****

**UNIVERSIDAD TECNOLÓGICA METROPOLITANA**

**ESCUELA DE INFORMÁTICA**

"Sistemas de Recomendaciones orientado al sector Turismo"

Trabajo de Titulación para optar al

Título de Ingeniero en Informática.

|  |  |  |
| --- | --- | --- |
| Profesor Guía: | |  | | --- | | Santiago Zapata Cáceres | |
| Alumno: | Marco Palacios Fuentes |

Santiago-Chile  
2015

Nota Obtenida:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Firma y Timbre Autoridad Responsable

Resumen

Abstract

Los sistemas de recomendaciones son mecanismos de filtrado de información que permiten a empresas generar de forma automatizada recomendaciones apuntadas a usuarios específicos o a grupos de usuarios específicos, basados en la recopilación de gustos y preferencia de estos.

Con el aumento del uso de internet, específicamente con el aumento de la compra on-line, los sistemas de recomendaciones se han vuelto una herramienta de software fundamental para páginas de venta on-line, ya que los sistemas de recomendaciones ayudan al usuario en el proceso de toma de decisiones, como lo son "¿que escuchar?" o "¿que comprar?".

En este informe se presentara el desarrollo de una aplicación web con un sistema de recomendaciones para un usuario en el área de turismo.

Índice

[Introducción 1](#_Toc458703222)

[Capítulo 1: Antecedentes Generales 2](#_Toc458703223)

[1.1 Motivación 2](#_Toc458703224)

[1.2 Objetivo General 3](#_Toc458703225)

[1.3 Objetivos Específicos 3](#_Toc458703226)

[1.4 Alcances y Limitaciones 3](#_Toc458703227)

[Capítulo 2: Marco Teórico 5](#_Toc458703228)

[2.1 Contexto: Sistema de Recomendación [1][2][5] 5](#_Toc458703229)

[2.2 Definición: Sistemas de Recomendación [7][5] 6](#_Toc458703230)

[2.3 Técnicas Actuales [1][2][4][5][6][7][10][11] 7](#_Toc458703231)

[2.4 Características de los Sistemas de Recomendación [9][5] 8](#_Toc458703232)

[2.4.1 Ventajas de los Sistemas de Recomendación según su tipo [11] [9][5] 8](#_Toc458703233)

[2.4.2 Desventajas de los Sistemas de Recomendación según su tipo [5][9][11] 10](#_Toc458703234)

[2.5 Sistemas de Recomendación Híbridas [1][5][12] 12](#_Toc458703235)

[2.6 Mecanismos de Retroalimentación [1][5][9] 13](#_Toc458703236)

[2.6.1 Retroalimentación Explicita [1][5][9] 14](#_Toc458703237)

[2.6.2 Retroalimentación Implícita [1][5][9] 14](#_Toc458703238)

[Capítulo 3: 16](#_Toc458703239)

[3.1 Metodología de desarrollo [3] [13] 16](#_Toc458703240)

[3.2 Metodología de Desarrollo Ágil [3] [13] 18](#_Toc458703241)

[3.3 Ventajas y desventajas de las Metodologías de Desarrollo Ágil [3] [13] 19](#_Toc458703242)

[Capitulo 4 : 22](#_Toc458703243)

[4.1 Diseño Conceptual del Proyecto 22](#_Toc458703244)

[4.2 definición de los requerimientos 22](#_Toc458703245)

[4.2.1 requerimientos funcionales 23](#_Toc458703246)

[4.2.2 requerimientos no funcionales 23](#_Toc458703247)

[4.2.3 requerimientos de hardware 24](#_Toc458703248)

[El sistema será desarrollado en un notebook con las siguientes características: 24](#_Toc458703249)

[4.3 Diagrama de Flujo de Datos 24](#_Toc458703250)

[4.3.1 Entradas y salidas del sistema 26](#_Toc458703251)

[4.3.2 Diagrama de contexto 26](#_Toc458703252)

[4.4 diagrama de casos de uso 27](#_Toc458703253)

[4.5 descripción de casos de uso 28](#_Toc458703254)

[capitulo 5 32](#_Toc458703255)

[5.1 modelo entidad-relación 32](#_Toc458703256)

[5.2 Modelo Físico Relacional 33](#_Toc458703257)

