

Universidad de las Ciencias Informáticas

Facultad 1



Sistema de recomendación para el catálogo en línea de la biblioteca de la Universidad de las Ciencias Informáticas

Trabajo de diploma para optar por el título de
Ingeniero en Ciencias Informáticas

Autor:

Yanisel Zamora Cesé

Tutores:

Ing. Yenisel Valdés Hernández

Ing. Lizandra Candelario Rodríguez

La Habana, 22 junio 2012

“Año 54 de la Revolución”



El único autógrafo digno de un hombre es el que deja escrito con sus obras.

DECLARACIÓN DE AUTORÍA

Declaración de autoría

Declaro ser el autor de la presente tesis y reconozco a la UCI los derechos patrimoniales de la misma, con carácter exclusivo.

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

Yanisel Zamora Cesé

Ing. Yenisel Valdés Hernández

Ing. Lizandra Candelario Rodríguez

Dedicatoria

A mi padre, que aunque no está conmigo físicamente se que está orgulloso de su hijita, te quiero mucho papito, siempre estarás en mi corazón.

A mi mamita, por ser la madre más buena del mundo, por ser fuerte, luchadora, eres mi ejemplo a seguir, te adoro.

A mis hermanos por su amor incondicional.

A mi novio Wilber por haberme ayudado con el desarrollo de mi tesis.

Agradecimientos

A mi madre por haberme traído al mundo, por brindarme todo su amistad, amor y cariño si esperar nada a cambio, por educarme y forjar buenos valores para poder ser hoy una persona de bien, por consentirme, mimarme, por estar a mi lado en todos momentos, por su sacrificio, dedicación, ternura, amor, por los sabios consejos que siempre me da, por darme las fuerzas para seguir siempre hacia delante. Te agradezco con la vida todo lo que has hecho por mí, lo que estás haciendo y lo que te falta por hacer, porque para una madre, los hijos seguirán siendo bebés. Eres mi ejemplo a seguir. Te amo con toda mi alma.

A mis hermanos Yoani y Yaniuska, por los momentos de alegría que hemos pasado juntos, por las peleas de hermanos que hemos tenido, por brindarme su amor, cariño, consejos, por cuidarnos el uno con el otro. Los quiero mucho.

A mi padrastro Hornedo, por hacerme una mujer de bien, por cuidarme y quererme como una hija, por su amor y cariño. Te quiero.

A mi novio Wilber, por quererme tanto, por cuidarme, mimarme, protegerme, por su amor tan grande, por ayudarme en todo el tiempo que hemos estado juntos, por complacerme en todo en lo que ha podido, por sus buenos consejos, por hacer de mi toda una mujer, por hacerme ver la vida desde otro punto de vista, por compartir conmigo momentos de alegría y tristeza, por todo lo que has hecho por mí. Te quiero con la vida.

A mi familia en general, a mis tíos, primos, a mi sobrinita Dayamanty, a Miliena, Dailin, Aida, Armelio, a mi perrito Rinty y en especial a mi tía Marianela.

A mis compañeros de aula, a Lianna, Yusmila, Elizabeth, Yaniris, Lissette, Tailén, Rayko y en especial a Jose y a Hector por haberme ayudado con el desarrollo de mi tesis. Siempre los recordaré a todos.

Resumen

Actualmente en la Universidad de las Ciencias Informáticas (UCI) se desarrolla un Sistema Integrado de Gestión Bibliotecaria que automatiza los procesos y servicios llevados a cabo en la biblioteca de la universidad. Este sistema posee un catálogo en línea que permite la búsqueda de temas según el criterio de los usuarios, sin embargo carece de un servicio que permita agilizar los procesos de búsqueda. Para darle solución a este problema se concibe un sistema de recomendación que permite la sugerencia de materiales bibliográficos a los usuarios de la biblioteca y facilitar los procesos de búsqueda. Para ello se realiza un estudio de los sistemas de recomendaciones, se selecciona un algoritmo de filtrado colaborativo para generar las recomendaciones, así como las herramientas, tecnologías y metodología para su desarrollo y se avalúa su correcto funcionamiento mediante pruebas de caja negra.

Palabras clave: catálogo en línea, filtrado colaborativo, recomendaciones, sistema de recomendación.

Tabla de contenido

Introducción	1
Capítulo I: Fundamentación teórica	5
1.1 Antecedentes de los sistemas de recomendación	5
1.2 Tipos de sistemas de recomendaciones	6
1.2.1 Recomendaciones basadas en contenido.....	6
1.2.2 Recomendaciones basadas en el filtro colaborativo.....	7
1.2.3 Recomendaciones híbridas.....	10
1.3 Realimentación de los sistemas de recomendación.....	10
1.3.1 Realimentación implícita	11
1.3.2 Realimentación explícita	11
1.4 Aplicaciones de los sistemas de recomendación	11
1.4.1 E-commerce y sistemas de recomendaciones	11
1.4.2 E-learning y sistemas de recomendación.....	12
1.5 Ejemplos de sistemas de recomendaciones	13
1.6 Metodología de desarrollo adoptada	15
1.7 Herramientas y tecnologías a utilizar	15
1.7.1 Eclipse 3.7.0	15
1.7.2 EPIC	15
1.7.3 Visual Paradigm 8.0.....	16
1.7.4 Mysql 5.1.62	16
1.7.5 Apache 2.2.20.....	16
1.8 Lenguajes a utilizar	16
1.8.1 Perl 5.12	16
1.8.2 HTML 3.0.....	17
1.8.3 CSS 2.1	17
1.8.4 Java Script.....	17
1.8.5 UML Lenguaje Unificado de Modelado	17
Capítulo II: Estudio algorítmico	19
2.1 Sistema de recomendación para el catálogo en línea de la biblioteca de la UCI19	
2.2 Algoritmos en sistemas de recomendación	20
2.2.1 Algoritmos basados en memoria.....	20
2.2.2 Métodos basados en modelo	22
2.3 Limitaciones en los algoritmos en sistemas de recomendación.....	24

2.3.1 Arranque frío	24
2.3.2 Dispersión de los datos	25
2.3.3 La sobre-especialización.....	25
2.4 algoritmo propuesto para el sistema de recomendación.....	26
2.4.1 Definición de la correlación de Pearson	26
Delimitación del vecindario	28
2.4.3 Definición de las recomendaciones.....	28
Definición de los pesos de las recomendaciones.....	28
2.4.5 Tratamiento del arranque frío.....	29
Capítulo III: Características del sistema de recomendación.....	31
3.1 Propuesta de solución	31
Modelo del dominio.....	31
3.2 Especificación de requisitos	32
Requisitos funcionales.....	33
Requisitos no funcionales	33
3.3 Definición de los actores y casos de usos	34
Descripción del caso de uso recomendar material a un usuario	35
Capítulo IV: Análisis y diseño.....	37
4.1 Análisis del sistema.....	37
4.1.1 Clases del análisis	38
4.1.2 Diagramas de interacción	39
4.2 Diseño del sistema.....	41
4.2.1 Diseño de la base de datos.....	46
4.3 Arquitectura del sistema.....	46
4.4 Tratamiento de errores.....	48
Capítulo V: Implementación y prueba	49
5.1 Diagrama de despliegue.....	49
5.2 Diagrama de componentes.....	50
5.3 Modelos de prueba.....	52
Conclusiones generales.....	56
Recomendaciones	57
Referencias bibliográficas	58
Anexo 1	62
Anexo 2	68

Introducción

El vertiginoso desarrollo de la *World Wide Web* (www) ha transformado a la red en un inmenso espacio de información con contenidos diversos, a menudo muy poco estructurados u organizados. Actualmente internet es la principal fuente de información para la mayoría de las personas, sin embargo, la búsqueda de información relevante es muy complicada, así como su clasificación y corrección, pues toma mucho tiempo y espacio. Estas condiciones conllevan a que sea cada vez más difícil para los usuarios localizar información de interés para ellos, debido a la sobrecarga de información existente en la red.

A menudo, es necesario tomar decisiones sin tener la experiencia necesaria o suficiente sobre las alternativas a elegir. Por ello, en el día a día las personas confían y tienen en cuenta las recomendaciones de terceros para poder llevar a cabo decisiones de mayor provecho o interés. Esta falta de experiencia o conocimiento sobre las alternativas, así como el gran incremento de la información que genera la misma sociedad, hace que cada día vaya cobrando más importancia el hecho de tener métodos automáticos o semiautomáticos de filtrado y/o selección de información para la toma de decisiones.

Como uno de los métodos de filtrado de información surgidos para ayudar a los usuarios a resolver el problema de la sobrecarga de información están los sistemas de recomendación o sistemas recomendadores; los cuales son muy utilizados en campos como el comercio electrónico y el ocio. Sin embargo, aún quedan campos poco explorados en este sentido como son las bibliotecas digitales, las cuales poseen gran cantidad de material bibliográfico, que conlleva a que los usuarios realicen una ardua tarea a la hora de realizar alguna búsqueda. Muchas son las ocasiones en que se desconoce sobre la existencia de cierto libro en particular, o de aquellos libros que en un momento dado pudieran ser del interés de algún usuario. Por consiguiente, se omite información valiosa que enriquecería el contenido de múltiples investigaciones, o simplemente los conocimientos.

Cuba cuenta con bibliotecas que prestan sus servicios a la comunidad, como son las bibliotecas escolares, las públicas y las universitarias, siendo estas de gran importancia ya que tienen la misión de facilitar el acceso y la difusión de los recursos de información y colaborar en los procesos de creación del conocimiento, a fin de contribuir a la consecución de los objetivos de la institución.

La Universidad de las Ciencias Informáticas (UCI), universidad de nuevo tipo que vincula la docencia-producción-investigación como nuevo modelo de formación profesional, cuenta con un Sistema Integrado de Gestión Bibliotecaria (SIGB) que automatiza los procesos y servicios de la biblioteca, los cuales facilitan el logro de los objetivos de la institución, visualizando mediante un catálogo los resultados de la actividad investigativa en la UCI, los cuales son cada vez más amplios.

Este sistema, al no contar con un método (servicio) que filtre información de relevancia para el usuario, hace que sea muy difícil encontrar con facilidad dicha información, pues no siempre los materiales más relevantes están ubicados en primera posición en los resultados de búsqueda, haciendo difícil la localización de los mismos de acuerdo con el interés de cada usuario, ocupando tiempo y espacio a la hora de adquirir alguna información, es por esto que en ocasiones, los usuarios abandonan la búsqueda sin tener resultado alguno.

De la situación antes expuesta se origina el siguiente **problema**: ¿Cómo facilitar a los usuarios de la biblioteca de la UCI la localización de información relevante para ellos en el catálogo en línea?

Se definen como **objeto de estudio**: las técnicas de filtrado de información, siendo el **campo de acción**: los sistemas de recomendación.

El **objetivo general de la investigación** es: desarrollar un sistema de recomendación para facilitar al usuario la percepción de la información en el catálogo en línea de la biblioteca de la UCI, mediante el análisis e implementación de algoritmos de filtrado y recuperación de la información.

Del objetivo general se derivan los siguientes **objetivos específicos**:

- Fundamentar los conceptos, características y antecedentes de los sistemas de recomendación mediante un estudio bibliográfico.
- Definir el sistema de recomendación que se pretende agregar al catálogo en línea de la Biblioteca UCI.
- Diseñar el sistema de recomendación.
- Implementar el sistema de recomendación para el catálogo en línea.
- Validar mediante pruebas de caja negra la solución propuesta.

El trabajo de diploma queda sustentado en la siguiente **idea a defender**: El desarrollo de un sistema de recomendación para el catálogo en línea de la biblioteca

de la UCI permitiría a los usuarios el acceso a objetos conocidos o desconocidos de su interés, teniendo en cuenta las preferencias y el comportamiento del mismo en el sistema.

Métodos teóricos:

Para la realización de esta investigación se utilizaron métodos que permitieron reproducir teóricamente el objeto, en el pensamiento, en toda su objetividad y concreción, así como comprenderlo en su desarrollo, historia y lógica. Los métodos empleados fueron:

Histórico lógico: posibilitó hacer un estudio de los fundamentos teóricos de los sistemas de recomendación desde sus inicios hasta la actualidad en el ámbito internacional, así como las tendencias actuales de estos sistemas.

Analítico-Sintético: permitió extraer y analizar los conceptos y definiciones más importantes relacionadas con el objeto de estudio, a partir de los cuales se obtuvo una propuesta que diera cumplimiento a la situación problemática.

Modelación: se utilizó para modelar las diferentes etapas del proceso de ingeniería de software por las que transitó el sistema de recomendación.

Estructura de la tesis:

El presente trabajo de diploma consta de una introducción, cinco capítulos, conclusiones, recomendaciones, referencias bibliográficas, anexos y glosario de términos.

Capítulo I. Fundamentación Teórica: el objetivo de este capítulo es realizar un estudio de los antecedentes de los sistemas de recomendación. Abordar conceptos y características que permitan comprender temas relacionados con los sistemas de recomendación. Incluye un análisis y estudio de los sistemas existentes, así como la selección de las herramientas, metodología, lenguajes y tecnologías a utilizar en el desarrollo de la solución.

Capítulo II. Estudio algorítmico: en este capítulo se hace un estudio de los algoritmos de filtrado colaborativo para generar las recomendaciones, se describen las características de cada uno de ellos, así como su funcionamiento, concluyendo con la selección del algoritmo a implementar para el sistema de recomendación a desarrollar.

Capítulo III. Características del sistema de recomendación: en este capítulo se representa el modelo del dominio y de casos de uso. Se hace una descripción general

de la propuesta de solución, donde serán expuestos los requisitos funcionales y no funcionales que debe presentar el mismo con sus respectivas descripciones.

Capítulo IV. Análisis y diseño: se especifican los detalles de la construcción de la solución, se representan los diagramas de clases del análisis y clases del diseño con sus respectivas descripciones, así como la representación del diseño de la base de datos y la arquitectura a utilizar.

Capítulo V. Implementación y pruebas: en este capítulo se detalla el modelo de despliegue y el modelo de componentes que conforman el sistema de recomendación. Además, se describen las pruebas de unidad realizadas que permiten validar los requerimientos funcionales de la aplicación.

Capítulo I: Fundamentación teórica

1.1 Antecedentes de los sistemas de recomendación

Diariamente las personas se ven expuestas a una gran cantidad de información, lo cual reduce sus capacidades para procesarla. Es, por tanto, el momento de desarrollar tecnologías que alivien esta sobrecarga de información. Para darle solución a este problema surgen los sistemas de recomendación, los cuales son técnicas que tratan de presentar a los usuarios *ítems* de información como películas, noticias, libros, música, entre otros, sobre los cuales el usuario está interesado, con el propósito de facilitar la toma de decisiones en temas/dominios en los que las posibilidades son muchas y variadas.

El origen de este tipo de sistema se remonta a principios de la década de los 90, cuando comienzan a surgir dentro de los servicios de grupos de noticias, servicios de filtrado de noticias que permitían a su comunidad de usuarios acceder exclusivamente a aquellas noticias que potencialmente podían ser de su interés. No obstante, el primer sistema que surgió en este ámbito fue el llamado *Tapestry*, sistema que permite almacenar el *feedback* de los usuarios sobre artículos o noticias interesantes para ellos, donde estos artículos o noticias pueden ser utilizados por otros usuarios para determinar si la información del documento es relevante o no (1).

Otra de las primeras formas de filtrado de información electrónica apareció con el trabajo de *Housman y Kaskela*, en el que se diseñó un método que de forma automática se pudiera mantener a los científicos informados sobre nuevos documentos publicados en sus áreas de trabajo o especialización. Este método se basaba en la creación de un perfil de usuario que contenía ciertas palabras clave relevantes para el usuario, que son utilizadas para buscar coincidencias entre estas y los nuevos documentos o artículos, con el fin de intentar predecir cuáles de las informaciones encontradas serían del interés para los científicos (1).

Otra aproximación a los sistemas de recomendación se dio con *The information Leasn System* el cual, basado en el contexto del correo electrónico, les permitía a los usuarios crear reglas para filtrar los correos electrónicos. Así pues, los usuarios podían, por ejemplo, crear una regla para etiquetar todos los mensajes provenientes de ciertas personas o correo electrónico (1).

Ya en 1997, Resnick y Varian proponen llamar a estas técnicas con el nombre de

sistema de recomendación por dos razones: en primer lugar porque puede ocurrir que los usuarios no colaboren explícitamente entre ellos y en segundo lugar, porque el sistema puede sugerir elementos no conocidos hasta el momento por el usuario y en ese caso se estaría realizando una recomendación (2).

1.2 Tipos de sistemas de recomendaciones

Existen dos grandes tipos de sistemas de recomendaciones en base a la fuente de conocimiento usada para realizar la recomendación: basados en contenidos, que son aquellos que realizan la recomendación al usuario en base a la descripción de los productos (3) y los colaborativos, que son aquellos que utilizan valoraciones asociadas a los productos dadas por el usuario y/o por otros usuarios (4).

Otras técnicas han sido propuestas para desarrollar recomendaciones por ejemplo, las basadas en conocimientos (5) y recomendaciones basadas en perfiles demográficos pero son menos usadas (6). Para mejorar el rendimiento de las distintas técnicas surgieron las recomendaciones híbridas. Estas recomendaciones surgen como combinación de otros enfoques (por ejemplo, basados en contenidos y colaborativos).

1.2.1 Recomendaciones basadas en contenido

Las recomendaciones basadas en contenido son aquellas que realizan la recomendación en base a la descripción de los productos a recomendar. Algunos autores sitúan dentro de las recomendaciones basadas en contenido a los basados en casos (7). Según estos autores, la diferencia entre la basada en contenido y la basada en casos radica básicamente en que en la basada en casos se dispone de una representación más estructurada de la descripción de los productos. Para otros autores (8), estas son simplemente dos denominaciones de una misma familia de sistemas de recomendación, que están inspiradas o tienen su origen en las técnicas de razonamiento basadas en casos.

El sistema de recomendación *Adaptive place Advisor*, recomendador de restaurantes en la zona de la bahía de San Francisco en Estados Unidos, fue uno de los primeros sistemas que utilizó el filtro de información basado en contenido. En este sistema los productos a recomendar son los diferentes restaurantes descritos por los atributos tales como el tipo de cocina, la valoración, el precio, la situación, si admite reservas, si tiene parqueo y las opciones de pago (9).

También se puede mencionar el sistema de recomendación *Entree*, un recomendador que sugiere restaurantes en la ciudad de Chicago. *Entree* pertenece a

la familia de los sistemas *FindMe* que fueron de los primeros recomendadores que consiguieron incluir retroalimentación por parte del usuario en forma de críticas a las recomendaciones propuestas (10).

Los pasos a seguir en las recomendaciones basadas en contenido son los siguientes:

1. Extraer las características de los objetos.
2. Comparar los objetos con el perfil del usuario para identificar las preferencias de este sobre dicho objeto.
3. Recomendar objetos similares en su contenido a los que forman parte de su perfil.

La ventaja de las recomendaciones basadas en contenido es que se pueden realizar recomendaciones sin la necesidad de tener un historial previo, por lo que permite realizar predicciones independientemente del historial del usuario.

El principal inconveniente en este tipo de sistema es la sobre-especialización, que consiste en que las recomendaciones resultantes suelen ser siempre muy similares, dado que se basa en la misma información y que dependiendo del dominio de los objetos a recomendar, es difícil extraer buena información de estos para realizar un cálculo falible (11).

