte del servicio de DSI

	y finaliza cuando termina la ejecución del script.	
Complejidad	Alta	
Prioridad	Crítico	
Precondiciones	El reloj ha sido establecido.	
Postcondiciones	El reloj debe haber ejecutado los scripts.	
Flujo de eventos		
Flujo básico Seleccionar estado.		
	Actor	Sistema
1.	Ejecuta el script diario de elaboración de notificaciones.	<p>1.1 Selecciona los registros nuevos añadidos al sistema con fecha igual a la vigente en el momento de la ejecución.</p> <p>1.2 Realiza una consulta para encontrar coincidencias con los registros seleccionados y los criterios de los perfiles de usuarios.</p> <p>1.3 Elabora la notificación teniendo en cuenta el período, de ella almacena el usuario, correo, título y el link del registro en el catálogo.</p>

Tabla 14. Descripción detallada del CU Elaborar notificación.

CU_6 Enviar notificación.

Objetivo	Se encarga de realizar el envío de las notificaciones a los usuarios, las cuales ya elaboradas están disponibles en la base de datos, elimina las notificaciones que se envíen con éxito y tiene en cuenta siempre el período establecido por el usuario.
Actores	Reloj: (Inicia) Envía las notificaciones.
Resumen	El caso de uso se inicia cuando el Reloj, agente de ejecución automática corre los scripts correspondientes al período ya sea diario semanal o mensual para el envío de notificaciones como parte del servicio de DSI y finaliza cuando termina la ejecución de los scripts
Complejidad	Alta

Prioridad	Crítico	
Precondiciones	El reloj ha sido establecido.	
Postcondiciones	El reloj debe haber ejecutado los scripts.	
Flujo de eventos		
Flujo básico Seleccionar estado.		
	Actor	Sistema
1.	Ejecuta el script correspondiente al período que le corresponde.	1.1 Selecciona todos los correos notificativos a enviar según el período vigente. 1.2 Envía las notificaciones. 1.3 Elimina las notificaciones enviadas con éxito.

Tabla 15. Descripción detallada del CU Enviar notificación.

Anexo 4. Guía para la configuración de una tarea programada utilizando el crontab de los sistema GNU/Linux.

[REGRESAR](#)

La sintaxis del archivo crontab es muy clara y rígida. Hay seis campos por cada línea y cada campo está separado por un espacio. Los primeros cinco campos especifican el momento preciso de ejecución, mientras que el sexto campo es el propio comando o script que se ejecutará. Los campos son:

```
m      h      dom     mon     dow     /dir/de/un/fichero o comando específico.
|      |      |      |      |
|      |      |      |      +----- día de la semana (0 - 6) (domingo = 0)
|      |      |      +----- mes (1 - 12)
|      |      +----- día del mes (1 - 31)
|      +----- hora (0 - 23)
+----- minuto (0 - 59)
```

Para editar el archivo crontab sólo es necesario abrir una consola de comandos y ejecutar el comando:

```
usuario@pc:~$ crontab -e
```

Si es la primera vez que lo ejecutamos, veremos un archivo vacío en el editor Nano (editor de textos en consola de Gnome en Ubuntu). En este caso, si agregamos la siguiente línea al archivo:

```
00 23 * * * sh /usr/share/koha/cronjobs/dsi-emailcompose-daily.sh
```

Estaremos indicándole al **Cron** que el fichero dsi-emailcompose-daily.sh será ejecutado todos los días a las 11:00 PM. Se usa el carácter (*) para especificar en cualquier campo la opción “todos”, es decir, en la posición dom estará estableciendo todos los días del mes por ejemplo. Al guardar y salir del editor Nano deberá aparecer en consola el mensaje:

```
crontab:installing new crontab
```

Que indicará la correcta adición de una nueva tarea al **Cron**.

Anexo 5. Imagen de la interfaz de usuario terminada. Módulo DSI.

UCI Dirección de Información

Inicio Quiénes Somos Servicios **Catálogo** Repositorio institucional Base de Datos Tutoriales

Tipo de Búsqueda

- BÚSQUEDA SIMPLE
- BÚSQUEDA AVANZADA
- EXPLORAR POR TEMA

Perfil

- MI RESUMEN
- SERVICIO DSI**
- MIS SANCIONES
- MIS DETALLES PERSONALES
- MIS ETIQUETAS
- MI HISTORIAL DE LECTURAS
- MIS SUGERENCIAS DE ADQUISICIÓN
- MIS LISTAS
- SUBIR RECURSOS DIGITALES

El servicio de Diseminación Selectiva de Información le permite recibir notificaciones a su correo electrónico acerca de nuevos ejemplares adquiridos que correspondan con sus necesidades informativas.

Para hacer uso del mismo, solo llene el perfil a continuación:

Busque y guarde sus autores preferidos:

- Torres de Diego, Mario
- Cintra Kindelán, Ornides
- Nocedo de León, Irma
- Chen Xiang, Yang
-
-
-
-
-
-

GUARDAR

Busque y guarde sus temas preferidos:

- HISTORIA
- CIENCIA
- CIENCIAS SOCIALES
- PROGRAMACION

Monografía seriada

Tesis de Maestría

Trabajo de Diploma

M. Acompañante

Libros

Tesis de Doctorado

Publicaciones seriadas

Obras de referencia

Artículos de revista

GUARDAR

Configuración general del servicio:

Estado del servicio:

Activado

Desactivado

Período para el recibo de notificaciones:

Diario

Semanal

Mensual

GUARDAR

Dirección de Información: informacion@uci.cu | Teléfono: 835 8440
 Copyright © 2011. [Biblioteca de la Universidad de las Ciencias Informáticas](#)
 Optimizado para [Mozilla Firefox](#) y una resolución de 1024 x 768 o superior

Figura 20. Interfaz de usuario del módulo DSI.