[Bibliografía 34](#_Toc458703258)

# Introducción

Con el crecimiento de la computación y la informática, la cantidad de información y la velocidad a la que viaja ha aumentado de forma exponencial en los últimos años, y la cantidad de información a la que un usuario se ve enfrentado es exorbitante que genera un desconcierto del usuario a la hora de adquirir un servicio o producto. Esto acaba en lo usuarios utilizando ayuda de forma externa, o en los usuarios desistiendo la idea de adquirir el producto. Para esto, los sistemas de recomendaciones están diseñados para filtrar la inmensa cantidad de información que es recibida por el usuario como también especificar dicha información a la que realmente es apreciada por el mismo. Estos sistemas, que están diseñados para ayudar al cliente, pueden asistir al usuario de forma explícita, ya sea pidiendo al usuario que especifique ciertas áreas de interés dentro de varias opciones o que genere una lista con todas sus preferencias. O de forma implícita, como es analizar las visitas a cierto articulo realizado por el usuario o generar un registro de los temas que el usuario ve en una tienda online.

De este modo, se realizara una aplicación web con un sistema de recomendaciones, orientado al contexto turístico, que lograra acotar las diferentes opciones entregadas al usuario, permitiendo entregar a este las más cercanas a sus preferencias personales.

# Capítulo 1: Antecedentes Generales

## 1.1 Motivación

Los sistemas de recomendación son herramientas muy poderosas que facilitan de gran manera las búsquedas de los usuarios, en el ámbito del turismo facilitaría al usuario elegir la mejor alternativa, ya sea para el lugar a visitar, el hospedaje, etc. en esta área en particular, el usuario se ve enfrentado al problema de no conocer específicamente lo que está buscando, ni las opciones disponibles, a diferencia de otras categorías. Por ejemplo, al visitar otra ciudad u otro país, se desconoce la cantidad de opciones de transporte, ya sea vía terrestre, aérea o marítima (en casos de islas como isla de pascua ), tampoco se conoce las opciones de hospedaje, a todo esto debe incluirse las opciones de pago como también lo diferentes planes existentes para cada opción, a diferencia de otras categorías como por ejemplo, cuando se desea ver un video, la cantidad de variables que están involucradas son mucho menores, ya que de por sí, la búsqueda está acotada por lo general dentro de una categoría, de un tiempo mínimo o máximo de duración, y el contenido específico que se desea ver (por ejemplo, no todos los videos en la categoría de acción son iguales). Por eso son fundamentales los sistemas de recomendaciones para asistir en la búsqueda para los usuarios para entregar un buen servicio.

Por otro lado existe un beneficio para las empresas y compañías, al entregar los servicios, ciertos productos o servicios son recomendados a los usuarios, que antes desconocía, por ejemplo, en el caso de hospedajes, los hoteles famosos tienden a sobresalir por sobre el resto, mientras que hoteles pequeños tienden a desaparecer ante la masiva cantidad de ofertas, pero este hotel pequeño puede tener las características necesarias para el usuario, por ejemplo, en temas de costo, o por ejemplo el hotel famoso entrega servicios que no le son necesarios a los usuarios, como por ejemplo servicios de almuerzo, porque el usuario prefiere consumir comidas típicas en los restaurantes de la ciudad. El sistema de recomendación se transforma en un mecanismo principal, a la hora de planificar y realizar compras de servicios o productos, ya que según la recomendación basado en las especificaciones personales del usuario, pueden cambiar entre un "comprar o no comprar" el servicio o producto ofrecido por sobre otro, puede cambiar totalmente la decisión del usuario. Por eso es indispensable no solo en aplicaciones web, si no de manera general, la existencia de sistemas de recomendación, ya que entregan beneficios tanto al usuario como a las compañías.

En este proyecto se realizara la implementación de una aplicación web con un sistema de recomendaciones para el sector turístico, atendiendo a las especificaciones de un usuario único, tanto de manera implícita como explicita.

## 1.2 Objetivo General

* Implementar una aplicación basada en sistemas de recomendaciones que permita obtener las preferencias del usuario en el contexto de turismo.