1.2.2 Recomendaciones basadas en el filtro colaborativo

Las recomendaciones basadas en el filtro colaborativo es, probablemente, la técnica más utilizada en los sistemas de recomendación propuestos en la literatura (11). El filtrado colaborativo es el proceso por el que se evalúan y seleccionan los productos a partir de las opiniones de otros usuarios. A diferencia de los basados en contenido, el filtrado colaborativo a la hora de ofrecer una recomendación a un usuario no considera únicamente las preferencias personales, sino también las de otros usuarios con intereses similares a los suyos. Tal similitud se estima a partir de las valoraciones definidas en sus perfiles personales. De ahí que los enfoques colaborativos no requieran la descripción de los productos.

Una de las clasificaciones para las recomendaciones basadas en el filtrado colaborativo es la propuesta por Victor Codina Busquet (11), que distingue dos tipos:

- **Filtrado colaborativo basado en usuario (*user-based*):** sugiere a cada usuario aquellos productos que han interesado a sus vecinos. Se considera que dos usuarios son vecinos, si los mismos presentan preferencias similares,

han clasificados los mismos productos en sus perfiles y le han asignado a estos índices parecidos.

- **Filtrado colaborativo basado en producto (*ítem-based*):** un producto es recomendado a un usuario si el mismo es similar a los incluidos en su perfil. En este caso, se considera que dos productos son similares si los usuarios que han valorado uno de ellos tienden a valorar el otro asignándole índices de interés parecidos.

En ambas técnicas, el objetivo es predecir el nivel de interés del usuario en relación a un producto dado y, en función de esto, sugerirlo (o no) a dicho usuario. En este sentido, Almudena Ruiz Iniesta identifica dos fases para este proceso (12):

1. En los enfoques colaborativos basado en usuario, la primera fase selecciona a los usuarios cuyas preferencias son similares a los del usuario activo. Por el contrario, los enfoques basados en producto extraen las preferencias del usuario que son más similares al producto objetivo. Luego se forma el vecindario, ya sea de los usuarios o de los productos definidos en su perfil, que estará formado por los usuarios y productos más relevantes.
2. En la segunda fase, el sistema debe predecir el nivel de interés del usuario en relación del producto objetivo. Para ello, los enfoques basados en usuario consideran el nivel de interés de los vecinos del usuario activo en relación al producto objetivo. Por el contrario, la versión colaborativa basada en producto considera el nivel de interés del usuario en relación a aquellos productos de su perfil que son más similares al objetivo.

Uno de los sistemas de recomendación que utiliza el filtrado colaborativo es *MovieLens*, un recomendador de películas donde los usuarios valoran películas en una escala de 1 a 5 estrellas, siendo 1 “malísima” y 5 “debe verse”. Una vez que las películas hayan sido valoradas, el sistema utiliza las valoraciones de la comunidad para recomendar otras películas en las que el usuario podría estar interesado (13).

La librería *Amazon.com* dispone también de uno de los más conocidos y utilizados recomendadores colaborativos en el ámbito del comercio electrónico, que es una aproximación de filtrado colaborativo basada en producto (12). De esta forma, el sistema recomienda una lista de productos similares al producto valorado por el usuario.

El sistema de recomendación *Tapestry*, sistema que también utiliza el filtro colaborativo, fue el primero que incorporó las opiniones de los usuarios en una base de datos de mensajes y un buscador. Este sistema recomienda listas de noticias y boletines electrónicos de artículos a usuarios de correo electrónico. *Tapestry* almacenaba el contenido de los mensajes junto con metadatos de los lectores, autores y los usuarios que respondían a los mensajes. También permitía que cualquier usuario almacenara notaciones sobre los mensajes (7).

En la familia de los sistemas de recomendación de filtro colaborativo el método más utilizado a la hora de medir la similitud entre los perfiles de varios usuarios es el método de selección de vecinos más cercanos. Este método se utiliza para detectar similitud entre las preferencias de los usuarios en los recomendadores colaborativos (tanto en los basados en usuarios como los basados en productos). En los enfoques basados en usuario cada perfil se presenta mediante un vector, cuyas componentes son las clasificaciones que el usuario ha asignado a cada uno de los productos incluidos en el mismo (12). Una vez modelados los usuarios, se aplican sobre sus respectivos vectores métodos como la similitud basada en coseno o la correlación de *Pearson*, detectando así parecidos entre sus preferencias.

$$similitud_{cos}(A,B) = \frac{\vec{A} * \vec{B}}{|\vec{A}| * |\vec{B}|}$$

Ecuación 1

$$corr_{pea}(\vec{A}, \vec{B}) = \frac{\sum_r(\vec{A}|r| - \bar{A}) * (\vec{B}|r| - \bar{B})}{\sqrt{\sum_r(\vec{A}|r| - \bar{A})^2 * \sum_r(\vec{B}|r| - \bar{B})^2}}$$

Ecuación 2

En las ecuaciones **(1)** y **(2)** A y B son los valores medios de las clasificaciones por los dos usuarios cuyas preferencias sean comparadas y los niveles de interés se registran en los vectores A y B respectivamente. En cuanto a la ecuación **(2)**, mientras más productos hayan clasificados los usuarios y mientras más parecidas sean las clasificaciones de los mismos, mayor será la correlación detectadas entre sus preferencias (12).

Por otra parte, los enfoques colaborativos basados en producto calculan la similitud entre dos productos. En este caso, se seleccionan los usuarios que han

clasificado a la vez los productos comparados y construyen grandes vectores a partir de los niveles de interés que estos han asignado a ambos productos. Finalmente se aplican sobre los vectores los métodos anteriormente mencionados, para así obtener los valores de similitud concretos entre los productos considerados (14).

Al reflexionar sobre este tema, se puede llegar a la conclusión que, cuantos más usuarios hayan clasificado a la vez los productos comparados y cuántos más parecidos sean los niveles de interés asignados a los mismos, más significativo será el valor de similitud medido entre ambos. Sin embargo, una de las principales desventajas de esta técnica de recomendación es la baja densidad de información disponible, ya que muchos usuarios con gustos especiales son mal recomendados, un nuevo objeto no es recomendable hasta que no sea valorado por un número determinado de usuarios y finalmente son pocos escalables.

1.2.3 Recomendaciones híbridas

Los enfoques básicos de la recomendación anteriormente mencionados han sido utilizados de manera exitosa en disímiles campos y escenarios. Pese a esa difusión, de forma general cada uno de ellos presenta varias desventajas relacionadas principalmente con la necesidad de un gran número de usuarios adheridos al sistema para un correcto funcionamiento del mismo y con la tendencia de una sobre-especialización que va aumentando a medida que aumenta la utilización de la aplicación por parte del usuario.

Con vistas a mitigar estas desventajas, una opción muy utilizada es la de combinar la recomendación basada en contenido con el filtrado colaborativo, conformando un sistema de recomendación con características híbridas. A través de estos, la construcción de preferencias se basa tanto en los intereses de usuarios, como en los intereses de usuarios similares, coincidiendo así con los dos tipos específicos de sistemas de recomendación antes mencionados.

Varios autores han demostrado cómo a través de este enfoque la calidad de las recomendaciones aumentan considerablemente (15), constituyendo de esta forma una alternativa a tener en cuenta a la hora de diseñar un sistema de este tipo.

1.3 Realimentación de los sistemas de recomendación

Un sistema de recomendación debe evolucionar en el tiempo en cuanto a la calidad de sus recomendaciones y pronóstico en base a la experiencia y nueva información adquiridas. Para conseguir este objetivo se utilizan mecanismos de

realimentación entre los sistemas y los gustos del usuario, siendo los mismos la realimentación implícita y la explícita.

1.3.1 Realimentación implícita

Un mecanismo de realimentación implícita es aquel que deduce las preferencias a partir del comportamiento del usuario y su historial. Esto permite que en la mayoría de los casos no sea necesario pedirle al mismo una gran cantidad de información sobre sus preferencias para que pueda ser recomendado.

1.3.2 Realimentación explícita

Se considera que un sistema de recomendación utiliza el mecanismo de realimentación explícita, cuando el usuario expresa sus preferencias conscientemente. Como por ejemplo, en el caso de valorar un contenido, donde la valoración consiste en darle a dicho contenido un valor numérico dentro de un rango definido.

1.4 Aplicaciones de los sistemas de recomendación

Los sistemas de recomendación tienen aplicaciones en áreas de la sociedad. Dentro de las aplicaciones de estos sistemas, es de remarcar su amplia y estratégica utilización en el marco del *e-commerce* y del *e-learning*.

1.4.1 E-commerce y sistemas de recomendaciones

Con el desarrollo y competitividad del comercio de hoy en día, conocer cuáles son sus clientes, cuáles son sus preferencias y cómo evalúan los productos existentes es de suma importancia para las grandes compañías. Trabajar en base al criterio de los usuarios finales constituye una de las llaves del éxito de muchas empresas con presencia en Internet.

Muchos sitios web de comercio electrónico han comenzado a retroalimentarse de sus usuarios de manera directa, al permitirle compartir opiniones y evaluar de manera cuantitativa los productos que compran. Esta información recopilada es de un alto valor para la creación de nuevas aplicaciones que perfeccionen la proyección de la empresa o negocio en cuestión, aplicaciones dentro las cuales se encuentran los sistemas de recomendación.

Unos de los ejemplos más clásicos de la aplicación de los sistemas de recomendación en este campo lo constituye *Amazon.com*, sitio de venta de productos ya mencionado anteriormente, cuya sección de venta de libros contiene implícita varias funcionalidades basadas en esta filosofía. Entre estas cabe destacar *Customers who bought*, la que se encuadra en la página de información de cada libro dentro del

catálogo y que contiene dos listas de recomendaciones, una con los libros frecuentemente comprados por aquellos clientes que compran el libro y otra con los autores frecuentemente leídos por estos mismos clientes (16).

Como otras aplicaciones populares en Internet que implementan sistemas de recomendaciones pueden citarse: *CDNOW*, portal concebido para la venta de *CD* de música, *eBay*, considerado como uno de los mayores portales de compra-venta del mundo, *Levis*, creada con el objetivo de dar soporte a los clientes de la ropa de la marca homónima y *MovieFinder.com*, capaz de localizar filmes con características similares a los de un filme dado. Todos estos sitios tienen en común un alto nivel de aceptación por parte de los clientes de la red global (17).

1.4.2 E-learning y sistemas de recomendación

El *e-learning* es un sistema de educación electrónico o a distancia en el que se integra el uso de las tecnologías de la información y otros elementos pedagógicos-didácticos para la formación, capacitación y enseñanza de los usuarios o estudiantes en líneas

En esta área, al igual que en el *e-commerce*, la existencia del desbordamiento de información imposibilita el acceso a los recursos disponibles en plataformas de programas educativos y a otras aplicaciones de semejante corte. La utilización de sistemas recomendadores aplicados al *e-learning* ha sido una de las soluciones para enfrentar esta problemática, perfeccionando la eficiencia y eficacia del medio de la situación actual antes descrita (18). Es válido señalar que aunque la introducción de sistemas recomendadores en este campo tuvo una adopción tardía con respecto a la aplicación en el *e-commerce*, estos actualmente revisten la misma influencia e importancia tanto en un campo como en otro.

Pueden citarse varios trabajos que muestran resultados enfocados a implementaciones reales de sistemas de recomendaciones aplicados al campo de la enseñanza. En (19) se presenta un sistema recomendador colaborativo para recursos de *e-learning*, el cual permite que profesores de perfil similar compartan los resultados de sus investigaciones, tras aplicar minería de datos de manera local sobre sus propios cursos. Por otra parte (19) muestra la implementación de un sistema para la recomendación de artículos científicos y técnicos, capaz de retroalimentarse de la web para la obtención de nueva información basándose en la adaptabilidad del perfil de usuario.

1.5 Ejemplos de sistemas de recomendaciones

Tapestry

Tapestry fue el primer sistema de recomendación basado en el filtro colaborativo, creado en 1992, el cual permitía que los usuarios hicieran anotaciones sobre documentos electrónicos pertenecientes a un grupo de noticias. El enfoque utilizado en este sistema era una mezcla de la técnica de filtrado basado en contenido y colaborativo. Basado en contenido porque permitía que los usuarios creasen filtros a través de sus ítems de interés y colaborativo, pues los usuarios ingresaban las anotaciones con sus opiniones sobre los documentos y de esa forma colaboraban entre ellos para filtrar la información. *Tapestry* utilizaba el filtrado colaborativo pero no lo hacía de forma automática. En este caso el sistema no era quien realizaba las sugerencias, sino que permitía a sus usuarios a partir de las anotaciones que se realizaban sobre los artículos, concluir si un determinado documento era de su interés (2).

GroupLens

Dos años más tarde surge *GroupLens*, quien fue el primer sistema automático de recomendación basado en el filtro colaborativo. Este sistema provee predicciones personalizadas para el grupo de noticias *Usenet*, ayudando a sus usuarios a encontrar artículos de su agrado dentro de la gran cantidad de artículos disponibles. *GroupLens* fue el primer sistema en usar el algoritmo basado en memoria para realizar el cálculo de las predicciones y el coeficiente de correlación de *Pearson* para calcular la similitud entre usuarios (2).

Ringo

En lo que respecta a la recomendación de música se encuentra *Ringo*, sistema web de recomendación de álbumes y artistas basado en el filtrado colaborativo, publicado en Internet el 1° de julio de 1994. *Ringo* utilizaba el algoritmo general basado en memoria para el cálculo de la predicción. Para el cálculo de la correlación utilizó una variante de la fórmula del coeficiente de correlación de *Pearson*, donde en vez de los *ratings* promedio de cada usuario utilizaba una constante, el valor medio de la escala de puntaje (2).

Avatar

Por otra parte el sistema de recomendación *Avatar*, es un sistema de recomendación híbrido de contenidos de vídeo que combina tres técnicas de filtrado:

dos de filtro colaborativo y una basada en el contenido. Este sistema se divide en tres módulos: el agente bayesiano (basado en el filtro colaborativo), que utiliza la información de los programas vistos y las preferencias de los usuarios para realizar las recomendaciones, el agente semántico (basada en el contenido), que utiliza la información actualizada sobre las preferencias de los usuarios y el agente de perfiles (basada en el filtro colaborativo), que compara distintos perfiles de usuarios para realizar las predicciones. Las recomendaciones de los tres agentes son combinadas utilizando técnicas de redes neuronales (11).

EBSCO

EBSCO no es en su totalidad un sistema de recomendación, sino una base de datos que realiza la búsqueda de información a través de *clusters*. La misma ofrece textos completos, índices y publicaciones periódicas académicas que cubren diferentes áreas de las ciencias y humanidades. Sus colecciones están disponibles a través de *EBSCOhost*, que es un sistema en línea que combina un contenido de gran calidad en páginas atractivas, con herramientas únicas de búsqueda y recuperación de información. Los resultados de la búsqueda ofrecen enlaces a los textos completos de los artículos. Además, permite seleccionar criterios para limitar o expandir la búsqueda por ejemplo: texto completo, referencias disponibles, publicaciones académicas, fecha de publicación, título publicado, tipo de publicación, tipo de documento, número de páginas, artículo de portada, artículos con imágenes, entre otros (20).

Después de realizar un análisis de los sistemas de recomendación anteriormente expuestos, se pudo constatar que la técnica más usada por la mayoría de estos es la basada en el filtro colaborativo, técnica que se basa solamente en los términos de similitud entre los usuarios para generar las recomendaciones, utilizando la correlación de *Pearson* para el cálculo de la similitud. Estos sistemas presentan características dentro de las que se pueden mencionar:

- Valorar un determinado objeto en dependencia de la preferencia del usuario.
- Comparar los perfiles de usuarios para determinar cuáles de estos son vecinos para realizar las predicciones.
- Permitir que los usuarios den a conocer en su perfil cuáles son sus preferencias.

Estas características sirvieron de base para el desarrollo del sistema de recomendación para el catálogo en línea de la biblioteca de la UCI, ya que se

descartaron los sistemas antes mencionados debido a que las necesidades y servicios que brindan los mismos son diferentes a las necesidades y servicios que brinda la biblioteca.

1.6 Metodología de desarrollo adoptada

La metodología RUP *Rational Unified Process*, es un proceso de desarrollo de software que junto con el Lenguaje Unificado de Modelado constituye la metodología estándar más utilizada para el análisis, implementación y documentación de sistemas orientados a objetos. Este se caracteriza por ser iterativo e incremental, dirigido por casos de uso y centrado en la arquitectura. Incluye artefactos (que son los productos tangibles del proceso como por ejemplo, el modelo de caso de uso, el código fuente, etc.) y roles (papel que desempeña una persona en un determinado momento, una persona puede desempeñar distintos roles a lo largo del proceso) (21). RUP es una metodología que ayuda a organizar y planificar todo el proceso de desarrollo de software con el objetivo de lograr un producto de calidad que satisfaga las necesidades del cliente.

1.7 Herramientas y tecnologías a utilizar

Para el desarrollo del sistema de recomendación es de vital importancia la correcta selección de las herramientas y tecnologías a utilizar, para garantizar un óptimo funcionamiento del mismo.

1.7.1 Eclipse 3.7.0

Eclipse es principalmente una plataforma de programación, usada para crear entornos integrados de desarrollo (del Inglés IDE). Fue desarrollado originalmente por IBM como el sucesor de su familia de herramientas para *VisualAge*. Es ahora desarrollado por la Fundación Eclipse, una organización independiente sin ánimo de lucro que fomenta una comunidad de código abierto. Cuenta con una característica ventajosa y es la de permitir la integración con *plugins* de programación para proporcionar todas sus funcionalidades al frente de la plataforma a diferencia de otros entornos monolíticos donde las funcionalidades están todas incluidas, sean de interés o no para el usuario. Uno de estos *plugins* es el EPIC (Editor de Perl y Entorno de Desarrollo Integrado para Eclipse), destinado para la programación en el lenguaje Perl (21).

1.7.2 EPIC

Es un Entorno de Desarrollo Integrado (IDE) de código abierto (con redactor y depuración de código incluidos) basado en la plataforma de Eclipse compatible con

Linux. Para la escritura de programas CGI o proyectos de Perl robustos con gran cantidad de módulos, EPIC es el más recomendable y extensible IDE disponible actualmente, gracias a la integración con todas las características principales y las convenciones de la Interfaz de Usuario Gráfica de Eclipse (21).

1.7.3 Visual Paradigm 8.0

Es una herramienta que soporta el ciclo de vida del desarrollo de software, la ingeniería inversa, generación de código, importación desde *Rational Rose*, importación y exportación XML, editor de figuras, generador de informes, etc. Es gratuito bajo la licencia GPL y comercial (21). Entre sus principales ventajas se encuentran:

- Soporta aplicaciones web.
- Generación de código para *Java* y exportación como HTML.
- Fácil de instalar y actualizar.
- Compatibilidad entre ediciones.

1.7.4 Mysql 5.1.62

MySQL Community Edition es una versión de descarga gratuita de la base de datos del mundo de código abierto más popular que es apoyada por una activa comunidad de desarrolladores de código abierto. *MySQL* pone a disposición una gran cantidad de herramientas que permiten establecer relaciones entre diferentes tablas, agregar nuevas columnas, editar las bases creadas, visualizar los ficheros SQL, agregar imágenes, etc. Cuenta con dos características que lo diferencian de otras aplicaciones similares: realizar cálculos relacionales y el manejo de álgebra. Gracias a ello, se puede obtener resultados mucho más precisos y objetivos (22).

1.7.5 Apache 2.2.20

Es un software libre de código abierto para plataformas *Unix*, *Windows*, *Macintosh* y otras, que implementa el protocolo HTTP/1.1 y la noción de sitio virtual. Apache presenta entre otras características mensajes de error altamente configurables, bases de datos de autenticación y negociado de contenido.