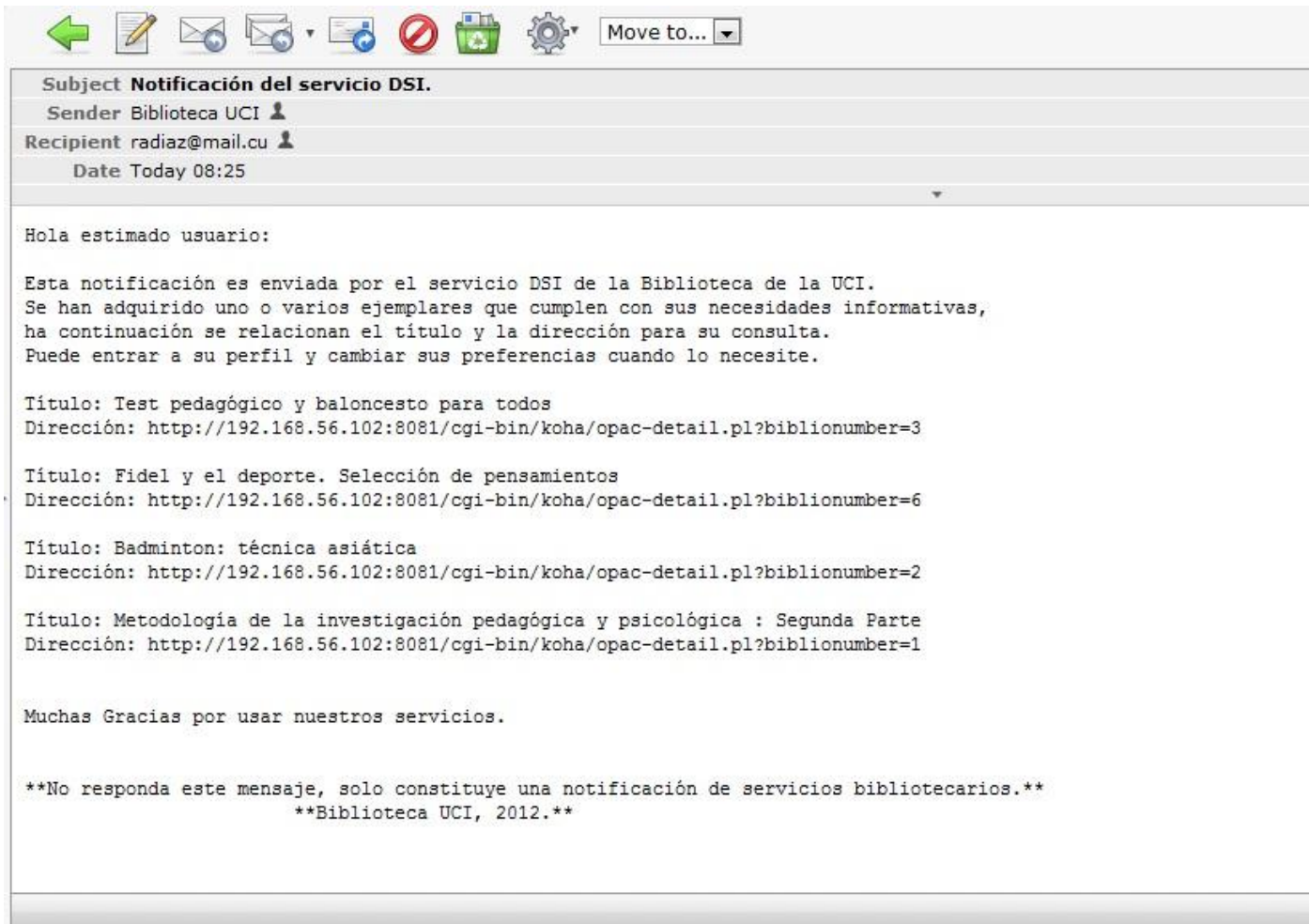
Anexo 6. Imagen de la notificación recibida por el usuario.[REGRESAR](#)

Figura 21. Notificación recibida por el usuario.

Anexo 7. Ejemplo del Plan de Pruebas de Integración.[REGRESAR](#)

1. Identificación.	Plan prueba Int001.
2. Elementos a probar.	Buscar, agregar, eliminar y mostrar autores preferidos.
3. Enfoque.	Ascendente.
4. Criterio de aceptación.	La salida coincide con la esperada.
5. Criterio de suspensión.	Hasta terminar todos los casos de pruebas.
6. Productos a entregar.	Plan de pruebas con la lista de casos ejecutados y sus resultados, lista de problemas identificados.
7. Tareas.	Preparar casos de prueba, ejecutar casos de pruebas, evaluar resultados y preparar informe de problemas.
8. Necesidades ambientales.	Computadora de escritorio con acceso físico al sistema y las herramientas utilizadas para la implementación.
9. Responsabilidades.	Probador: Preparar casos de pruebas, ejecutar casos de pruebas, evaluar resultados y preparar informe de problemas. Programador: Preparar componentes auxiliares si se requieren.
10. Personal.	Se requiere probador con dominio de las pruebas determinadas para efectuarse. El programador será el mismo desarrollador.
11. Calendario.	Preparar caso de prueba: 2 días. Ejecutar casos de prueba: 1 día. Evaluar resultados y reportar 1 día.
12. Riesgos y contingencias.	R1: Probador enfermo o no disponible. C1: Asumir el rol por otro miembro del proyecto que posea conocimientos sobre pruebas. R2: Problemas imprevistos de hardware C2: Negociar más tiempo. R3: Programado no entrega componentes auxiliares necesarios. C3: Probador los realiza, se requiere al programador.

Tabla 16. Plan de Pruebas de Integración.