## 1.3 Objetivos Específicos

* Analizar las técnicas de recomendación que más se adecuen al problema.
* Definir el modelo de sistema de recomendaciones a implementar.
* Definir requerimientos funcionales de la aplicación web.
* Definir requerimientos no funcionales de la aplicación web.
* Definir la arquitectura del sistema y sus requisitos básicos a nivel de hardware.
* Implementar el sistema.

## 1.4 Alcances y Limitaciones

Este trabajo de Titulo se va a desarrollar una aplicación Web con un Sistema de recomendación orientado al sector Turismo, el cual abarcara el Territorio Nacional de Chile. Para el desarrollo e implementación de este trabajo de Titulo, se tomará como base la siguiente información:

Albín Rodriguez, Antonio Pedro. «SISTEMA DE RECOMENDACIÓN COLABORATIVO BASADO.» Jaén.

Álvarez Vidales, José Borja. «Sistemas de recomendación de servicios turísticos personalizados en el destino turístico inteligente.» Máster Universitario en Dirección y Planificación del Turismo, Málaga, 2014.

Cendejas Valdéz, José Luis. «modelo de desarrollo de software integral colaborativo.» Tesis Doctoral, Puebla, Mexico , 2014.

Francesco Ricci, Lior Rokach, Bracha Shapira, Paul B. Kantor. *Recommender Systems Handbook.* Springer, 2011.

González Calero, Pedro Antonio. «Recomendación Social con Aplicaciones a Turismo.» Memoria Máster, Madrid, 2014.

Guzman Álvarez, Cesar Augusto. «Sistema de Recomendación y planificación turística de la ciudad de valencia vía web.» Memoria, Valencia, 2009.

Leiva Olivencia, José Luis. «Realidad aumentada bajo Tecnología Móvil basada en el contexto Aplicada a destinos Turísticos.» Tesis Doctoral, Málaga, 2014.

Martínez Mimbrera, Fransisco Jesús. «Sistema de Recomendación Actualizable y con Gestión de usuarios.» Trabajo de titulo, Jaén, 2008.

Navarro Benites, Karlen. «Recomendación social con aplicaciones a turismo.» Memoria de Master, madrid, 2014.

Pérez Cordón, Luis Gonzaga. «Modelos de Recomendación con falta de información. Aplicaciones al sector trístico.» Memoria de Tesis, Jaén, 2008.

Ruiz Iniesta, Almudena. «Estrategias de recomendación aplicadas a repositorios de recursos educativos.» Proyecto Fin de Máster, Madrid, 2009.

# Capítulo 2: Marco Teórico

## 2.1 Contexto: Sistema de Recomendación [1][2][5]

Con el aumento exponencial que ha recibido la comunicación on-line y la facilidad que se tiene a la información hoy en día, para muchos se puede ver como algo positivo, pero por el lado de búsqueda de información, para los usuarios es un punto negativo, ya que en la cultura de hoy la rapidez es algo necesario, saber dónde está lo que queremos es fundamental. Los usuarios se encuentran frente a una masiva cantidad de información que es irrelevante, para esto, el filtrado de información es fundamental, no solo para los usuarios, sino también para empresa y compañías que tiene algún producto o servicio, que puede estar escondido bajo la extensa cantidad de información. Para todo esto, los sistemas de recomendaciones no son solo potentes herramientas, si no que indispensables tanto para usuarios como para empresas. Es por esto que los sistemas de recomendaciones están integrados a muchas aplicaciones web y móviles, para el ámbito comercial, como en Amazon, eBay o mercado libre, que se recomienda productos según las compras realizadas por usuarios con gustos similares. En el ámbito de la entretención, como [Netflix](https://www.netflix.com/) o Youtube, que recomiendan según las preferencias del usuario. O como en casos de las relaciones sociales, como Facebook, que recomiendan amigos según gustos similares (libros en común, películas en común, amigos en común).

Un Sistema de recomendaciones es una herramienta de filtrado, que a diferencia de muchas otras herramientas de esta misma categoría, el sistema de recomendaciones se caracteriza principalmente por su adaptabilidad con el usuario. Caso contrario a los conocidos motores de búsqueda, como Google, que generan los resultados a base de búsqueda semántica y árboles de jerarquía, estos no pueden diferenciar entre 2 resultados a base de su relevancia para al usuario, para el caso de los sistemas de recomendación, su principal tarea es entregar los resultados más relevantes, filtrando los resultados innecesarios. Mientras el motor de búsqueda entregue el mismo resultado a diferentes usuarios, el sistema de recomendación entregara diferentes resultados para cada uno del os usuarios utilizándolo, ya que cada usuario tendrá preferencias diferentes del resto a los cuales el sistema deberá adaptarse.