1.8 Lenguajes a utilizar

1.8.1 Perl 5.12

Es un lenguaje de programación muy utilizado para construir aplicaciones CGI (*Common Gateway Interface*) para la *web*. Perl es un acrónimo de *Practical Extracting and Reporting Language*, se trata de un lenguaje de programación gratuito muy

práctico para extraer información de archivos de texto y generar informes a partir del contenido de los ficheros (22). Tiene una gran potencia en la manipulación de textos debido a que incluye expresiones regulares que facilitan el trabajo con los mismos. Antes estaba muy asociado a la plataforma UNIX, pero en la actualidad está disponible en otros sistemas operativos como *Windows*.

1.8.2 HTML 3.0

Se conoce HTML (por sus siglas en inglés *Hypertext Markup Language*) como un lenguaje de marcado de hipertexto, diseñado para estructurar el contenido de las páginas web y presentarlo en forma de hipertexto. Básicamente se trata de un conjunto de etiquetas que marcan el inicio y el fin de cada elemento del documento. Se caracteriza por ser un lenguaje extensible, ya que se le puede añadir características, etiquetas y funciones adicionales para el diseño de páginas web generando un producto vistoso, rápido y sencillo (23).

1.8.3 CSS 2.1

Las hojas de estilo en cascada CSS (por sus siglas en inglés *Cascading Style Sheets*), es un lenguaje usado para definir la presentación de un documento estructurado escrito en HTML o XML (y por extensión en XHTML). Gracias a las CSS se pueden hacer muchas cosas que no se podrían hacer utilizando solamente HTML, como incluir márgenes, tipos de letra, fondos, colores, etc. Con este lenguaje se puede aplicar el mismo estilo a diferentes componentes de una aplicación web. Por lo que proporciona un control centralizado de la presentación de un sitio web completo. Esto permite agilizar de forma considerable la actualización de los mismos y disminuye el tiempo desarrollo de una aplicación de este tipo (24).

1.8.4 JavaScript

Es un lenguaje de *script* multiplataforma orientado a objetos pequeño y ligero. Está diseñado para una fácil incrustación en otros productos y aplicaciones, tales los navegadores web. *JavaScript* puede ser conectado a los objetos de su entorno para proveer un control programable sobre estos. Contiene un conjunto central de objetos como arreglos y fechas así como un conjunto de elementos de lenguajes como los operadores, estructuras de control y sentencias (23).

1.8.5 UML Lenguaje Unificado de Modelado

Es un lenguaje gráfico para visualizar, especificar, construir y documentar un sistema. Brinda un estándar para representar un plano del sistema, agregando aspectos conceptuales tales como: procesos del negocio, funciones del sistema,

expresiones de lenguajes de programación, esquemas de base de datos y componentes de software reutilizables (25).

Conclusiones parciales

En este capítulo se realizó un estudio bibliográfico sobre los sistemas de recomendación, obteniéndose los principales conceptos, características y antecedentes de los mismos. Se determinaron las técnicas de recomendación a utilizar a través del análisis de sus ventajas, llegando a la conclusión de que el sistema a desarrollar para el catálogo en línea de la biblioteca de la UCI es un sistema de recomendación híbrido combinando las técnicas basada en contenido y la basada en el filtro colaborativo, ya que el mismo permite mitigar las limitaciones que poseen estas técnicas y generar recomendaciones con mejores calidad. Se determinaron además las herramientas, tecnologías y metodología para el desarrollo del mismo.

Capítulo II: Estudio algorítmico

En el presente capítulo se describe la propuesta de un sistema de recomendación enfocado a facilitar la interacción de los usuarios con el catálogo en línea de la biblioteca de la UCI. Dicho sistema se centra fundamentalmente en sugerir materiales almacenados en el catálogo, basándose en la trayectoria y perfil de los usuarios. Como parte de esta descripción, se realiza un análisis de diferentes soluciones desde el punto de vista algorítmico y se propone el algoritmo a utilizar para el desarrollo del sistema.

2.1 Sistema de recomendación para el catálogo en línea de la biblioteca de la UCI

El sistema de recomendación que se propone utiliza los mecanismos de retroalimentación implícita y explícita para obtener información sobre el usuario, combinando las técnicas de recomendación basada en contenido y en el filtro colaborativo, conformando así un sistema de recomendación híbrido para un buen funcionamiento del mismo.

Este sistema es una vía de facilidad para el usuario en los procesos de búsqueda ya que, a la hora de obtener un material determinado, el usuario puede dirigirse a su perfil y ver las recomendaciones que tiene, si el material deseado se encuentra dentro de la lista de materiales recomendados, este se ahorraría tiempo en realizar una búsqueda del mismo en el catálogo, ya que el sistema brinda la posibilidad de localizar los materiales que puedan ser del interés para el usuario con mayor facilidad.

Para establecer una correcta correspondencia entre los materiales a recomendar y el usuario, es necesario asociarles a los usuarios una serie de metadatos que permitiría identificarlos dentro de un conjunto de semejantes. Estos datos son recolectados en su perfil y consisten en la especificación de los temas, autores y tipos de autoridades preferidos, para sugerirle materiales en correspondencia a estas preferencias.

El sistema permite además que los mismos usuarios puedan hacer recomendaciones: en el caso que alguno de ellos considere que algún material por determinadas razones les puede ser de utilidad a otro usuario, se les brinda la posibilidad de recomendar el mismo, el cual se mostrará en la lista de recomendaciones de usuarios en el perfil del usuario recomendado.

Otras de las posibilidades que brinda el sistema es la de valorar un material

determinado, donde la valoración consiste en darle un valor numérico dentro de una escala del 1-5, en dependencia de la preferencia que tenga el usuario sobre dicho material.

2.2 Algoritmos en sistemas de recomendación

Con el fin de dar una solución eficaz y computacionalmente eficiente al problema de brindar recomendaciones que se correspondan con los intereses de un usuario determinado, se han utilizados diferentes enfoques que pueden ser agrupados en dos categorías fundamentales: los algoritmos basados en memorias y los basados en modelo, de los cuales se realiza un estudio de los distintos algoritmos de filtrado colaborativo para elegir el mejor de ellos y, posteriormente, desarrollarlo en la práctica.

2.2.1 Algoritmos basados en memoria

Este algoritmo es uno de los métodos utilizados en las aplicaciones en sistema de filtro colaborativo. Puede utilizar los datos que se actualizan rápidamente y obtienen la predicción relativamente exacta. La técnica conocida como el método de los k vecinos más cercanos *k-nearest neighbor* o filtrado colaborativo basado en usuario *user-based collaborative filtering*, es una de las técnicas que se utilizan en el filtro colaborativo para el cálculo de las predicciones o recomendaciones (2).

Generalmente estos métodos emplean técnicas para encontrar vecinos, es decir, usuarios con un historial de valoraciones sobre los elementos similar al usuario actual. Una vez construida la lista de estos, se combinan sus preferencias para generar una lista con los n elementos más recomendables para el usuario actual. Entre sus inconvenientes se encuentra la necesidad de disponer de un número mínimo de usuarios con un número mínimo de predicciones cada uno, incluyendo al usuario para el cual se pretende realizar la recomendación (26).

2.2.1.1 K-Nearest Neighbor o vecinos más cercanos

Los algoritmos *k-nearest neighbor* o vecinos más cercanos fueron los primeros algoritmos de filtrado colaborativos en implementarse. Están conformados por tres etapas fundamentales: la representación de los datos, la formación de vecinos y por último, la generación de las recomendaciones, siendo la segunda la más crítica de las tres por poder afectar el rendimiento y eficiencia de las recomendaciones en el caso de los sistemas actuales que poseen miles de usuarios (16).

Para elegir el número de vecinos se puede:

1. Establecer un umbral de correlación y tomar todos los que superen dicho

umbral. El problema es que puede haber usuarios que no tengan muchos vecinos con correlación alta, por tanto el número de vecino sería bajo y esto provocaría que el número de elementos sobre los cuales la vecindad de un usuario puede opinar sea también bajo (26).

2. Tomar siempre n vecinos, teniendo en cuenta que un número demasiado alto puede diluir la influencia de los vecinos con más pesos y un número demasiado bajo provoca los mismos problemas del método anterior (26).

Se han definido varias medidas de similitud de elementos para estos algoritmos con vistas a utilizarlos como métrica durante el proceso de formación de vecinos. La más general de todas la constituye la distancia euclidiana, la cual se define por toda estructura que pueda asumir la forma [23]:

$$(a_1(x), a_2(x), a_3(x), \dots, a_n(x))$$

Ecuación 3

Donde $a_r(x)$ denota el valor del atributo r de la distancia x . Para esta representación, las distancias entre dos instancias, x_i y x_j definida como $d(x_i, x_j)$ viene dada por (16):

$$d(x_i, x_j) = \sqrt{\sum_{r=1}^n (a_r(x_i) - a_r(x_j))^2}$$

Ecuación 4

Esta función de cercanía no es propia de sistema de recomendación. No obstante, su aplicación en algunos casos puede dar resultados satisfactorios si se selecciona de manera eficaz el vector que representa los datos del usuario, con los cuales se determinarán las distancias (16).

Otras de las funciones de semejanzas, en este caso ya con considerables utilización en el campo de los sistemas de recomendación, lo constituye la similitud basada en coseno. Se considera cada elemento como un vector dentro de un espacio vectorial de n dimensiones y se calcula la similitud como el coseno del ángulo que forman. Es decir, si se tiene dos vectores r_i y r_j consistentes en un arreglo cuyos

elementos son las votaciones recibidas por cada usuario, su similitud estará dada por (16):

$$sim_{ik} = cos_{ik} = \frac{r_i * r_k}{||r_i|| * ||r_k||} = \sum_j \frac{r_{ij}}{\sqrt{\sum_j r_{ij}^2}} \frac{r_{kj}}{\sqrt{\sum_j r_{kj}^2}}$$

Ecuación 5

Esta ecuación permite calcular la proximidad entre dos usuarios u_i y u_k siendo r_{ik} la evaluación dada por el usuario u_i al ítem u_j .

Una manera más eficiente que la similitud basada en coseno para establecer semejanzas dentro de los sistemas de recomendación son las similitudes basadas en correlación, conjunto de votaciones de los usuarios con votos sobre el elemento x_1 y x_2 . El coeficiente de correlación de *Pearson* para los usuarios a y b viene dado por (16):

$$w_{a,b} = \frac{\sum_k (V_{a,k} - V_a)(V_{b,k} - V_b)}{\sqrt{\sum (V_{a,k} - V_a)^2 \sum (V_{b,k} - V_b)^2}}$$

Ecuación 6

Donde $V_{i,k}$ es el voto emitido por el usuario i acerca del ítem k , V_i es la medida de todos los votos para el usuario i y k es el conjunto de elementos evaluados tanto por el usuario a , como por el usuario b . En el caso de dar como resultado 1, esto se interpreta como que existe una correlación total, o en este caso relación de proporcionalidad directa total entre un vector y otro, en el caso de obtenerse -1, esto indicaría que la relación es proporcionalmente inversa total. En cambio, si el coeficiente de *Pearson* da como resultado 0, indica que la relación entre un vector y otro es nula o no existe (16).

2.2.2 Métodos basados en modelo

Este algoritmo es otro método de aplicaciones en sistema de filtrado colaborativo. Utiliza las preferencias de un usuario para aprender un modelo, el cual es usado para mostrar las predicciones o las recomendaciones (2).

El modelo resultante es pequeño, rápido y esencialmente tan exacto como los métodos basados en memoria. Los métodos basados en modelo, se pueden probar

prácticos en ambientes en los cuales las preferencias del usuario cambian lentamente con respecto al tiempo necesario para construir el modelo. Sin embargo, no es conveniente para ambientes en los cuales los modelos de preferencias del usuario se actualizan rápidamente o con frecuencia. En este caso no se tiene en cuenta de forma explícita la similitud entre el usuario activo y los demás usuarios del sistema. Las técnicas a utilizar en los algoritmos de filtrado colaborativos basados en modelos son las redes bayesianas y *clustering* (16).

2.2.2.1 Redes bayesianas

El modelo de las redes bayesianas resulta más efectivo que el modelo *clustering*. Una red bayesiana es una red donde los nodos se corresponden con cada ítem del dominio de la aplicación y los estados de los nodos, con los posibles valores de votos sobre los *ítems* (2). La información cuantitativa a suministrar para cargar la red está dada por las probabilidades a priori de los nodos que no tienen padres y la probabilidad condicionada de los nodos con padres, tal como se muestra en la figura (17):

Red bayesiana con parámetros

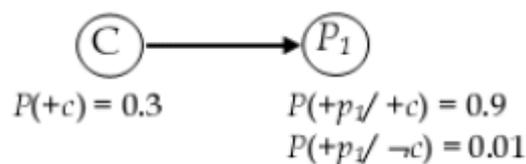


Ilustración 1

Es preciso dar a conocer que este modelo cuenta con algunas desventajas importantes. En primer lugar, es necesario disponer de una gran cantidad de datos de entrenamiento para poder lograr un modelo de gran tamaño que permita realizar buenas predicciones. Sin embargo, contar con un conjunto de datos importantes puede ocasionar que el modelo sea extremadamente grande, requiriendo mucho tiempo de entrenamiento. En los sistemas de filtrado colaborativo donde la escala de valoración es demasiado grande, aplicar este modelo resultaría complejo; ya que precisamente los estados de cada nodo son los posibles valores de puntajes sobre los cuales el usuario puede darle valores a los *ítems*, no siendo así en los sistemas donde se considera una escala binaria. Por tal motivo, se recomienda utilizar este modelo en sistema de filtrado colaborativo donde la escala de valoración es binaria o de tamaño reducido. A pesar de estas desventajas, este modelo presenta la ventaja de ser muy rápido a la hora de calcular las predicciones luego de tener el modelo entrenado (2).

2.2.2.2 Clustering

Las técnicas de *clustering* trabajan a través de la identificación de grupos o *cluster* con usuarios que evidencian tener semejantes intereses. Una vez que los *cluster* son creados, las predicciones de cualquiera de sus miembros pueden ser hechas promediando las opiniones del resto de los usuarios en el *cluster* correspondiente (16).

Este enfoque directamente aplicado produce recomendaciones que en muchos casos tiene menor calidad que las obtenidas por herramientas similares. No obstante, es importante resaltar que su principal objetivo no es el de sugerir preferencias, sino el de reducir la dimensión del espacio contenedor de todos los datos de los usuarios e *ítems*, particionándolos en varios espacios más pequeños con menor cantidad de *ítems*, menor número de predicciones pre-almacenados y menor cantidad de usuarios, pudiendo entonces aplicarse algoritmos de filtrado colaborativo que anteriormente no se podía utilizar garantizando la eficiencia, por la gran cantidad de elementos y asociaciones en el espacio inicial (16).

2.3 Limitaciones en los algoritmos en sistemas de recomendación

El hecho de trabajar en sentido general con información con alto grado de incertidumbre ocasiona que los métodos para establecer recomendaciones se encuentran con limitaciones que pueden dificultar la concepción de su objetivo final. A la hora de concebir un sistema de recomendación es importante tener en cuenta estas limitaciones. A continuación se realiza una breve descripción de las más importantes.

2.3.1 Arranque frío

Constituye uno de los problemas más difícil de resolver al inicio del despliegue de un sistema de recomendación, independientemente del objetivo con que se haya implementado. Este problema se debe a que en un principio se cuenta con muy pocas evaluaciones o consideraciones para realizar las recomendaciones, a consecuencia de tres razones diferentes:

- **Nuevo usuario:** cuando un nuevo usuario se registra en el sistema, existen muy pocos datos disponibles para definir con cierta precisión su perfil
- **Nuevo ítem:** cuando un nuevo *ítem* es insertado al catálogo, inicialmente no presenta evaluaciones.

- **Nuevo sistema:** cuando se pone a funcionar una nueva instancia de un sistema de recomendación, el número promedio de evaluaciones o consideraciones por usuario o por *ítem* es bajo, lo que puede afectar significativamente la calidad de las recomendaciones.

Las estrategias para mitigar este tipo de problemas han sido la utilización de los sistemas de recomendación híbrida que combinen las recomendaciones basadas en contenido con el filtro colaborativo y el empleo de criterios emergentes tales como la popularidad y la entropía de los *ítems* (16).

2.3.2 Dispersión de los datos

En mucho de los sistemas de recomendaciones que han sido comprobados en la práctica se tiene que, el número de evaluaciones reales que se dispone acerca de un ítem en específico queda muy por debajo del número mínimo que se requiere para hacer las recomendaciones, haciéndose particularmente críticos para aquellos con preferencias alejadas de las de la mayoría. Esto implica que la predicción efectiva a partir de un número reducido de ejemplos juegue un papel clave para el éxito del sistema (16).

Una de las formas para dar solución a este tipo de problema es la de utilizar los datos del usuario almacenados en su perfil para establecer la similitud del mismo. De esta forma, para determinar si un usuario es semejante a otro usuario, no solo se tomaría en cuenta si ambos muestran su preferencia por determinados *ítems*, sino que también se tomaría en cuenta los datos que contiene su perfil.

2.3.3 La sobre-especialización

La sobre-especialización está directamente asociada con los sistemas de recomendación basados en contenidos. Esto se debe cuando en estos sistemas solamente se recomiendan *ítems* que se correspondan a los que están almacenados en el perfil del usuario, obteniendo como resultado recomendaciones muy similares, puesto a que estas siempre se basan en la misma información y no tienen en cuenta las similitudes que pueden existir con otros usuarios, limitando de esta forma sugerencias de elementos similares que los mismos hayan evaluado o considerado.

Como alternativa para garantizar la diversidad de las recomendaciones para los usuarios, en muchos de estos sistemas se ha dado cabida a la aleatoricidad, a través de técnicas como la de los algoritmos genéticos. El uso de sistemas de recomendaciones híbridas, donde se utilicen como basamento para la recomendación

tanto la información del propio usuario como la de usuarios semejantes, también se ha utilizado para evitar la tendencia a la aparición de este problema (16).

2.4 algoritmo propuesto para el sistema de recomendación

Para el desarrollo del sistema de recomendación para el catálogo en línea de la biblioteca de la UCI se optó por una variación del algoritmo del filtrado colaborativo basado en el vecino más cercano, ya que es el que mejor se adapta a la propuesta de solución.

Obtención de los votos

Inicialmente se obtienen los votos realizados por el usuario actual (usuario autenticado). La apreciación del usuario sobre los objetos se mide con la siguiente escala:



- 1: Falta mucha información.
- 2: Contiene algo de información.
- 3: Presenta información clave, pero con carencia.
- 4: Contiene la mayoría de información clave.
- 5: Cobertura completa.

Definición del vecindario

La definición del vecindario consiste en buscar todos los usuarios que votaron por el mismo objeto que el usuario autenticado. Ya cargado los datos se inicializa el vecindario, el vecindario está constituido por todos los usuarios que tienen un nivel de correlación con el autenticado.

2.4.1 Definición de la correlación de Pearson

Se definen los pesos para cada usuario con base al coeficiente de la correlación de *Pearson*, índice que mide el grado de dispersión conjunta de dos variables relacionadas linealmente. Sus valores absolutos oscilan entre 0 y 1. Dicho coeficiente se usa para determinar el nivel de cercanía o relación entre los usuarios con respecto a la percepción que estos tienen sobre los objetos.

En la ecuación (8) el peso $w_{a,b}$ que se asigna al usuario b para predecir al usuario actual autenticado a viene dado por: $r_{a,i}$ que corresponde a la votación del usuario a al elemento i .

$$w_{a,b} = \frac{\sum_{i=1}^m (r_{a,i} - \bar{r}_a) * (r_{b,i} - \bar{r}_b)}{\sigma_a \sigma_b}$$

Ecuación 7

Para calcular el coeficiente de correlación de *Pearson*, se tiene en cuenta las siguientes consideraciones:

r_a	Usuario actual autenticado
$r_{a,i}$	Voto del usuario actual por el objeto i
\bar{r}_a	Promedio de los votos del usuario actual
$r_{b,i}$	Voto de un vecino u por el mismo objeto i
r_b	Un vecino del usuario autenticado
\bar{r}_b	Promedio de los votos del vecino b
σ_a	Desviación estándar de a (usuario autenticado)
σ_b	Desviación estándar de b (vecino del usuario autenticado)
m	Números de votos en común entre los usuarios

Tabla 1: Consideraciones para calcular el coeficiente de correlación de *Pearson*

La desviación estándar especifica el grado de dispersión de un grupo de datos de su media aritmética. En este caso determina que tanto se aleja los votos del usuario sobre los objetos del promedio de votos.