## 2.2 Definición: Sistemas de Recomendación [7][5]

Existen diferentes tipos de sistemas de recomendación, que priorizan diferentes características del usuario o grupo de usuario, pero la teoría fundamental es la misma. Supongamos un usuario realiza la búsqueda de cierto elemento, por ejemplo, una película, se le entregara una infinidad de películas como resultado, a cada uno de estos resultados se les denominara "ITEM" (I), por lo tanto, si el resultado entregado es un conjunto de ítems de 1 hasta K ( I={1,2,...,k} ), el sistema de recomendaciones deberá encargarse de acotar los resultados, según la prioridad que se le entrega al sistema de recomendaciones, para entregar una lista personalizada al usuario con los "mejores" ítem , los que sean de utilidad o que más satisfaga sus necesidad.

Como ya se mencionó, un sistema de recomendación es un tipo de filtrado adaptativo a las preferencias del usuario, por ende, el sistema trabaja con el perfil de los usuarios para realizar las recomendaciones, mientras más información de usuario se pueda obtener (por ejemplo gustos musicales, edad, lugar geográfico), más exacto y relevante será la recomendación entregada al usuario. Por eso en necesario de que el perfil del usuario tenga las información más relevante, generalmente se consigue preguntando a un usuario un poco de información, de manera que no sea tediosa, para no perder al usuario, luego el sistema utilizara mecanismos de retroalimentación para seguir adquiriendo información y poder mejorar las recomendaciones, como se explicara más adelante en el punto 2.5.

Los sistemas de recomendación son elementos vitales tanto para empresas como para usuarios. Mientras que la empresa desea que cierto producto o servicio sea más visible para su público, como a los usuarios que pueden filtrar la información que se le es entregada al momento de buscar, por lo tanto, un sistema de recomendaciones debe tener 2 características muy importantes para poder aumentar la satisfacción de los clientes, los cuales son predecir y recomendar el mejor ítem. Predecir es la característica que permite al sistema identificar el producto o servicio que más le interesara al usuario antes de que el usuario lo especifique. Mientras que recomendar el mejor ítem es la característica que permite al sistema comparar todos los posibles productos o servicios a recomendar, e identificar cuál de todos será el que satisfaga más al usuario y recomendarlo. Estas 2 características son en esencia lo que separa a un sistema de recomendaciones de los demás sistemas de filtrado de información ya que estas características nos dan la posibilidad de adaptarse a las preferencias de los usuarios utilizando el sistema.

## 2.3 Técnicas Actuales [1][2][4][5][6][7][10][11]

Se han diseñado diferentes paradigmas para los sistemas de recomendaciones, que priorizan diferentes características del usuario para realizar las recomendaciones de ítem, las más importantes son:

* **Recomendación basada en Contenido: utilizado en páginas como YouTube, es uno de los paradigmas más utilizado, que recomienda según las preferencias de contenido del usuario, por ejemplo, si el usuario disfruta ver videos de futbol, YouTube recomendara videos que tengas el mismo contenido.**
* **Recomendación Basada en conocimiento: a diferencia de las recomendaciones basadas en contenido, esta compara los ítems similares según la satisfacción , por lo tanto, genera una relación entre la satisfacción de un ítem y la necesidad de un usuario, con esto el sistema infiere la utilidad de un ítem para el usuario especifico y entrega ese ítem como recomendación**
* **Recomendación Basada en Utilidad: parecidas al sistema de recomendaciones basado en conocimiento, pero se diferencia en que este modo de recomendación calcula cada una de las utilidades entre todos los servicios y las preferencias del usuario, y utilizando el cálculo de la utilidad genera la recomendación.**
* **Recomendación Demográfica: utilizado muchas veces en páginas de relaciones sociales (como por ejemplo paginas para encontrar pareja), filtran los ítems según el usuario, como por ejemplo: edad, sexo, residencia, profesión.**
* **Recomendación Colaborativa: utilizado en páginas como eBay, recomienda al usuario los mismos ítems que realicen los demás usuarios con los gustos similares, por ejemplo si un usuario intenta comprar un celular, al final de página encuentran una sección que dice: "usuarios que han buscado este ítem también han comprado lo siguiente".**

## **2.4 Características de los Sistemas de Recomendación [9][5]**

Se han desarrollado muchos tipos de sistemas de recomendación, cada uno abarcando a necesidades diferentes para cada tipo de usuarios o a las necesidades de la compañía que los implemento, para poder entender qué tipo de sistema de recomendación es más eficiente para cada situación necesitamos analizar las ventajas y desventajas para cada tipo.

## 2.4.1 Ventajas de los Sistemas de Recomendación según su tipo [11] **[9][5]**

Para cada tipo de usuario o para cada tipo de situación, los sistemas de recomendaciones están diseñados para acomodar a las necesidades especificadas y entregar la mejor recomendación, a continuación veremos las ventajas de los diferentes sistemas de recomendación según su tipo:

* Recomendación basadas en contenido: Las principales ventajas de este sistema de recomendaciones, es que no basa la recomendación en la subjetividad del gusto del usuario o de otros usuarios, ya que basa la recomendación en contenido que ya ha sido visitado o aprobado por el usuario. Como consecuencia no existe en el sistema de recomendación, intercambio de información para hacer recomendaciones a otros usuarios, el sistema separa las recomendaciones por usuario, por lo tanto, si un ítem es del agrado de un usuario, el sistema no intenta recomendar ese ítem a otros usuarios.
* Recomendaciones basadas en conocimiento: A diferencia de las recomendaciones basadas en contenido, las basadas en conocimiento comparan los ítems actuales con recomendaciones exitosas, por lo cual esta realiza recomendaciones más deseables para el usuario. Tampoco requieren mucha cantidad de información, sino que solo es necesario que se tenga un conocimiento general de los ítems a recomendar y un conocimiento informal de las preferencias del usuario.
* **Recomendación Basada en Utilidad: Las principales ventajas de este modo de recomendación, es que el cálculo de utilidad no es necesariamente dependiente del servicio o producto en sí, sino que también puede generar el cálculo de utilidad en las condiciones o contexto del producto, como lo son disponibilidad del producto o la puntuación de un vendedor (como en páginas de venta online como eBay o Amazon).**
* Recomendación Demográfica: para este tipo de recomendación, su principal ventaja es la posibilidad de filtrar de una manera más especifica la cantidad de ítems, basándose en la información personal del usuario, con esto, el sistema procede a comparar la información de diferentes usuarios, encontrando un factor común entre la información filtrada en los diferentes perfiles, así puede entregar recomendaciones basándose en lo que se ha recomendado a otros usuarios con las mismas características, además de prescindir de muy poca información para realizar las recomendaciones.
* Recomendación colaborativa: Para este sistema de recomendación, las ventajas están basadas en el hecho de que compara usuarios para realizar las recomendaciones, este permite al sistema recomendar ítems que son difíciles de analizar y filtrar, además de entregar recomendaciones no esperadas, pero con un alto nivel de satisfacción al usuario, ya que como un usuario, con las mismas características que otro, cataloga como bueno cierto ítem, a otro usuario con los mismos gustos o mismos filtros, se le entrega ese ítem como recomendación.

## 2.4.2 Desventajas de los Sistemas de Recomendación según su tipo [5][9]**[11]**

A pesar de que los sistemas de recomendaciones son herramientas muy poderosas, estos no están exentos a los problemas, ya que, como fundamento de los sistemas de recomendaciones, estos son de tipo predictivos, y en algunas condiciones a los sistemas de recomendación se les vuelve compleja la recomendación de un ítem, a continuación analizaremos algunos casos que dificultan el trabajo a los sistemas de recomendaciones:

* Recomendaciones basadas en contenido : Como el sistema depende del contenido que el usuario a designado como bueno , este sistema se ve limitado entregar solo Ítems de la misma categoría, por lo tanto, si el usuario desea encontrar algo diferente, el sistema no podrá realizar ninguna recomendación afuera de los contenidos ya explorados. Como el contenido debe haber sido explorado por el usuario, para los usuarios nuevo, esto genera un limitante al sistema, ya que sin contenidos revisados con anterioridad por el usuario, el sistema se ve imposibilitado para recomendar.
* Recomendaciones basadas en conocimiento: las Desventajas son similares a las recomendaciones basadas en conocimiento, ya que como compara las posibles recomendaciones con recomendaciones exitosas, las recomendaciones futuras tienden ir por una misma temática, por lo que si el usuario busca algo nuevo, las recomendaciones no podrán cumplir con las expectativas, a menos de que sea refinado explícitamente por el usuario.
* **Recomendación Basada en Utilidad: la desventaja de este modo de recomendación, es que necesita una participación muy activa del usuario, para poder refinar el cálculo, por ejemplo, si se desea que cada recomendación considere la puntuación del vendedor, esto debe ser explícitamente identificado. Por esto, este sistema de recomendación es estático, a diferencia de otros sistemas de recomendación, este solo recomendara lo que el usuario especifica.**
* Recomendación Demográfica: la recomendación demográfica está muy limitado a las características personales del usuario, como no se utiliza preferencias ni gustos personales sobre los contenidos que debe recomendar el sistema, esto genera una limitación al sistema para recomendar ítems que cumplan con la satisfacción de los usuarios.
* Recomendación colaborativa: como este sistema utiliza la recomendación basado en comparación de usuarios, la información que se intercambia por el sistema por recomendaciones funciona excelente para cuando los usuarios son pocos, pero cuando aumenta el tamaño de usuarios del sistema, las comparaciones que debe hacer entre usuarios aumenta y las recomendaciones que puede realizar el sistema disminuyen. Por otro lado, cuando aumenta la cantidad de ítems que satisfacen a los usuarios, el cómputo que debe realizar el sistema para comparar a un usuario con otro también aumenta, dificultando la tarea del sistema.

Como se puede ver, todos los sistemas de recomendación tienen sus características a favor y sus características en contra, por lo que se hace primordial identificar el tipo de usuario o el tipo de negocio al cual el sistema se implementara para poder tener mayor efectividad. Por otro lado, se pensó en diseñar un sistema que combinara diferentes metodologías de sistema de recomendación y combinarlos en uno, a los cuales se les denomina sistemas de recomendación hibrido, para aprovechar sus ventajas combinadas y para opacar las desventajas de estos, pero al igual que usar solo una metodología para el sistema de recomendación, de todas maneras es imprescindible identificar el contexto en donde se aplicara el sistema de recomendación, para poder aprovechar las ventajas que ofrecen las distintas metodologías.

## 2.5 Sistemas de Recomendación Híbridas [1][5][12]

**Un sistema de Recomendación Hibrida es una combinación de 2 tipos diferentes de metodologías de recomendación con el objetivo de eliminar las desventajas de una de los sistemas de recomendación con las ventajas de la otra, para así optimizar las recomendaciones, usualmente están involucrados con lógica difusa y con computación evolutiva.**

**Al igual que los sistemas de recomendaciones normales, existen diversos tipos de recomendaciones hibridas, a continuación mencionaremos y explicaremos alguna de ellas.**

* Recomendación Hibrida Mezclada: para una misma recomendación se le aplican diferentes metodologías de recomendación, luego, los resultados de combinan para así realizar una recomendación.
* Recomendación Hibrida Ponderada: para una misma recomendación, se obtiene los resultados (Ítems) de los diferentes Sistemas de Recomendaciones, a cada uno de estas recomendaciones se les entrega un puntaje dependiendo del nivel de satisfacción del usuario , luego de esto, los resultados se combinan entre los 2 sistemas anteriores, para entregar una nueva recomendación .
* Recomendación Hibrida alternada: El sistema de recomendación tendrá un parámetro, definido previamente en el sistema, que le indicara cuando debe cambiar entre un sistema de recomendación con otro.
* Recomendación Hibrida en cascada: los Ítem candidatos a recomendar son refinados sucesivamente por diferentes sistemas de recomendación, por ejemplo, una serie de películas es pasado por un sistema de recomendación demográfico, las recomendaciones candidatas resultantes de aquello, serán expuesto a otro sistema de recomendación, por ejemplo uno basado en contenido, así se sigue el proceso hasta refinar lo más posible las recomendaciones.
* Recomendación Hibrida de combinación de características: En esta metodología, se unen en un solo conjunto todas las características por las cuales se debe someter una recomendación, de todos los sistemas de recomendación a utilizar y se implementan en un solo algoritmo, de tal forma las recomendaciones solo son evaluados por un único algoritmo.
* Recomendación Hibrida de aumento de características: similar al modelo en cascada, este expone los candidatos a filtros sucesivos por diferentes sistemas de recomendación, pero se diferencia en que la información entregada por el sistema de recomendación anterior se utiliza como característica adicional a los datos.
* Recomendación Hibrida meta-level: Similar al modelo anterior, la recomendación resultante y el modelo completo del sistema anterior, entran dentro de un nuevo sistema de recomendaciones, la diferencia principal con el sistema de recomendación hibrida de aumento de características, es que en este caso, el modelo completo como dato de entrada, en cambio el modelo anterior el resultado de la recomendación anterior, solo se usa como datos parametrizables a utilizar en la siguiente recomendación.