La tendencia de votación del usuario autenticado se da por la desviación estándar (σ_a) y la tendencia de votación del vecino actual se define por la desviación estándar (σ_b). Dado lo anterior, las respectivas fórmulas para el cálculo serían:

$$\sigma_a = \sqrt{\frac{\sum_{i=1}^m (r_{a,i} - \bar{r}_a)^2}{\sigma_b}}$$

Ecuación 8

$$\sigma_b = \frac{\sqrt{\sum_{i=1}^m (r_{b,i} - \bar{r}_b)^2}}{m}$$

Ecuación 9

La fórmula anterior calcula la desviación estándar que es igual a la raíz cuadrada de la varianza, si el usuario tiene más de un voto. Si sólo tiene un voto la desviación es igual al valor votado y si no ha realizado voto alguno, la desviación estándar es igual al valor máximo que puede tener un voto. Si la desviación estándar de los votos del vecino por la desviación estándar de los votos del usuario autenticado es diferente de cero, se calcula el peso dividiendo la suma entre el producto de las desviaciones. De lo contrario, el peso del vecino con respecto al usuario autenticado es cero.

Delimitación del vecindario

Los pesos asignados a cada vecino del vecindario se ordenan de forma descendiente según el peso de cada uno, mediante la utilización del método de ordenamiento burbuja para luego seleccionar los n primeros (27). Delimitado el vecindario se definen los objetos (sin duplicidad) que servirán como recomendaciones, lo cual se logra sacando todos los objetos que han sido votados por los vecinos y que no ha hayan sido votados por el usuario autenticado.

2.4.3 Definición de las recomendaciones

Consiste en encontrar los objetos que hayan sido de interés para los vecinos del usuario autenticado y por los cuales se haya realizado una valoración (votación) y que el usuario autenticado no haya votado aún. Es importante resaltar que si dos o más vecinos han valorado el mismo objeto este se escoge solo una vez. Luego se obtienen los objetos por los cuales los demás usuarios hayan votado, pero que el usuario autenticado no haya votado por ellos.

Definición de los pesos de las recomendaciones

Las valoraciones realizadas a un objeto por los vecinos del usuario autenticado y la cantidad de vecinos que hayan valorado dicho objeto, van a definir el peso del mismo o qué tan recomendable es para el usuario autenticado. Estas valoraciones se asocian con su desviación estándar para no sesgarla, según el nivel de optimismo de las valoraciones de los usuarios.

$$p_{a,i} = r_a + \sigma_a \frac{\sum_{b=1}^n \frac{r_{b,i} * \bar{r}_b}{\sigma_a} * w_{a,b}}{\sum_{b=1}^n w_{a,b}}$$

Ecuación 10

Para calcular el peso de las recomendaciones $P_{a,i}$ se tiene en cuenta las siguientes consideraciones: r_a denota el usuario autenticado, σ_a representa la tendencia de votación del usuario autenticado, $r_{b,i}$ son los votos de un vecino b por el mismo objeto i , mientras que \bar{r}_b se refiere al promedio de los votos del vecino b y el peso asignado al usuario autenticado viene dado por $w_{a,b}$. Las recomendaciones se ordenan por el peso y se seleccionan los n primeros para sugerirle al usuario autenticado.

2.4.5 Tratamiento del arranque frío

Para un buen funcionamiento y una mejor eficiencia para realizar las recomendaciones, el sistema de recomendación propuesto utiliza la combinación de los dos tipos de sistemas de recomendación más comunes, la basada en contenido y en el filtro colaborativo, conformando un sistema de recomendación híbrida. Para solucionar el problema del arranque frío se tuvo en cuenta lo siguiente:

- **Primero:** en caso de que un usuario sea nuevo en el sistema, este recomienda los materiales que han sido calificados con una evaluación de 4 ó 5 puntos por otros usuarios, ya que estos son los materiales más gustados por los mismos.
- **Segundo:** en caso de que un usuario sea nuevo en el sistema y de no existir votación alguna sobre los materiales, el sistema especifica que no se han encontrado contenidos bibliográficos y le recomienda que debe actualizar su perfil para poder ser recomendado.

Conclusiones parciales

En esta capítulo se realizó una descripción detalla del sistema de recomendación, el cual servirá como vía de facilidad para los usuarios de la biblioteca en los procesos de búsqueda, ya que el mismo brindará a estos recomendaciones de los materiales disponibles en el catálogo de acuerdo a sus preferencias. Además se hizo un estudio de los principales enfoques algorítmicos utilizados para dar solución efectiva al problema de la recomendación, definiéndose dos tipos de algoritmos: los basados en memoria y los basados en modelos. Como parte de los enfoques se analizaron los

métodos del vecino más cercano, asociados al primer tipo y las redes bayesianas y *clustering* asociados al segundo tipo. Finalmente se eligió el algoritmo a utilizar para el desarrollo del sistema de recomendación, siendo este el algoritmo basado en el vecino más cercano, ya que es el que mejor se adapta para darle solución al problema planteado.

Capítulo III: Características del sistema de recomendación

En el presente capítulo se llevará a cabo la descripción de la propuesta de solución para el presente trabajo, proporcionando un mejor entendimiento del funcionamiento del sistema. Además se presentarán los requerimientos que debe tener el sistema divididos en requisitos funcionales, que muestran las funcionalidades específicas que serán implementadas y representadas a través de los casos de uso y los no funcionales, que definirán las características y restricciones que deberá cumplir la aplicación para poder funcionar correctamente. También incluirá la descripción de los procesos que serán objeto de automatización, diagramas de casos de uso y modelo de dominio.

3.1 Propuesta de solución

Se propone implementar un sistema de recomendación que haga recomendaciones a los usuarios de materiales de acuerdo a su interés. Este sistema utilizará los mecanismos de realimentación explícita e implícita para obtener información sobre los usuarios, combinando las técnicas de recomendación basada en contenido y la basada en el filtrado colaborativo, conformando así un sistema de recomendación híbrida para mitigar las dificultades que puedan presentar las mismas y obtener un buen funcionamiento y calidad en las recomendaciones, basándose principalmente en las preferencias y perfiles de usuarios. Este último estará compuesto por los autores, materiales y tipos de autoridades preferidos por los mismos donde, según sus preferencias dará a conocer los que más le gustan. El sistema además, hará uso de una variación del algoritmo de filtrado colaborativo vecino más cercano para generar las recomendaciones.

Modelo del dominio

Después de realizar un estudio no se lograron identificar procesos de negocios claros en el marco de la investigación, solo elementos conceptuales, por lo que se decide realizar un modelo de dominio. Con este modelo se pretende lograr una mejor comprensión de como ocurren los procesos, dando paso a la identificación de los requisitos funcionales. A continuación se muestra en la ilustración 2 el modelo de dominio donde se describen los principales conceptos relacionados con el sistema, así como las relaciones entre ellos.

CARACTERÍSTICAS DEL SISTEMA DE RECOMENDACIÓN

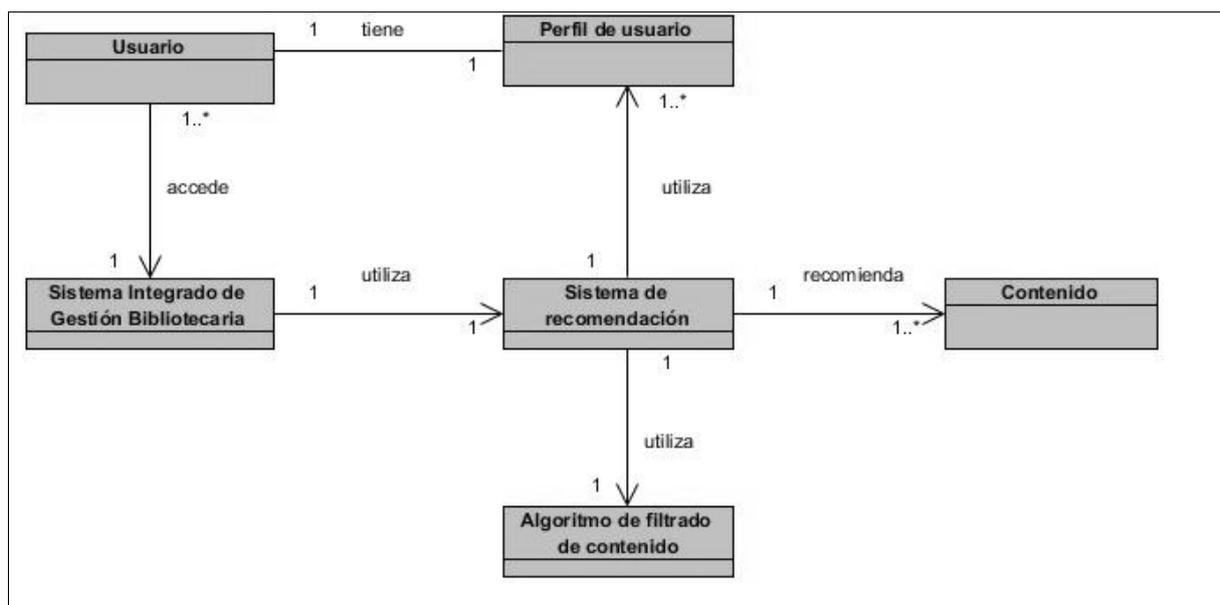


Ilustración 2: Modelo del dominio

Definición de los conceptos del modelo del dominio:

- **Usuario:** es la persona que interactúa con el sistema y sus funcionalidades.
- **Perfil de usuario:** lugar donde se recogen los datos y preferencias del usuario.
- **Sistema de recomendación:** es el sistema o aplicación que recomienda registros o contenidos a los usuarios según las preferencias basadas en el perfil y de las preferencias de otros usuarios con gustos similares.
- **Material:** es el material que el sistema o usuarios recomienda.
- **Sistema Integrado de gestión Bibliotecaria:** sistema que automatiza los procesos y servicios que brinda la biblioteca a sus usuarios.
- **Algoritmo de filtrado de contenido:** algoritmo que utiliza el sistema de recomendación para el filtrado de información.

3.2 Especificación de requisitos

La ingeniería de requisitos facilita el mecanismo apropiado para comprender lo que desea el cliente, analizando conformidades, confirmando su viabilidad, negociando una solución razonable, especificando la solución sin ambigüedad, validando la especificación y gestionando los requisitos para que se transformen en un sistema operacional. Los requerimientos para un sistema de *software* determinan lo que hará el sistema y definen las restricciones de su operación e implementación. [32].

CARACTERÍSTICAS DEL SISTEMA DE RECOMENDACIÓN

Requisitos funcionales

Los requisitos funcionales del sistema definen las funciones que el sistema será capaz de realizar. Son un conjunto de condiciones o capacidades que debe cumplir el software para que tenga éxito en el entorno en el cual se implantará y usará. Estos deben ser comprensibles por los clientes, usuarios y desarrolladores, deben tener una sola interpretación y estar definidos de forma medible y verificable.

RF-1 Recomendar material a un usuario: permite al usuario recomendar una material a otro usuario.

RF-2 Mostrar recomendaciones automáticas: permite mostrar al usuario la lista de recomendaciones que le ha hecho el sistema.

RF-3 Mostrar recomendaciones de usuarios: permite mostrar al usuario la lista de recomendaciones que le han hecho otros usuarios del sistema.

RF-4 Ponderar material: permite al usuario evaluar los materiales en dependencia de su criterio sobre los mismos.

RF-5 Mostrar ponderación de registro: permite mostrar al usuario el promedio de las evaluaciones sobre los materiales.

RF-6 Eliminar recomendaciones de usuarios: permite eliminar al usuario las recomendaciones que le han hecho otros usuarios del sistema.

Requisitos no funcionales

Los requisitos no funcionales especifican propiedades o cualidades que el software debe tener. Estos de una forma u otra restringen el entorno del sistema o de la implementación como por ejemplo, interfaz de usuario, rendimiento entre otros.

- **Usabilidad:** el sistema 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. El sistema está concebido para ser usado por una gran cantidad de usuarios por lo que se hace necesario, a pesar de que el nivel de experiencia sea medio, cuente con un diseño de interfaz de fácil uso para los mismos.
- **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.

CARACTERÍSTICAS DEL SISTEMA DE RECOMENDACIÓN

- **Eficiencia:** las páginas no estarán cargadas de imágenes para que la ejecución de las funcionalidades no exceda los 5 segundos y garantizar una respuesta rápida y eficiente del sistema.
- **Requisitos de soporte:** el sistema debe ser configurable a diferentes sistemas operativos. Debe mostrar la misma interfaz gráfica con distintos navegadores web. Se utilizará como servidor web Apache v2, Perl como lenguaje de programación y MySQL como gestor de base de datos.
- **Requerimientos de interfaz o apariencia:** se necesita una interfaz amigable, profesional, fácil, sencilla y clara de usar para la interacción con los usuarios, con el objetivo de evitar la resistencia humana al uso del sistema, ya que el factor humano determina en gran medida el éxito o fracaso del mismo.

3.3 Definición de los actores y casos de usos

Los actores del sistema representan entidades externas que interactúan directamente con el sistema (personas, maquinas u otros sistemas) y los casos de uso son parte del análisis que ayudan a describir qué es lo que el sistema debe hacer desde el punto de vista del usuario.

Actores	Justificación
Usuario	Es la persona que interactúa directamente con el sistema.

Tabla 2: Descripción de los actores del sistema

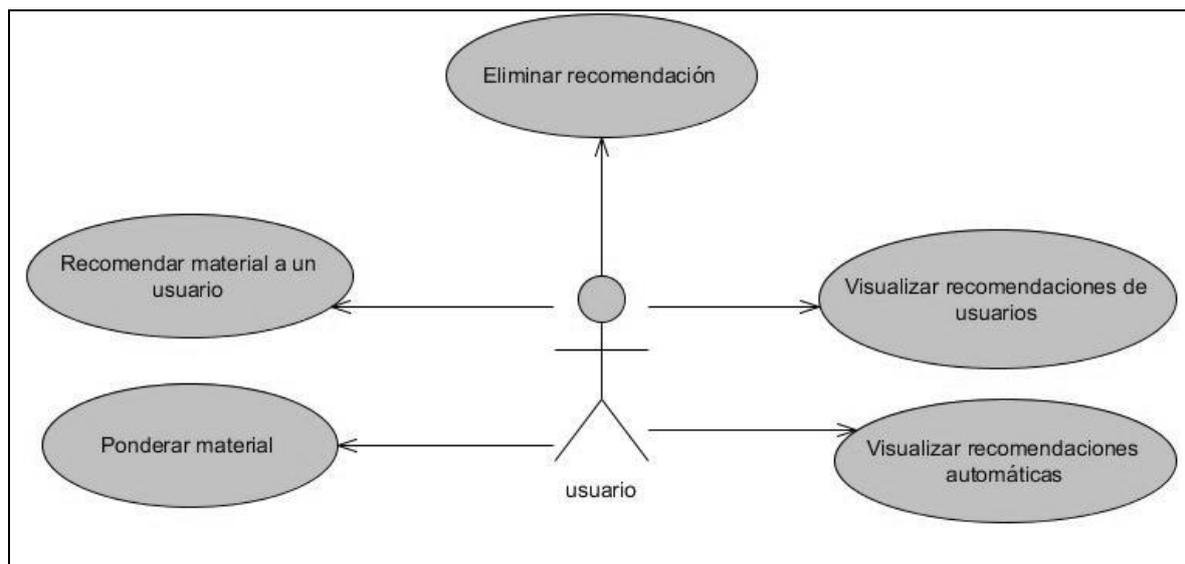


Ilustración 3: Modelo de casos de uso

CARACTERÍSTICAS DEL SISTEMA DE RECOMENDACIÓN

Descripción del caso de uso recomendar material a un usuario

CU_1	Recomendar material a un usuario
Actor	Usuario
Descripción	Permite al usuario recomendar contenidos a otro usuario que pueden ser del interés para el mismo.
Referencia	RF-1

Tabla 3: Descripción de caso de uso Recomendar a un usuario

Descripción del caso de uso visualizar recomendaciones automáticas

CU_2	Visualizar recomendaciones automáticas.
Actor	Usuario
Descripción	Permite al usuario visualizar las recomendaciones realizadas por el propio sistema.
Referencia	RF-2

Tabla 4: Descripción del caso de uso Visualizar recomendaciones automáticas

Descripción del caso de uso visualizar recomendaciones de usuarios

CU_3	Visualizar recomendaciones de usuarios.
Actor	Usuario
Descripción	Permite al usuario visualizar las recomendaciones realizadas por otros usuarios.
Referencia	RF-3

Tabla 5: Descripción del caso de uso Visualizar recomendaciones de usuarios

Descripción del caso de uso ponderar material

CU_4	Ponderar material.
Actor	Usuario
Descripción	Permite al usuario darle valor numérico a un material de acuerdo a su preferencia dentro de un rango determinado, una vez de ser puntuado el material se visualiza el promedio de las puntuaciones de 1 a 5 estrellas.
Referencia	RF-4, RF-5

Tabla 6: Descripción del caso de uso: Ponderar registro

Descripción del caso de uso eliminar recomendación

CU_3	Eliminar recomendación.
Actor	Usuario
Descripción	Permite al usuario eliminar las recomendaciones de usuarios que él desea.
Referencia	RF-6

Tabla 7: Descripción del caso de uso: Ponderar registro.

Descripción textual de los caso de usos del sistema de recomendación. Ver [anexo 1](#).

Conclusiones parciales

Con la realización de este capítulo se planteó la estructura de datos de la propuesta de solución definiendo el formato que tendrá la base de datos del sistema de recomendación. Se define la propuesta del sistema presentando un listado de requisitos funcionales y no funcionales donde se recogen las principales necesidades del sistema de recomendación a desarrollar. Estas necesidades fueron traducidas a un conjunto de casos de usos que representan las principales funcionalidades del sistema y se realizaron las descripciones detalladas de los mismos.

Capítulo IV: Análisis y diseño

Este capítulo tiene como propósito modelar artefactos que tiene lugar durante el flujo de trabajo análisis y diseño. En él se traducen los requisitos a una especificación que describe cómo implementar el sistema. Se realiza el análisis para obtener una visión del sistema, que se preocupa en ver cómo se hace desde el punto de vista de los requisitos funcionales, obteniendo un modelo de clases del análisis por cada caso de uso. Por otro lado se realiza el diseño para refinar los modelos obtenidos del análisis, partiendo de los requisitos no funcionales y ver cómo cumple el sistema sus objetivos, representado en diagramas de clases del diseño y diagramas de interacción respectivamente.

4.1 Análisis del sistema

El análisis consiste en tener una visión que se preocupa de ver qué hace, de modo que sólo se interesa por los requisitos funcionales. El objetivo del análisis es comprender los requisitos del software y no precisar cómo se implementará. A continuación se describirán las clases que serán utilizadas en la realización de los diagramas de clases del análisis de cada caso de uso.

CI_<Nombre de la clase>: Estas clases modelan la interacción entre los actores y el sistema.

CC_<Nombre de la clase>: Estas clases coordinan la relación de los casos de uso y además, coordinan el trabajo de la clase interfaz y entidad.