## 2.6 Mecanismos de Retroalimentación [1][5][9]

Como ya se mencionó anteriormente, los sistemas de recomendación tienes diferentes formas para generar la recomendación a los usuarios, pero una parte esencial que determina la efectividad del sistema de recomendación es la necesidad de mecanismos de retroalimentación, ya que, el usuario va a ir añadiendo nuevas preferencias, ya sea por preferencia personal (el mismo usuario desea que las nuevas recomendaciones abarcan ítem diferentes a los antes entregados) o por que las recomendaciones anteriores no han sido de su agrado. Los mecanismos de retroalimentación permitirán al sistema afinar y aumentar la calidad de las recomendaciones que el sistema realiza.

Principalmente existen 2 métodos para la retroalimentación que se adecuan dependiendo del tipo de sistema de recomendación, y estos son retroalimentación implícita y explicita.La diferencia entre estos 2 métodos, más allá de que uno esté más relacionado con el usuario (explicito) y el otro no (implícito), son los paradigmas a los cuales se han orientado los sistemas de recomendación y como se integra el mecanismo de retroalimentación al sistema.

## 2.6.1 Retroalimentación Explicita [1][5][9]

Como su nombre bien lo indica, la retroalimentación explicita requiere participación directa del usuario donde el sistema le preguntara al usuario y este entregará sus preferencias al sistema de recomendación, la forma como se realiza el sistema la consulta de las preferencias al usuario puede variar mucho, como métodos simple como preguntar si la recomendación que recibió es de su agrado o no , o método más complejo como preguntar si el ítem fue de su agrado, pero indicar cuanto fue la satisfacción de la recomendación de forma numérica, como seria recibir la recomendación, indicar del 1 al 5, cuanto se cumple de mi preferencia o necesidad en esta recomendación (por ejemplo un video sobre autos, marcar el numero 5 representaría la opinión del usuario como "si, este tipos de videos son los que busco." o si es el valor 1 representaría la opinión del usuario como "no, no me interesa ver videos como este.").

la ventaja que se encuentra al realizar el mecanismo explicito es que se disminuye en parte, la predicción del sistema de recomendación, ya que entrega al usuario parte fundamental sobre la aceptación o rechazo sobre las recomendaciones, con esto el sistema puede trabajar de una forma más exacta. Por otro lado se crea una desventaja, porque como depende del usuario, este también depende de la voluntariedad y de la disponibilidad de tiempo que este tenga.

## 2.6.2 Retroalimentación Implícita [1][5][9]

Este mecanismo de retroalimentación, obtiene la información sobre si una recomendación fue exitosa o no sin involucrarse con el usuario de forma directa, ósea, obtener las preferencias del usuario sin que este sea consciente de lo que realiza el sistema de recomendación. Por lo tanto, este mecanismo utiliza herramientas que trabajan de forma indirecta con el usuario, utilizando métodos como cuánto tiempo dedica un usuario a ver determinado ítem (por ejemplo cuanto tiempo dedica el usuario a ver un video, antes de cambiar a otro), o cuantas veces el usuario ingresa a ver determinado ítem (como por ejemplo, un artículo en venta en una página de venta online).