CE_<Nombre de la clase>: Estas clases modelan información que posee larga vida y que es a menudo persistente.

4.1.1 Clases del análisis

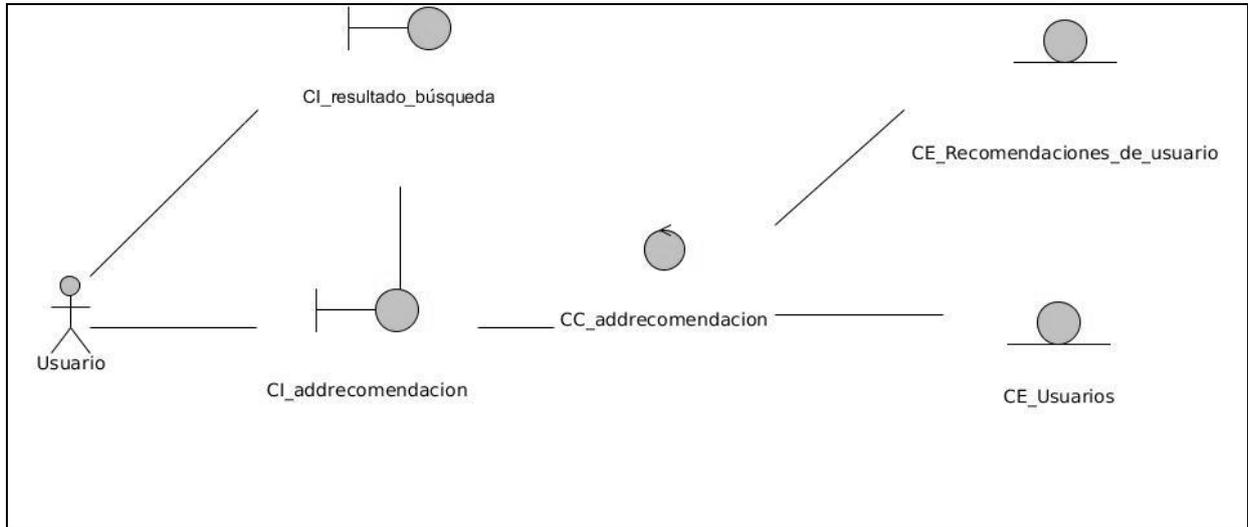


Ilustración 4: Clase del análisis del CU_Recomendar material a un usuario

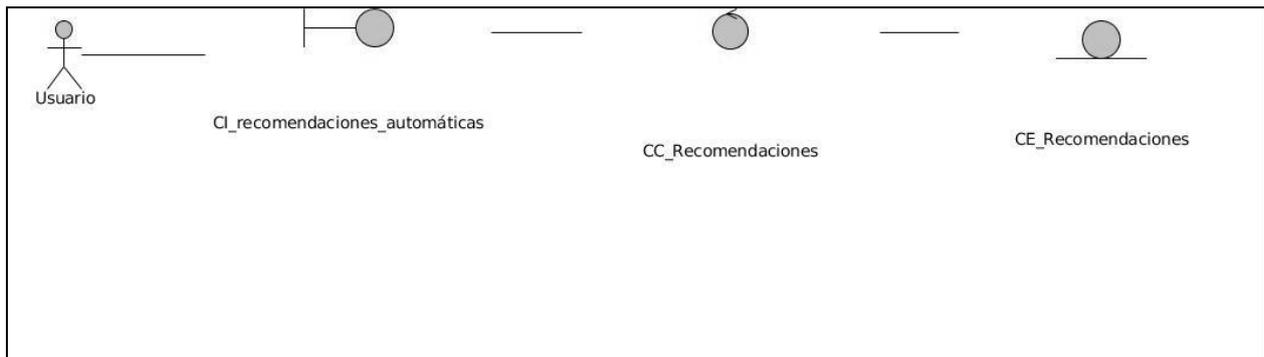


Ilustración 5: Clase del análisis del CU_Visualizar recomendaciones automáticas

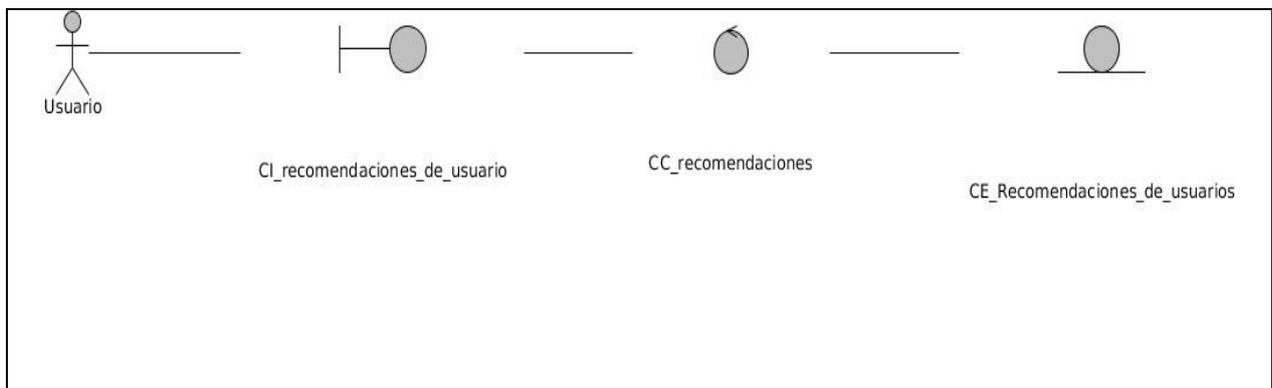


Ilustración 6: Clase del análisis del CU_Visualizar recomendaciones de usuario

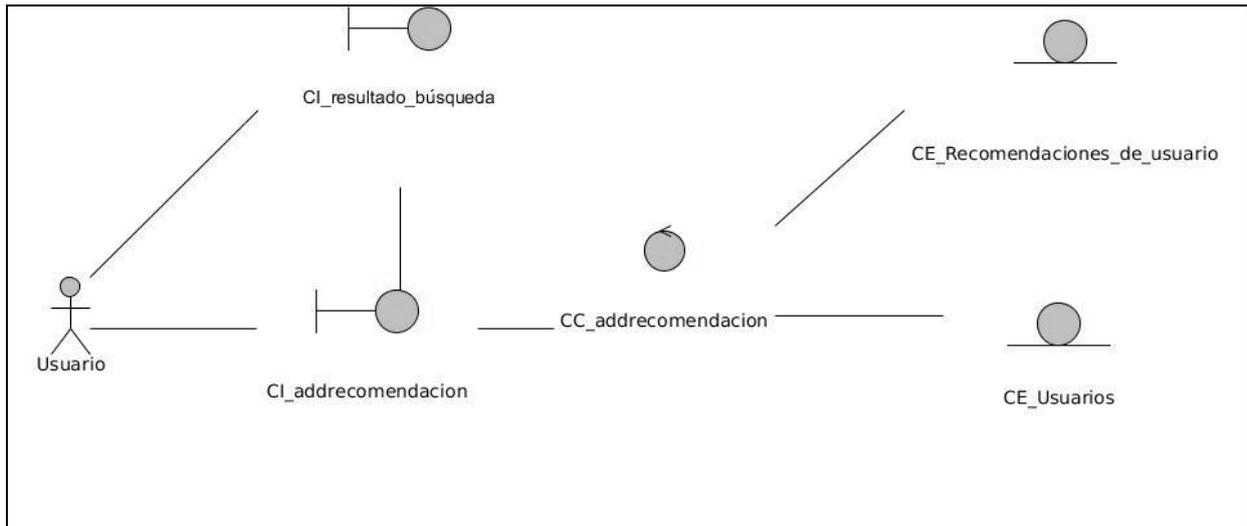


Ilustración 7: Clase del análisis del CU_Ponderar material

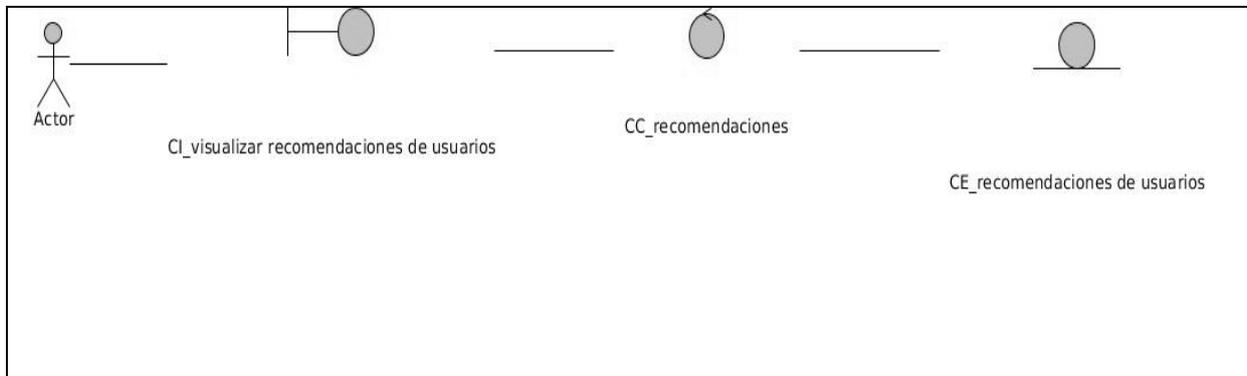


Ilustración 8: Clase del análisis del CU_Eliminar recomendación

4.1.2 Diagramas de interacción

Los diagramas de interacción se utilizan para modelar los aspectos dinámicos de un sistema. La mayoría de las veces, esto implica modelar instancias concretas de clases, interfaces, componentes y nodos, juntos con los mensajes enviados entre ellos y en el contexto de un escenario que ilustra un comportamiento. Los diagramas de interacción pueden utilizarse para visualizar, especificar, construir y documentar la dinámica de una sociedad particular de objetos o se puede utilizar para modelar un flujo de control particular de cada caso de uso.

Representación de los diagramas de colaboración

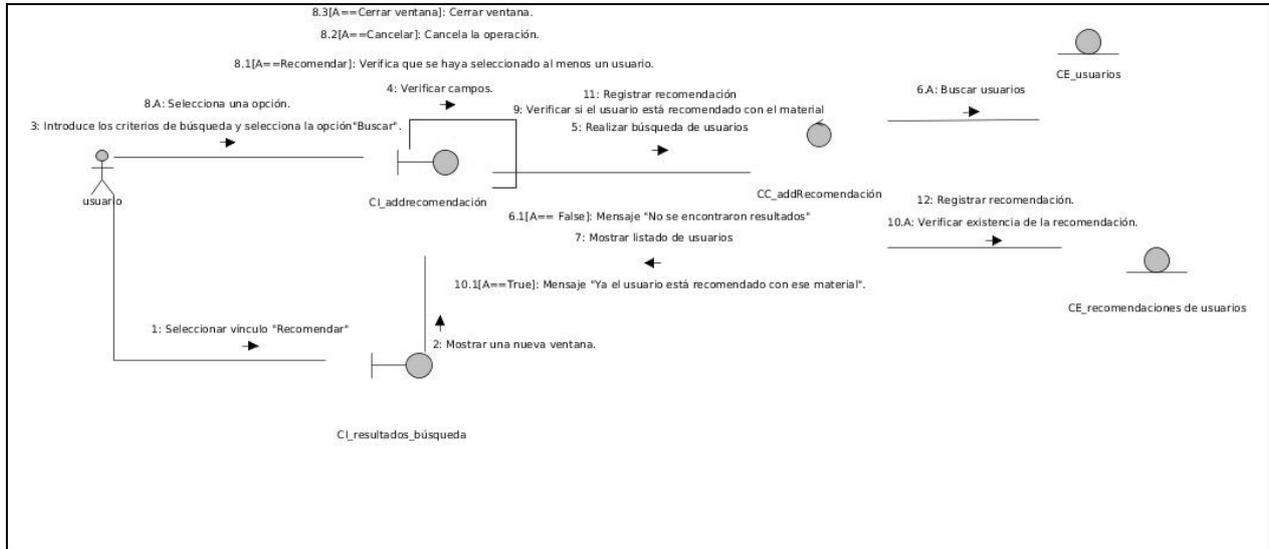


Ilustración 9: Diagrama de colaboración del CU_recomendar material a un usuario

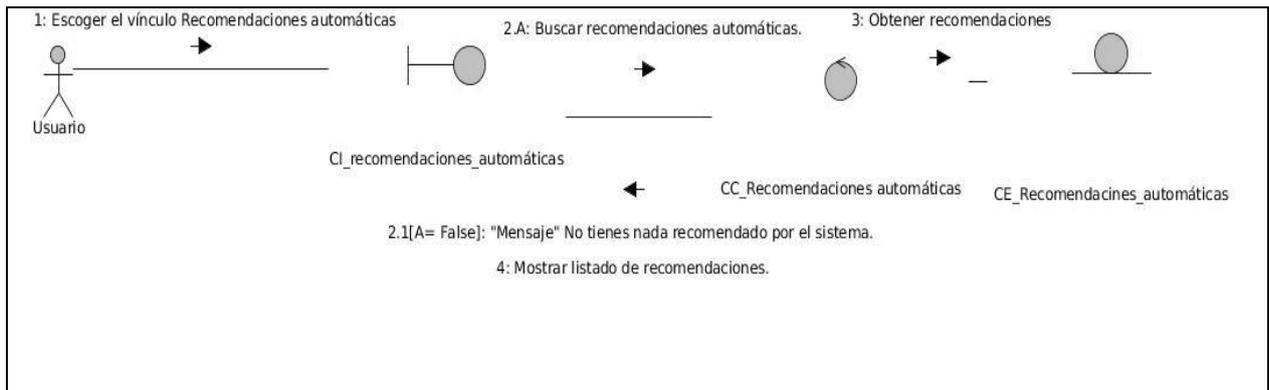


Ilustración 10: Diagrama de colaboración del CU_visualizar recomendaciones automáticas.

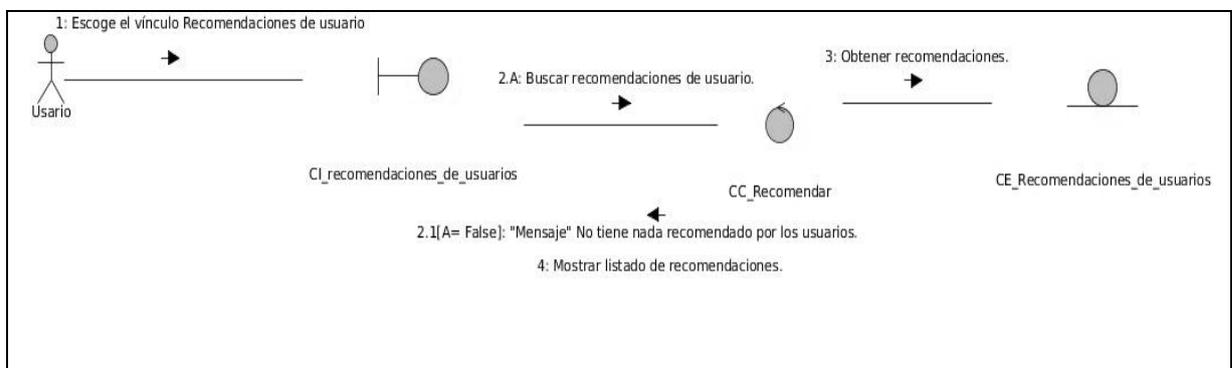


Ilustración 11: Diagrama de colaboración del CU_visualizar recomendaciones de usuario

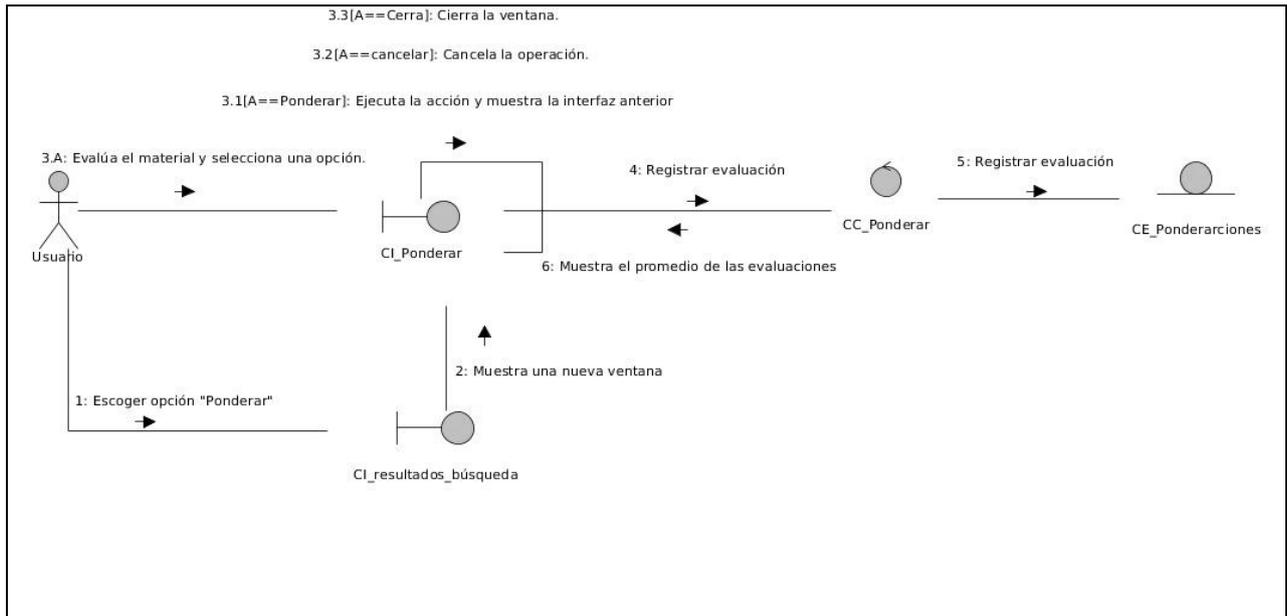


Ilustración 12: Diagrama de colaboración del CU_ponderar material

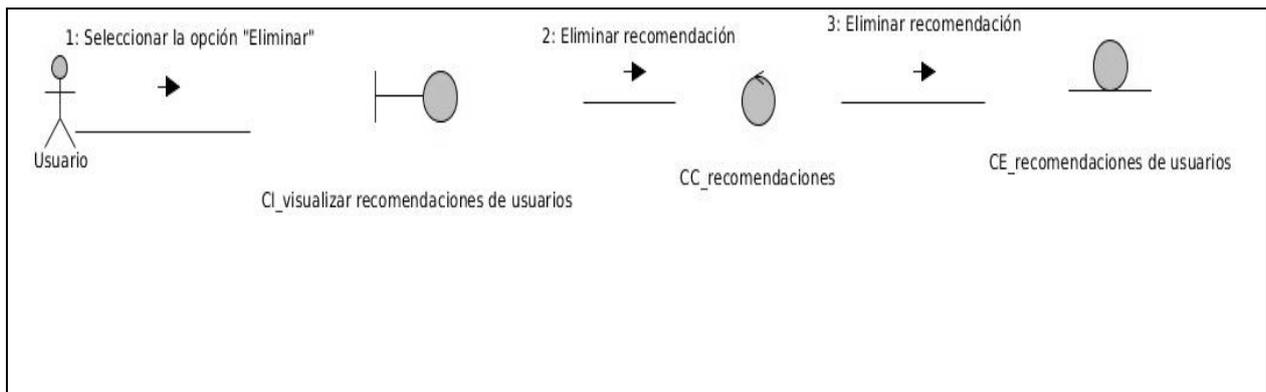


Ilustración 13: Diagrama de colaboración del CU_eliminar recomendación

4.2 Diseño del sistema

En el diseño se tiene el propósito de formular los modelos para preparar la entrada a las actividades de implementación y prueba del sistema, preparando un plano para los artefactos que se crean durante cada uno de estos flujos de trabajo. En este flujo de trabajo se modela el sistema y se encuentra la forma que soporte todos los requisitos, incluyendo los nos funcionales y las restricciones que lo supone.

Clases del diseño

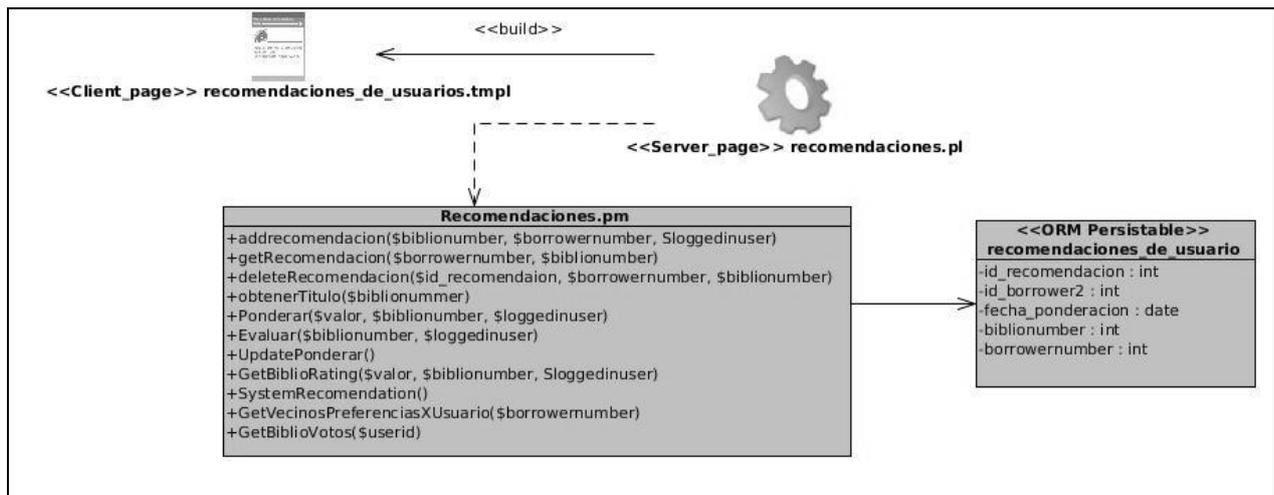


Ilustración 14: Diagrama de clases del diseño del CU Visualizar recomendaciones de usuarios

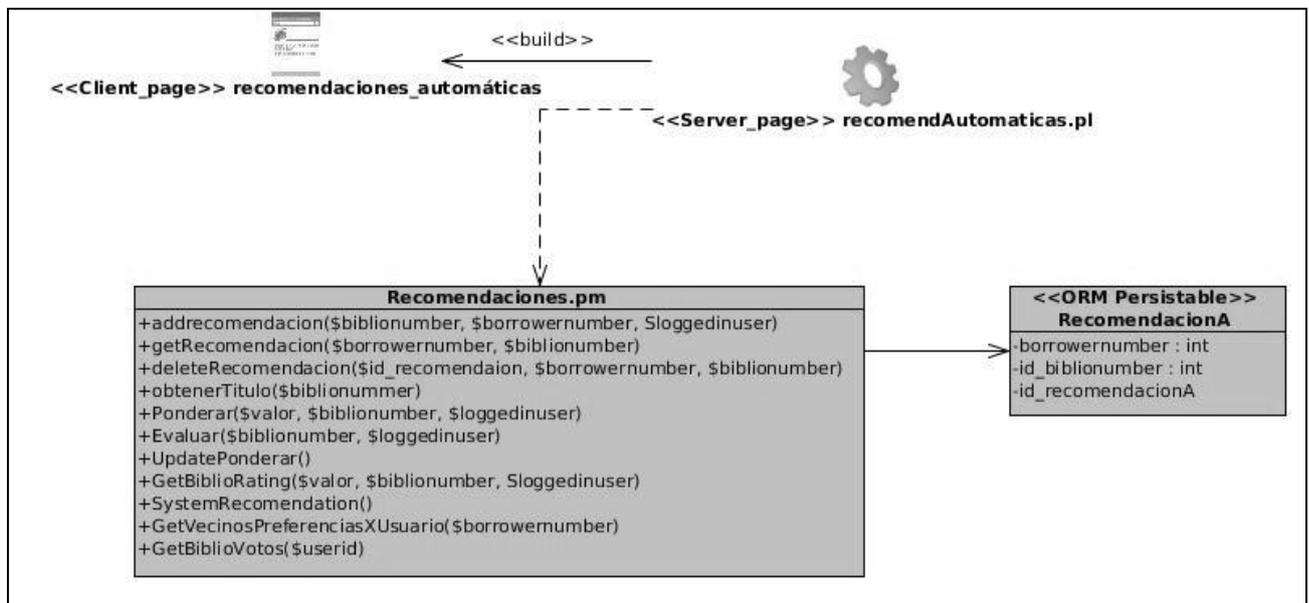


Ilustración 15: Diagrama de clases del diseño del CU Visualizar recomendaciones automáticas

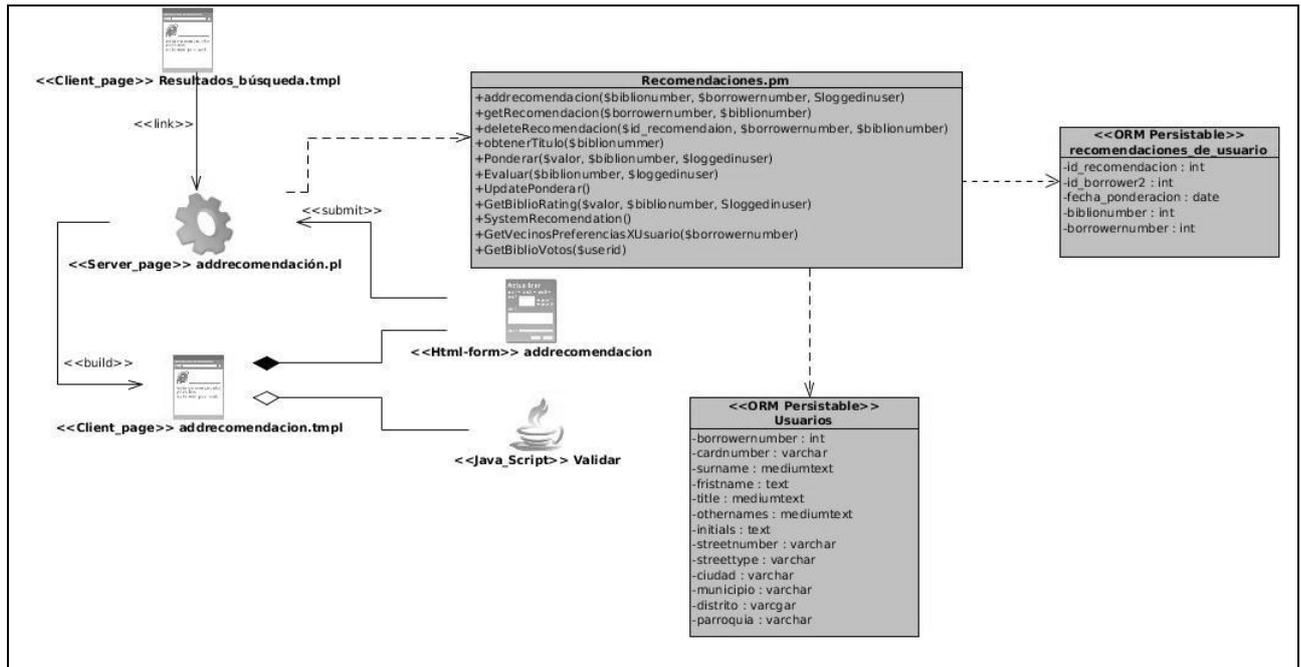


Ilustración 16: Diagrama de clases del diseño del CU Recomendar material a un usuario

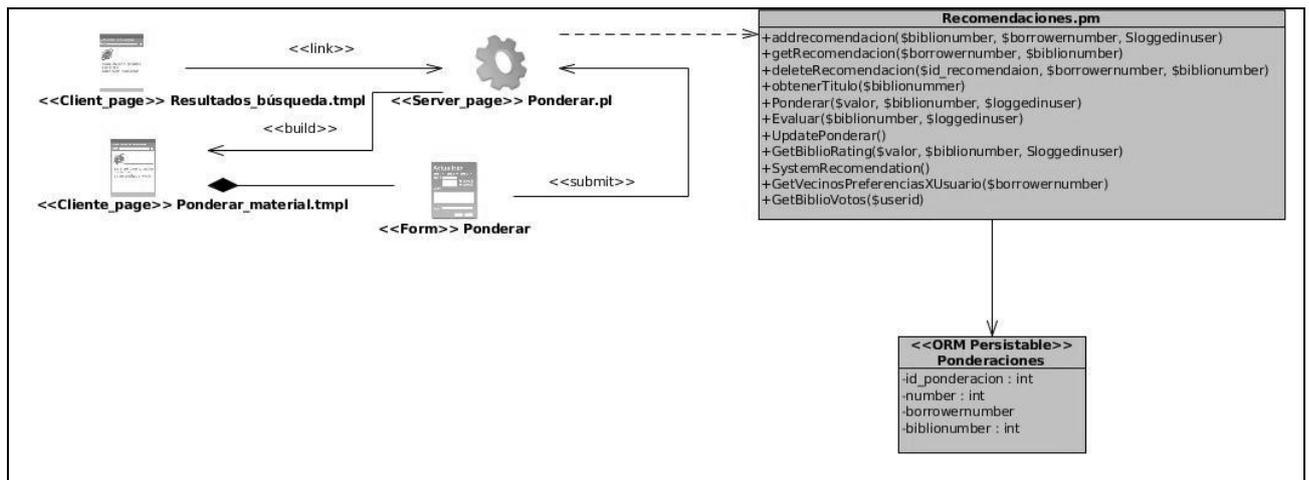


Ilustración 17: Diagrama de clases del diseño del CU Ponderar material

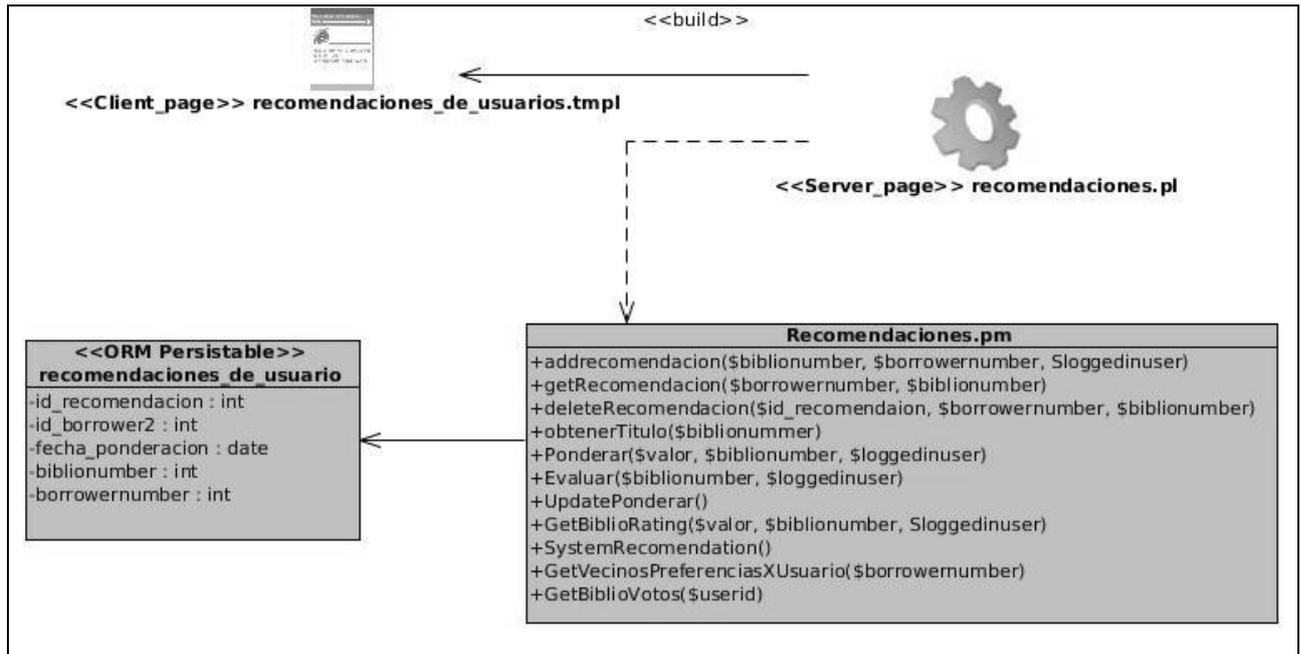


Ilustración 18: Diagrama de clases del diseño del CU Eliminar recomendación

Descripción de las clases del diseño

A continuación se describen las clases del diseño que se usan en el sistema compuestas por sus respectivos métodos y atributos.

Nombre: Sp_addRecomendaciones	
Tipo de clase: Controladora	
Atributo	Tipo
Para cada responsabilidad	
Nombre:	Recomendar(\$biblionumber, \$borrowernumber, Sloggedinuser)
Descripción:	Inserta una nueva recomendación

Tabla 8: Descripción de la CC gestionar-materiales

Nombre: Sp_Recomendaciones	
Tipo de clase: Controladora	
Atributo	Tipo
Para cada responsabilidad	

Nombre:	GetRecomendaciones(\$borrowernumber, \$biblionumber)
Descripción:	Muestra las recomendaciones realizadas por los usuarios.
Nombre	DeleteRecommendation(\$id_recomendaion, \$borrowernumber, \$biblionumber)
Descripción	Elimina la recomendación deseada por el usuario.
Nombre	GetVecinosPreferenciasXUsuario(\$borrowernumber)
Descripción	Obtiene la lista de los usuarios con las mismas preferencias del usuario actual.
Nombre	GetBiblioVotos(\$userid)
Descripción	Verifica si existe algún usuario con la misma preferencia del usuario actual.
Nombre	SystemRecommendation()
Descripción	Define las recomendaciones del sistema.

Tabla 9: Descripción de la CC gestionar-materiales

Nombre: Sp_Ponderar	
Tipo de clase: Controladora	
Atributo	Tipo
Para cada responsabilidad	
Nombre:	Ponderar(\$valor, \$biblionumber, \$loggedinuser)
Descripción:	Registra la ponderación de un material en la base de datos.
Nombre>	Evaluar(\$biblionumber, \$loggedinuser)
Descripción:	Verifica si el usuario ha evaluado el material seleccionado.
Nombre:	UpdatePonderacion(\$valor, \$biblionumber, Sloggedinuser)
Descripción:	Actualiza la evaluación de un material.
Nombre:	ObtenerTitulo(\$biblionummer)
Descripción:	Muestra el autor y título de un material.

Tabla 10: Descripción de la CC gestionar-materiales

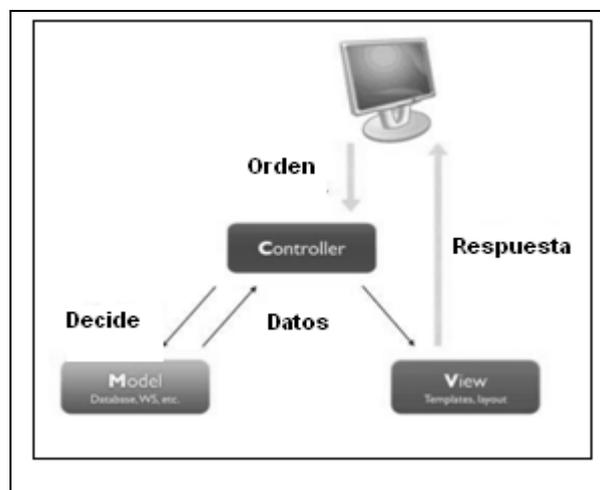
servicios (SOA), arquitectura basadas en objetos, arquitectura basada en n capas, modelo vista controlador (MVC), siendo este último el que se acuerda a utilizar debido a que separa los datos de una aplicación, la interfaz de usuario y la lógica de control en tres componentes.

Definición de las partes:

- **Modelo:** componente encargado del acceso a datos. Modela los datos y el comportamiento detrás de los procesos del negocio. A través de este componente se realizan las consultas a la base de datos, se ejecutan los cálculos de los procesos de negocio y se procesan las diferentes órdenes. Se encapsulan los datos y el comportamiento que son independientes de la presentación.
- **Vista:** en la vista se despliega la información de acuerdo al tipo de cliente, así como los resultados de la lógica del negocio (modelo). Una vista puede ser una página web o una parte de una página.
- **Controlador:** sirve como una conexión lógica entre la interacción de usuario y los servicios de negocio disponibles. En otras palabras, el controlador actúa como intermediario entre el modelo, la vista y cualquier otro recurso necesario para generar una página (21).

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 termina sus operaciones devuelve el flujo al controlador y este envía el resultado a la vista. Este proceso se puede ver más claro en la figura siguiente:



4.4 Tratamiento de errores

El tratamiento de errores se puede definir como uno de los pasos más importantes a tener en cuenta para lograr que el sistema funcione correctamente. Para el tratamiento de errores se utilizaron los mensajes de textos en la misma página donde se ejecuta la acción, explicando de esta forma en qué consiste el error y dando la oportunidad al usuario de corregirlo y seguir trabajando fácilmente. También se utilizaron técnicas de validación por parte del cliente para evitar el envío de información incorrecta por parte del usuario.

Conclusiones parciales

En este capítulo se detallaron las principales actividades desarrolladas en el flujo de análisis y diseño; uno de los más importantes dentro de la etapa de elaboración del software ya que brinda una visión general del mismo. Para ello se presentaron los requisitos que debe tener el sistema mediante los diagramas de análisis del sistema, diagramas de colaboración y clases del diseño. También quedó representado el modelo de datos a utilizar para el manejo de la información, el cual forma parte del diseño de la base de datos del sistema, además de realizar las descripciones de las tablas de la base de datos a utilizar por el sistema de recomendación.

Capítulo V: Implementación y prueba

Este capítulo expone lo referente al flujo de trabajo implementación y prueba, el cual es determinante en el proceso de desarrollo de software, para ello se modela el diagrama de componentes, haciendo una representación de la implementación de las clases del diseño en términos de componentes 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 puede tener el sistema de recomendación, corregirlos y obtener un óptimo funcionamiento.

5.1 Diagrama de despliegue

El diagrama de despliegue se utiliza para modelar la configuración de los elementos de procesamiento en tiempo de ejecución y de los componentes, procesos y objetos de software que viven en ellos. Se modelan los modelos físicos y las asociaciones que existen en ellos (28).

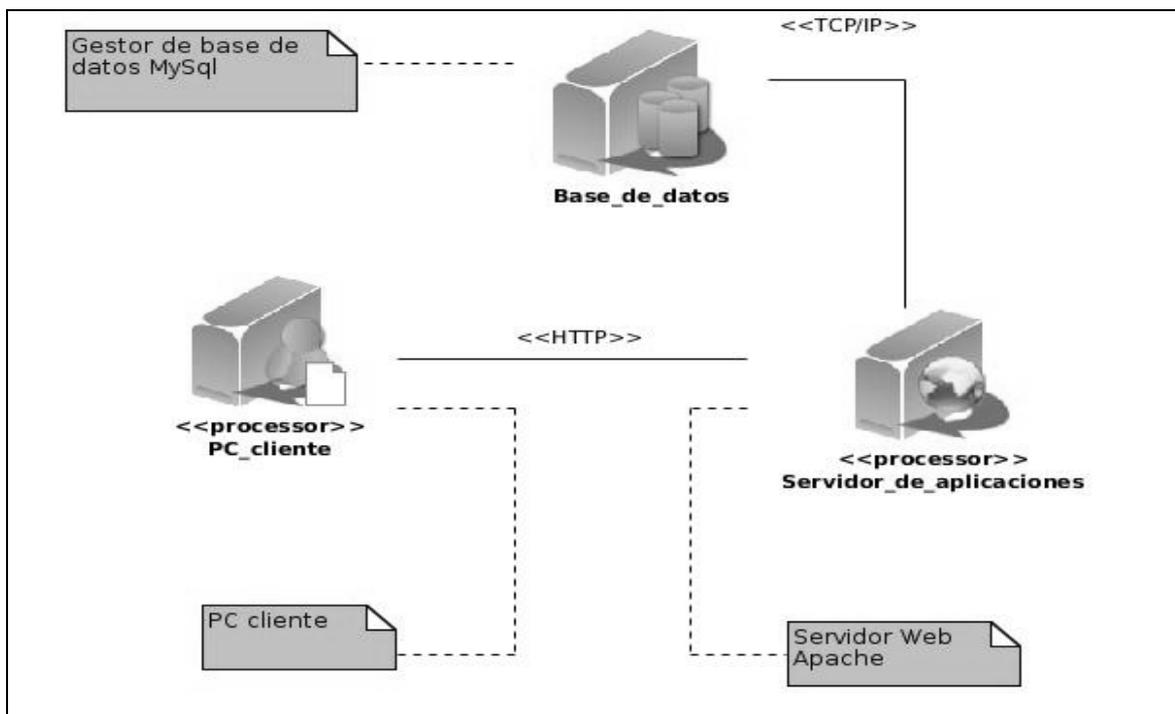


Ilustración 20: Diagrama de despliegue

- **Nodo pc_cliente:** representa las computadoras que utilizarán los usuarios para interactuar con el sistema. Establece comunicación con el servidor de aplicaciones con el protocolo HTTP.

- **Nodo servidor_de_aplicaciones:** en este nodo se encuentran los scripts de la aplicación.
- **Nodo servidor_de_base_de_datos:** en este nodo se encuentra el servidor de base de datos del SIGB Koha.

5.2 Diagrama de componentes

Los diagramas de componentes describen los elementos físicos del sistema y su relación, mostrando las dependencias lógicas entre componentes de software. Este diagrama hace parte de la vista física de un sistema, el cual modela la estructura de implementación de la aplicación por sí misma, su organización en componentes y su despliegue en nodos de ejecución. La vista de implementación se representa mediante los diagramas de componentes (28).

El sistema de recomendación está dividido en tres subsistemas de implementación fundamentales: el subsistema Vista, el subsistema Controladoras y el subsistema Modelos, estructurados de forma tal que se agrupan los scripts de acuerdo con el rol que desempeñan dentro del patrón arquitectónico MVC.

A continuación se muestra el diagrama de componentes de forma global.

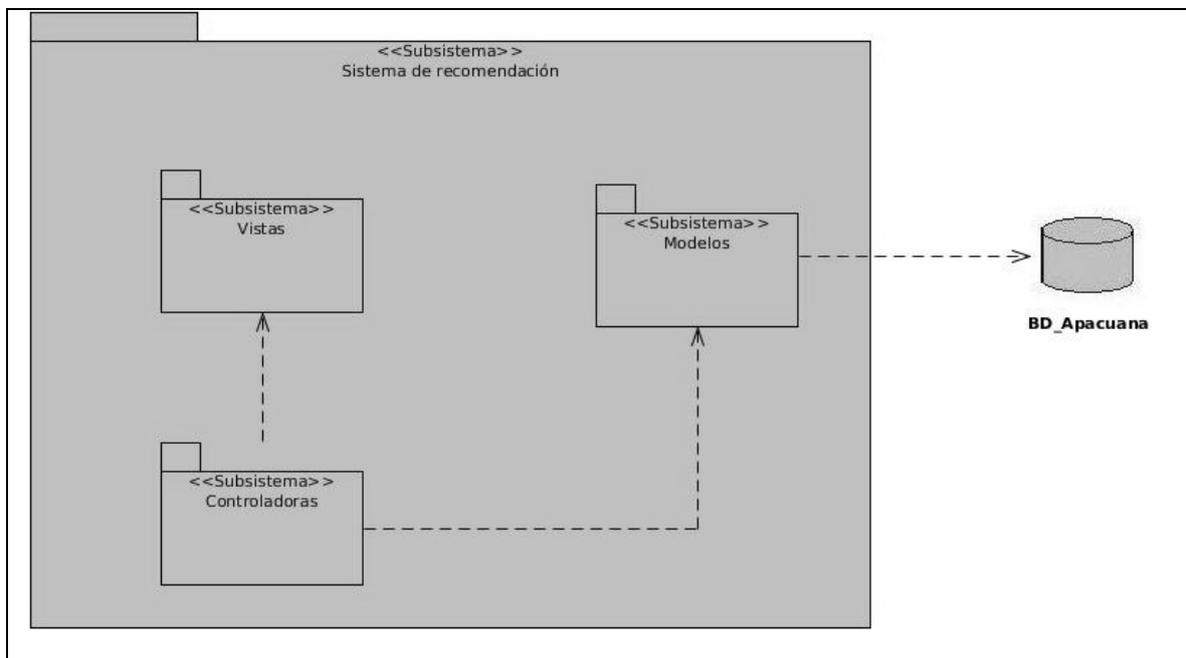


Ilustración 21: Diagrama de componentes global

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

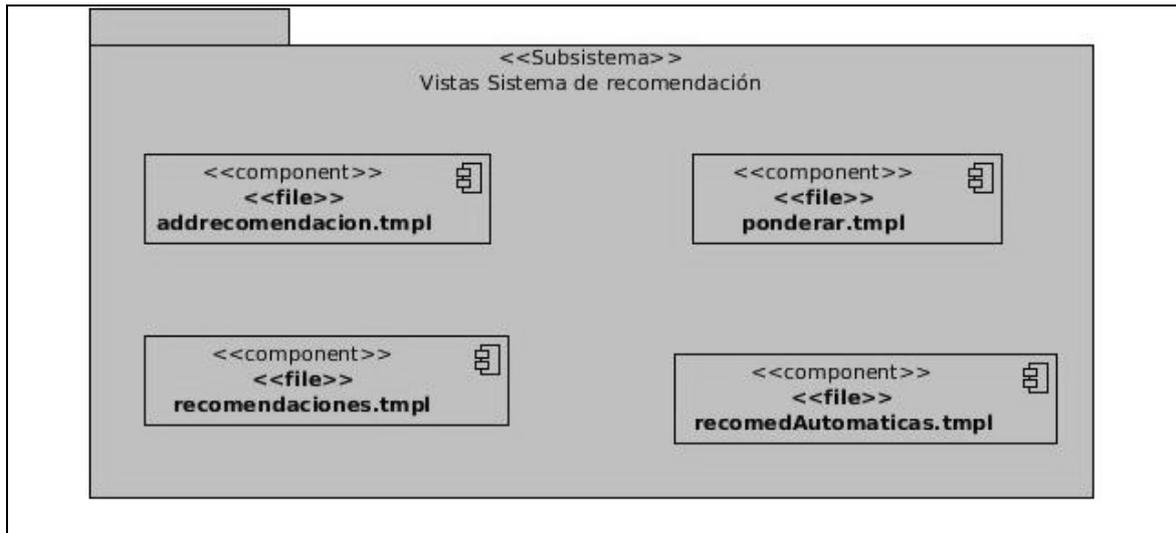


Ilustración 22: Subsistema Vistas

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

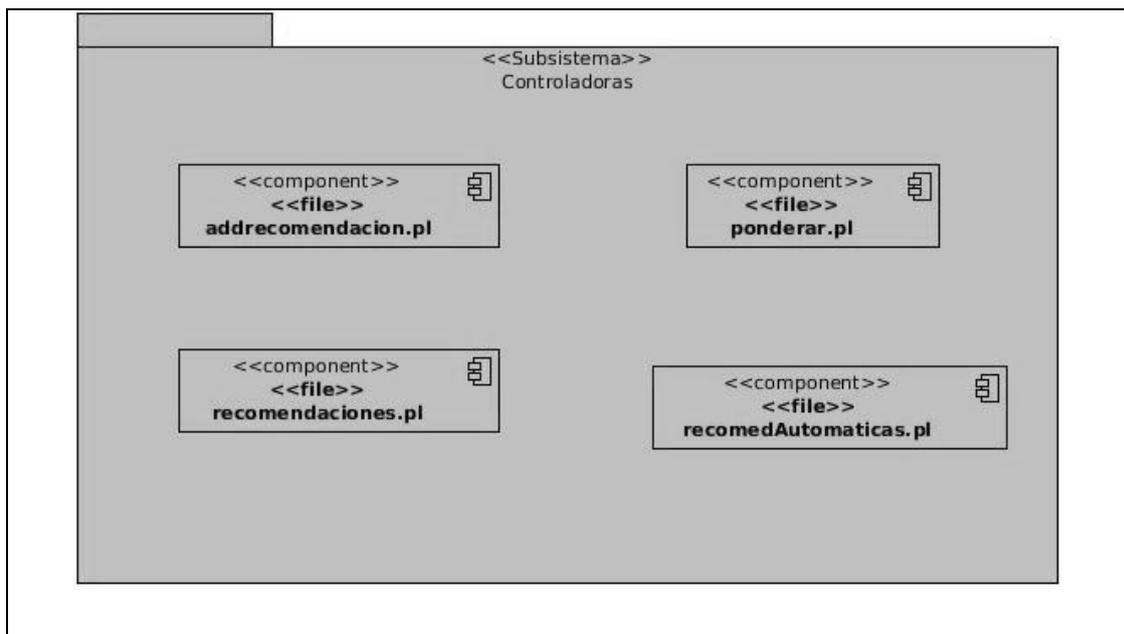


Ilustración 23: Subsistema Controladoras

El subsistema Modelos es el encargado de la interacción con la base de datos, para de esta forma gestionar la información con la que trabaja el sistema de recomendación.

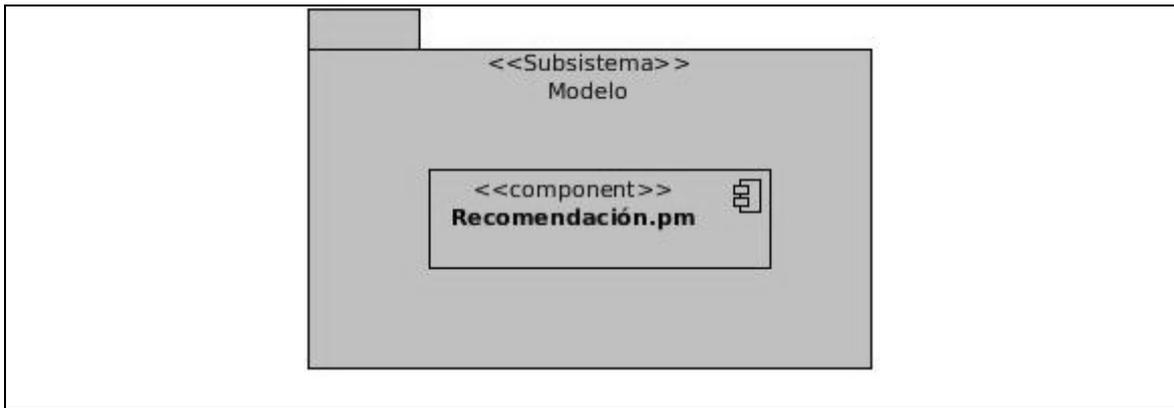


Ilustración 24: Subsistema Controladoras

5.3 Modelos de prueba

Las pruebas son actividades en las cuales un sistema o componente es ejecutado bajo condiciones o requerimientos especificados, los resultados son observados y registrados y una evaluación es hecha de algún aspecto del sistema o componente. La prueba de software es un elemento crítico para la garantía de la calidad del software y representa una visión final de las especificaciones del diseño y de la codificación.

Existen dos métodos de pruebas fundamentales: el método de caja negra y el de caja blanca. El método de caja negra consiste en pruebas que se realizan sobre la interfaz del software, por lo que los casos de prueba pretenden demostrar que las funciones del software sean operativas. Esta prueba examina algunos aspectos del modelo fundamentalmente del sistema sin tener en cuenta la estructura interna del software. Además, permite obtener un conjunto de condiciones de entrada que ejerciten completamente todos los requisitos funcionales de un programa.

Por otro lado, en las pruebas de caja blanca se comprueban los caminos lógicos del software proponiendo casos de prueba donde se ejerciten conjuntos específicos de condiciones y/o bucles. Además, se puede examinar el estado del programa en varios puntos para determinar si el estado real coincide con lo esperado o mencionado.

En este caso se utiliza el método de caja negra aplicando la técnica de partición de equivalencia, que permite examinar los valores válidos o 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 genético.

A continuación se describen los casos de pruebas desarrollados para cada caso de uso definido, especificando la información de entrada, los resultados obtenidos una vez ejecutado el caso de prueba y las condiciones que debe cumplirse mientras este se ejecuta.

IMPLEMENTACIÓN Y PREUBA

Nombre del caso de uso	Recomendar material a un usuario	
Entrada	Resultados	Condiciones
El usuario accede a realizar la búsqueda y no especifica el criterio de la misma.	El sistema muestra un mensaje: Campo requerido.	La operación se repite hasta que el usuario especifique el criterio por el cual desea realizar la búsqueda.
El usuario accede a realizar la búsqueda de un usuario e introduce los datos incorrectamente.	El sistema muestra un mensaje: Nombre incorrecto.	La operación se repite hasta que el usuario introduzca los datos correctamente.
El usuario accede a realizar la búsqueda de un usuario e introduce los datos correctamente pero que no se encuentran registrados en la base de datos.	El sistema muestra un mensaje: No se encontraron resultados	La operación se repite hasta que el usuario corrija el error.
El usuario accede a realizar la búsqueda de un usuario e introduce los datos correctamente.	El sistema muestra el formulario con la lista de resultados que coinciden con el criterio de búsqueda.	Seleccionar el o los usuarios a recomendar y oprimir el botón recomendar
El usuario no especifica el usuario a recomendar.	El sistema muestra un mensaje: Debes de seleccionar por lo menos un usuario.	La operación se repite hasta que el usuario seleccione un usuario.
El usuario selecciona un usuario que ya esté recomendado con el material seleccionado.	El sistema muestra un mensaje: Ya existe la recomendación.	La operación se repite hasta que el usuario corrija el error.
El usuario selecciona un usuario.	El sistema registra la recomendación.	No exista la recomendación.

Tabla 12: Modelo de prueba del CU_ Recomendar material a un usuario

Nombre del caso de uso	Ponderar material	
Entradas	Resultados	Condiciones

IMPLEMENTACIÓN Y PREUBA

El usuario selecciona la evaluación.	El sistema modifica la evaluación del material	El usuario haya evaluado el material anteriormente.
El usuario selecciona la evaluación.	El sistema registra la evaluación del material.	El usuario no haya evaluado el material anteriormente.

Tabla 13: Modelo de prueba del CU_Ponderar material

Nombre del caso de uso	Visualizar recomendaciones de usuarios.	
Entradas	Resultados	Condiciones
El usuario accede visualizar las recomendaciones de usuarios	El sistema muestra un mensaje: No tienes recomendaciones por los usuarios.	El usuario no tenga recomendaciones.
El usuario accede visualizar las recomendaciones de usuarios	El sistema muestra un listado con todas las recomendaciones.	El usuario tenga recomendaciones.

Tabla 14: Modelo de prueba del CU_Visualizar recomendaciones de usuarios

Nombre del caso de uso	Visualizar recomendaciones automáticas.	
Entradas	Resultados	Condiciones
El usuario accede visualizar las recomendaciones automáticas.	El sistema muestra un mensaje: No tienes recomendaciones por el sistema.	El usuario no tenga recomendaciones.
El usuario accede visualizar las recomendaciones automáticas.	El sistema muestra un listado con todas las recomendaciones.	El usuario tenga recomendaciones.

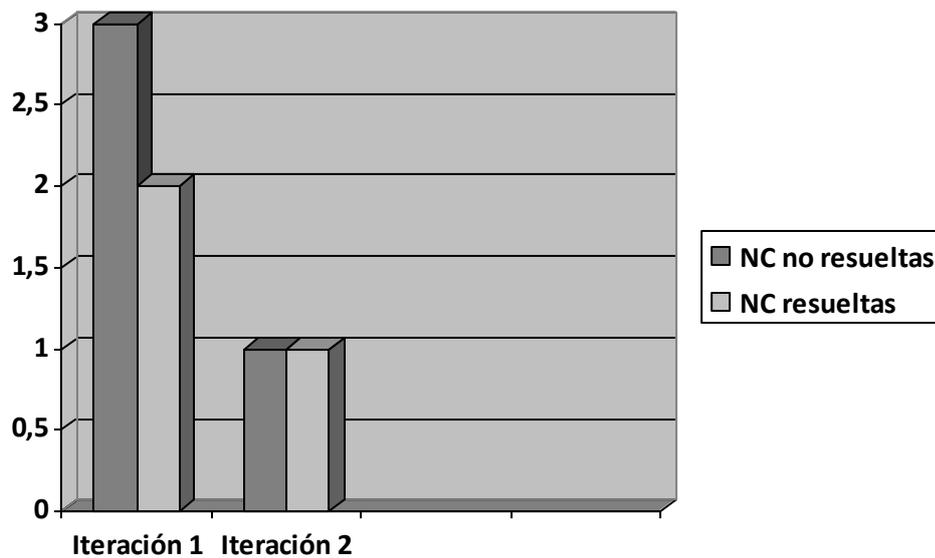
Tabla 15: Modelo de prueba del CU_Visualizar recomendaciones automáticas

Nombre del caso de uso	Eliminar recomendaciones	
Entradas	Resultados	Condiciones
El usuario accede a eliminar	El sistema elimina la	Tener recomendaciones.

algunas de sus recomendaciones	recomendación seleccionada.	
--------------------------------	-----------------------------	--

Tabla 16: Modelo de prueba del CU_ Eliminar recomendación

Para corregir los errores cometidos durante el diseño e implementación del sistema se realizaron dos iteraciones de pruebas, en las cuales se detectaron un conjunto no conformidades. De un total de 6 requisitos se encontraron en la primera iteración 3 no conformidades pudiendo resolverse solo 2 de ellas, en la segunda iteración se detectó 1 nueva no conformidad además de la primera pendiente de la iteración anterior, en esta oportunidad se le dio respuesta a la no conformidad de la primera iteración y a la no conformidad de la segunda iteración, quedando el sistema listo para su uso.



Conclusiones parciales

En este capítulo se abordaron los aspectos fundamentales del flujo de trabajo implementación y prueba, donde se presentó el diagrama de despliegue, especificando la distribución física por nodos del sistema y sus características, además de representar el diagrama de componentes representando la relación entre los mismos. Por su parte en el flujo de pruebas se realizaron las descripciones de los casos de prueba pertenecientes a los casos de usos documentados en el trabajo.

Conclusiones generales

La concepción y desarrollo de un sistema de recomendación para el catálogo en línea de la biblioteca de la UCI posibilitó arribar las siguientes conclusiones:

- La realización de un estudio del estado del arte de los sistemas de recomendación permitió arrojar que los mismos constituyen una herramienta de gran importancia, ya que proporcionan a los usuarios consejos e información personalizada sobre productos y/o servicios que pueden ser de interés a la hora de tomar decisiones.
- Se realizó un estudio de los algoritmos de filtrado colaborativo, los cuales permiten realizar cálculos para obtener las predicciones de las recomendaciones, eligiendo de ellos el algoritmo a incluir en la propuesta de solución.
- Se implementó un sistema de recomendación para el catálogo en línea de la biblioteca de la UCI que recomienda materiales de interés para los usuarios de la misma, teniendo en cuenta los artefactos generados durante la fase de análisis y diseño, dándole solución al problema planteado.
- Tras obtener la herramienta se realizaron varias pruebas del algoritmo y del sistema de recomendación en sí, obteniéndose índices aceptables en cada una de ellas, por lo que se concluye un resultado general positivo.

Recomendaciones

Se propone que:

- Se modifique el diseño del sistema para que las recomendaciones automáticas aparezcan en un lugar más visible.
- Modificar la funcionalidad visualizar recomendaciones automáticas, para que se tengan en cuenta aquellos contenidos que han sido más votados.
- Agregar una nueva funcionalidad que permita a los usuarios conocer cuáles son sus vecinos.

Referencias bibliográficas

1. **SEGUIDO FONT, MIGUEL.** Sistemas de recomendación para webs de información sobre la salud. 2009. *Sitio Web UPCommons*. [En línea] [Citado el: 5 de 10 de 2011.] Disponible en: <http://upcommons.upc.edu/pfc/handle/2099.1/7193>.
2. **BERTATE, LETICIA., MACHADO, ROFRIGO., VALERIA.** PGMúsica Sistema de Recomendación de música. 2006. *Sitio web Inco*. [En línea] Última actualización: 2006/4/18. [Citado el: 18 de 3 de 2012.] Disponible en: www.fing.edu.uy/inco/grupos/pln/prygrado/InformePGMusica.pdf.
3. **J.PZZANI, MICHAEL., BILLSUS, DANIEL.** Content-Based Recommendation Systems. 2007. *Sitio web Springerlink*. [En línea] [Citado el: 25 de 4 de 2012.] Disponible en: <http://www.springerlink.com/content/gg35wt68l6774261>.
4. **SCHAFER, J.BEN., FRANKOWSKI, DAN., HERLOCKER, JON., SEND, CHILAD.** Collaborative Filtering Recommender Systems. 2007. *Sitio web Information Search and Retrieval*. [En línea] [Citado el: 25 de 4 de 2012.] Disponible en: <http://www.inf.unibz.it/~ricci/ISR>.
5. **BURQUE, ROBIN.** Knowledge-based recommender systems. [En línea] [Citado el: 25 de 4 de 2012.] Disponible en: www.cs.odu.edu/~mukka/cs795sum10dm/Lecturenotes/Day6/burke-elis00.pdf.
6. **J.PAZZANI, MICHAEL.** Content-Based and Demographic Filtering. 2000. [En línea] [Citado el: 15 de 3 de 2012.] Disponible en: http://www.cs.northwestern.edu/~pardo/courses/mmml/papers/collaborative_filtering_a_framework_for_content_based_demographic_filtering_AIR99.pdf.
7. **D.BURKE, ROBIN., J.HAMMOND, KRISTIAN., C.YOUNG, BENJAMIN.** Knowledge-Based Navigation of Complex Information Spaces. 1996. [En línea] [Citado el: 15 de 3 de 2012.] Disponible en: www.cs.pitt.edu/~mrotaru/comp/rs/Burke%20AAAI%201996.pdf.
8. **PLAZA, ERIC., MCGINTY, LORRAINE.** Distributed case-based reasoning. 2005. [En línea] [Citado el: 20 de 3 de 2012.] Disponible en: http://www.iiia.csic.es/People/enric/papers/Distributed_CBR.pdf.
9. **A.THOMPSON, CYNTHIA., HÖKER, MHEMET., LANGLEY, PAT.** A Personalized System for Conversational Recommendations. 2004. [En línea] [Citado el: 24 de 3 de 2012.] Disponible en: www.jair.org/media/1318/live-1318-2250-jair.pdf.

10. **D.BURKE, ROBIN., J.HAMMOND, KRISTIAN., C.YOUNG, BENJAMIN.** The FindMe approach to assisted browsing. 1997. [En línea] [Citado el: 2 de 5 de 2012.] Disponible en:
<http://www.cs.pitt.edu/~mrotaru/comp/rs/Burke%20IEEE%201997.pdf>.
11. **CODINA BUSQUET, VICTOR.** Sistema de recomendación personalizada de contenido vídeo. 2008. *Sitio web: UPCommons1*. [En línea] [Citado el: 16 de 2 de 2012.] Disponible en: <https://upcommons.upc.edu/pfc/handle/2099.1/5125>.
12. **RUIZ INIESTA, ALMENUDA.** Estrategias de recomendación aplicadas a repositorios de recursos educativos. 2009. *Sitio web: E-Prints Complutense*. [En línea] [Citado el: 20 de 1 de 2012.] Disponible en: <http://eprints.ucm.es/9908>
13. **N.MILLER, BRADLEY., ISTVAN, ALBERT., K.LAM, SHYONG., A, JOSEPH., REIDEL, JHON.** MovieLens Unplugged: Experiences with a Recome. [En línea] [Citado el: 25 de 4 de 2012.] Disponible en:
<http://knuth.luther.edu/~bmiller/Papers/hci03.pdf>
14. **SARWAR, BADRUL., KARYPIS, GEORGE., KONSTAN, JOSEPH., REID, JHON.** Item-Based Collaborative Filtering Recommendation. [En línea] [Citado el: 10 de 12 de 2011.] Disponible en: <http://www.www10.org/cdrom/papers/pdf/p519.pdf>
15. **BURKE, ROBIN.** Hybrid Web Recommender Systems. 2007. [En línea] [Citado el: 21 de 11 de 2011.] Disponible en:
www.dcs.warwick.ac.uk/~acristea/courses/CS411/2010/Book%20-%20The%20Adaptive%20Web/HybridWebRecommenderSystems.pdf
16. **TOLEDO YERA, RACIEL.** *Concepción y desarrollo de un sistema de recomendación para jurados online de programación*. . s.l. : Universidad de las Ciencias Informáticas, 2011.
17. **BEN SCHAFER, J., KONSTAN, J.** Recommender Systems in E-Commerce. 1999. *Sitio web GroupLens Search*. [En línea] [Citado el: 8 de 3 de 2012.] Disponible en:
<http://www.grouplens.org/papers/pdf/ec-99.pdf>
18. **VOLLRATH, IVO., W.OLFGANG, W ILKE., R.ALPH, B ERGMANN.** Case-based reasoning support for online catalog sales. [En línea] [Citado el: 10 de 4 de 2012.] Disponible en: http://www.wi2.uni-trier.de/publications/1998_w4voll.pdf
19. **GARCÍA SALCINES, ENRIQUE., ROMERO, CRISTOBAL., VENTURA, SEBASTIÁN., LOZANO, CARLOS DE CASTRO.** Sistema recomendador

- colaborativo usando minería de datos distribuida para la mejora continua de cursos e-learning. [En línea] [Citado el: 8 de 3 de 2012.] Disponible en: <http://arnetminer.org/publication/sistema-recomendador-colaborativo-usando-miner-a-de-datos-distribuida-para-la-mejora-continua-de-cursos-e-learning-1183593.html>
20. *Base de datos EBSCO*. [En línea] [Citado el: 25 de 5 de 2012.] Disponible en: www.bvsayuda.sld.cu/ayudas/tutorial/base-de-datos-ebSCO/bfque-es-ebSCO.
21. **ORTEGA RETURETA, LAURA SUSANA., ROSELLÓ CARRAZANA, ADNIER.** *Desarrollo del módulo Adquisición del Sistema Integrado de Gestión Bibliotecaria Koha para la Biblioteca Nacional José Martí*. Universidad de las Ciencias Informáticas. : s.n., 2010.
22. *Sitio web mpr.es Descargar MySQL gratis*. [En línea] [Citado el: 20 de 5 de 2012.] Disponible en: <http://descargar.mp3.es/lv/group/view/kl51464/MySQL.htm>.
23. *Introducción al lenguaje html. HOOPIN.NET*. [En línea] [Citado el: 25 de 1 de 2012.] Disponible en: <http://www.hooping.net/faq-html.aspx>.
24. **CABRERA, H. C.** *sitio web CSS Habana*. [En línea] Última actualización: 11 de 10 de 2011. [Citado el: 20 de 1 de 2012.] Disponible en: <http://www.ecured.cu/index.php/CSS>.
25. **VALDÉS NUALLA, YESLY CECILIA., CANDELARIO RODRÍGUEZ, LIZANDRA.** *Sistema de búsqueda de términos y definiciones*. Universidad de las Ciencias Informáticas. : s.n., 2010.
26. **GALÁN NIETO, SERGIO MANUEL.** *Filtrado Colaborativo y Sistemas de Recomendación*. 2007. [En línea] Disponible en: <http://www.it.uc3m.es/jvillena/irc/practicas/06-07/31.pdf>.
27. **BIELZA, CONCHA., LARRANAGA, PEDRO.** *B.3 Vecinos más cercanos*. [En línea] [Citado el: 8 de 3 de 2012.] Disponible en: <http://www.dia.fi.upm.es/~concha/B.3-knn.pdf>.
28. **LAURENCIO PÉREZ, DANIUSKA., DIÉGUEZ GARCÍA, YUNIESKI.** *Sistema de Gestión de la Trayectoria*. Universidad de las Ciencias Informáticas. : s.n., 2009.

Clustering: es una técnica estadística que permite una generación automática de grupos en los datos. Incluso, existen algoritmos de clustering que permiten la generación de grupos jerárquicos, consiguiendo una mayor abstracción y representación de la información para poder recuperarla más eficiente.

EBSCOhost: es una base de datos de información científica sobre medicina, física, química, economía, educación y otros campos. Es propiedad de la compañía *EBSCO Publishing*.

e-commerce (comercio electrónico): consiste en la compra y venta de productos o de servicios a través de medios electrónicos, tales como Internet y otras redes informáticas.

e-learning (aprendizaje electrónico): a la educación a distancia completamente virtualizada a través de los nuevos canales electrónicos (las nuevas redes de comunicación, en especial Internet), utilizando para ello herramientas o aplicaciones de hipertexto (correo electrónico, páginas web, foros de discusión, mensajería instantánea, plataformas de formación, entre otros) como soporte de los procesos de enseñanza-aprendizaje.

Feedback (retroalimentación): el proceso de compartir observaciones, preocupaciones y sugerencias con otra persona.

Karl Pearson: (Londres 27 de marzo de 1857- Londres, 27 de abril de 1936) fue un prominente científico, matemático y pensador británico, que estableció la disciplina de la estadística matemática. Desarrolló una intensa investigación sobre la aplicación de los métodos estadísticos en la biología y fue el fundador de la bioestadística. Escribió el libro "*Mathematical Contributions to the Theory of Evolution*" (1893 – 1912). Consta de 18 volúmenes y en ellos se encuentran todas sus principales aportaciones a la estadística: las contribuciones al análisis de regresión, el coeficiente de correlación e incluye la prueba de chi-cuadrado de significación estadística.

Anexo 1

Caso de uso recomendar material a un usuario

Caso de uso	Recomendar material a un usuario.	
Actores	Usuario	
Resumen	El caso de uso inicia cuando el usuario desea recomendar un contenido después de haber realizado una búsqueda y finaliza cuando termina de ejecutar la acción.	
Pre-condiciones	Al menos en el sistema debe estar registrado un documento.	
Referencias	RF-1	
Complejidad	Baja	
Prioridad	Media	
Pos-condiciones		
Flujo normal de eventos		
Acción del autor	Respuesta del sistema	
1. Selecciona vínculo "Recomendar".	2. Muestra una ventana para buscar el usuario a recomendar.	
3. Introduce los datos: Nombre y apellidos/Número carné de identidad. Presiona el botón "Buscar".	4. Validar la entrada de datos.	
	5. Realiza la búsqueda.	
	6. Verifica que el usuario esté registrado en la base de datos.	
	7. Muestra un listado con los nombres relacionados con el criterio de búsqueda.	
8. Selecciona al menos un usuario y escoge una de las siguientes opciones: "Recomendar" "Cancelar" "Cerrar ventana"		

8.1 Selecciona la opción "Recomendar"	9. Verifica que el usuario haya seleccionado por lo menos un usuario.
	10. Verifica que el usuario no esté recomendado con el material.
	11. Registra la recomendación.
	12. Muestra un mensaje "Registro satisfactorio". Termina el caso de uso.
Flujo Alternativo 1	
Acción del autor	Respuesta del sistema
	4.1 Muestra un mensaje "Campo requerido".
	6.1 Muestra un mensaje "No se encontraron resultados".
	9.1 Muestra un mensaje "Debes de seleccionar por lo menos un usuario".
	10.1 Muestra un mensaje "Ya existe la recomendación."
Flujo Alternativo 2	
Acción del autor	Respuesta del sistema
8.2 Selecciona la opción "Cancelar"	9.1 Cancela la operación y muestra la interfaz anterior.
Flujo Alternativo 3	
Acción del autor	Respuesta del sistema.
8.3 Selecciona la opción "Cerrar ventana"	9.1 Cierra la ventana y muestra la interfaz anterior.

Tabla 8: Descripción textual del caso de uso recomendar material a un usuario

Caso de uso visualizar recomendaciones automáticas

Caso de uso	Visualizar recomendaciones automáticas.	
Actores	Usuario	
Resumen	El caso de uso comienza cuando el usuario desea ver las recomendaciones realizadas por el propio sistema, terminando el caso de uso una vez mostrado el listado de las mismas.	
Pre-condiciones	Usuario autenticado	
Referencias	RF-2	
Complejidad	Alta	
Prioridad	Alta	
Pos-condiciones		
Flujo normal de eventos		
Acción del autor	Respuesta del sistema	
1. Escoge el vínculo “Mis recomendaciones.”	2. Muestra una interfaz para escoger entre las recomendaciones de usuarios y las automáticas.	
3. Escoge la opción “Recomendaciones automáticas”.	4. Verifica que el usuario tenga recomendaciones.	
	5. Muestra un listado con todas las recomendaciones realizadas por el sistema. Termina el caso de uso.	
Flujo Alterno 1		
Acción del autor	Respuesta del sistema	
	5.1 Muestra un mensaje “No se encontraron registros bibliográficos”.	

Tabla 9: Descripción textual del caso de uso visualizar recomendaciones automáticas

Caso de uso visualizar recomendaciones de usuarios

Caso de uso	Visualizar recomendaciones de usuario.	
Actores	Usuario	
Resumen	El caso de uso inicia cuando el usuario desea ver las recomendaciones realizadas por algún usuario, terminando el caso de uso mostrando un listado de las mismas.	
Pre-condiciones	Usuario autenticado	
Referencias	RF-3	
Complejidad	Baja	
Prioridad	Baja	
Pos-condiciones		
Flujo normal de eventos		
Acción del autor	Respuesta del sistema	
1. Escoge el vínculo "Mis recomendaciones."	2. Muestra una interfaz para escoger entre las recomendaciones de usuarios y las automáticas.	
3. Escoge la opción "Recomendaciones de usuarios".	3. Escoge la opción "Recomendaciones de usuarios".	
	5. Muestra un listado con todas las recomendaciones realizadas por otros usuarios. Termina el caso de uso.	
Flujo Alterno 1		
Acción del autor	Respuesta del sistema	
	5.1 Muestra un mensaje "No se encontraron registros bibliográficos".	

Tabla 10: Descripción textual del caso de Visualizar recomendaciones de usuario

Caso de uso ponderar material

Caso de uso	Ponderar material.	
Actores	Usuario	
Resumen	El caso de uso inicia cuando el usuario selecciona la opción Evaluar, terminando el caso de uso una vez realizada la acción.	
Pre-condiciones	El usuario debe realizar una búsqueda satisfactoria.	
Referencias	RF-4, RF-5	
Complejidad	Media	
Prioridad	Media	
Pos-condiciones	Se visualiza el promedio con las ponderaciones hechas hasta el momento.	
Flujo normal de eventos		
Acción del autor	Respuesta del sistema	
1. Escoge la opción "Ponderar".	2. Muestra una ventana para ponderar el contenido.	
3. Evalúa el material y selecciona una de las siguientes opciones: "Ponderar" "Cancelar" "Cerrar ventana"	.	
3.1 Selecciona la opción "Ponderar"	4. Registra la evaluación del material, actualiza y muestra el promedio de las evaluaciones. Termina el caso de uso.	
Flujo Alternativo 1		
Acción del autor	Respuesta del sistema	
3.2 Selecciona la opción "Cancelar"	4.1 Cancela la operación y muestra la interfaz anterior.	
3.3 Selecciona la opción "Cerrar ventana"	4.1 Cierra la ventana y muestra la interfaz anterior.	

Tabla 11: Descripción textual del caso ponderar material

Caso de uso eliminar recomendación

Caso de uso	Eliminar recomendación.	
Actores	Usuario	
Resumen	El caso de uso inicia cuando el usuario desea eliminar alguna de las recomendaciones hechas por otros usuarios, terminando el caso de uso cuando se elimina la recomendación.	
Pre-condiciones	Usuario autenticado y tener recomendaciones	
Referencias	RF-6	
Complejidad	Baja	
Prioridad	Baja	
Pos-condiciones		
Acción del autor	Respuesta del sistema	
1. Selecciona el vínculo "Mis recomendaciones".	2. Muestra una interfaz con las recomendaciones.	
3. Selecciona la opción "Eliminar"	3. Elimina la recomendación de la lista. Termina el caso de uso	

Tabla 12: Descripción textual del caso eliminar recomendación

Anexo 2

Interfaz visualizar recomendaciones de usuarios

Recomendaciones

De usuarios Automáticas

Ejemplares recomendados por los usuarios (2 total)

Tipo de ejemplar	Título	Autor	Eliminar
	Declaraciones de autorías : facultad 10 : curso 2008-2009	Declaraciones de autorías : facultad 10 : curso 2008-2009	ELIMINAR
 Trabajo de Diploma	Desarrollo del módulo de gestión de tesis del sistema integral de gestión administrativa de la facultad 7	Guedes Hernández, Alexander	ELIMINAR

Interfaz visualizar recomendaciones automáticas

Recomendaciones

De usuarios Automáticas

Ejemplares recomendados por el sistema (6 total)

Tipo de ejemplar	Título	Autor	Signatura topográfica
 Libros	Crónicas del mañana : 50 años de cuentos cubanos de ciencia ficción	Sánchez, José Miguel	C863-San-C
 Libros	La Tesis y el Trabajo de Tesis : recomendaciones metodológicas para la elaboración de los trabajos de Tesis.	García Córdoba, Fernando	808.066378-Gar-T
 Libros	Nueva Autodidáctica. Enciclopedia Temática Universal. Tomos 1-8	Gispert, Carlos	Ref.030.Nue.N
 Trabajo de Diploma	Desarrollo del módulo de gestión de tesis del sistema integral de gestión administrativa de la facultad 7	Guedes Hernández, Alexander	005.12-Gue-D-TD-03087-10
 Trabajo de Diploma	Sistema de Gestión de Tesis Facultad 2	Sosa Veranes, Maité	005.12-Sos-S-TD-2697-09

Interfaz recomendar material a un usuario

Realizar búsqueda del usuario:

Seleccionar

BUSCAR

Resultados de la búsqueda:

	Ficha	Nombre y Apellidos	Catálogo	Biblioteca
<input type="checkbox"/>	E11758	Zamora Cese, Yanisel	Usuario (A)	UCI

RECOMENDAR **CANCELAR**

Cerrar esta ventana.

Interfaz evaluar material

Evalúe este documento.

Resultado

Desarrollo del módulo de gestión de tesis del sistema integral de gestión administrativa de la facultad 7 por Guedes Hernández, Alexander

Fecha:2011-12-07

Ponderar: 1 2 3 4 5

PONDERAR **CANCELAR**

Cerrar esta ventana.