Como ventaja de este mecanismo está el hecho de que no necesita interactuar con el usuario, por lo que se puede realizar en cualquier momento por el periodo de tiempo que sea necesario sin necesidad de fastidiar al usuario, pero por otro lado, es un mecanismo que se basa totalmente en el contexto y es excesivamente hipotética. Lo cual hace que este mecanismo no sea el más apropiado para todos los casos.

# Capítulo 3:

## 3.1 Metodología de desarrollo [3] [13]

Con el avance de la tecnología y la computación, también se ha cambiado la forma de desarrollar software, durante los años 50 y 60, en el mundo todavía no existía una forma estandarizada de desarrollar software, tanto en el área de programación como en el área de gestión del proyecto, por consiguiente, se obtenían resultados impredecibles, donde no se conocía la fecha exacta en que se concluiría el proyecto de software. Por otro lado, en la programación, no se tenía documentación estandarizada, lo que dificultaba muchísimo la actualización y mejoramiento de software por parte del personal que no fuera el mismo programador. En el año 1968 se realiza la primera conferencia organizada por la OTAN (Organización del Tratado del Atlántico Norte) sobre el desarrollo de software, con el cual crea el término "crisis del software" que se refiere a la dificultad de crear programas libres de errores, que sean fácil de entender y que puedan ser verificados, este término engloba una serie de sucesos que se producían al desarrollar un software, tales como:

* los proyectos no se terminaban en los plazos estipulados
* los proyectos no se ajustaban al presupuesto inicial
* los proyectos no cumplían con las especificaciones
* código inmantenible que dificulta la gestión y evolución del proyecto
* baja calidad del software

Por esto, a finales de la década de los 60, la ingeniería de software tiene como principal objetivo es convertir el desarrollo de software, en un proceso formal, metódico y estructurado, que permita entregar un producto final de alta calidad. Desde el 1985, se ha visto un crecimiento en la cantidad de herramientas y metodologías que tratan de dar con la solución a los problemas del desarrollo de software, pero por la experiencia se ha demostrado que las metodologías que dan como resultado un proyecto exitoso son los que tienen como eje central una serie de procesos que permiten organizar y luego controlar el proyecto, mientras que las metodologías que no siguen esa idea, corren un alto riesgo de fracasar.

Existen diferentes metodologías de desarrollo y cada uno de ellos realiza los procesos de una perspectiva en particular, la cual se potenciara en las diferentes fases y procesos que esta tiene, las fases y procesos no son definitivos, sino que son abstractos y más bien, encasillan y regulan como se debería realizar el proceso. Algunas de las metodologías más conocidas son:

* Metodología de Cascada: es un enfoque que ordena de manera rigurosa las etapas para el desarrollo de software, de tal forma, que el inicio de cada etapa debe esperar la finalización de la etapa anterior. Las etapas fundamentales de esta metodología son: Análisis de requisitos, Diseño del Sistema, Diseño del Programa ,Codificación, Pruebas, Implantación y Mantenimiento.
* Metodología en espiral: en esta metodología funciona de una manera similar al de cascada, pero cada una de las fases de la metodología de cascada termina con una evaluación de riesgos y un prototipo. Los prototipos permiten a los usuarios determinar si el proyecto continúa, debe volver a fases anteriores, o debe terminar. Sin embargo, las fases son todavía lineales, los requisitos se realizan en la fase de requisitos, el diseño en la fase de diseño, y así sucesivamente. Las etapas principales de la metodología son: Determinar Objetivos, Análisis del riesgo, Planificación, y por ultimo Desarrollar y probar.
* Metodología Iterativa e incremental: Es un enfoque que centra un grupo de procesos en un ciclo, que se realiza varias veces durante el desarrollo del software, para así corregir errores que se pueden haber realizado en etapas anteriores,

Cuando se diseñaron estas metodologías, se pensaba en alguna forma de estandarizar y lograr predecir a largo plazo el costo y resultado, de la manera más exacta posible, del software a realizar, pero con esto, terminaron creando metodologías muy rígidas, estrictas y difíciles de adaptar a los cambios no esperados por el programador, por esto, a mediados de los 90, se creó una metodología denominada "Metodología de Desarrollo Ágil", que son basados en la metodología incremental, donde permite que las soluciones y los requisitos del programa sean evolutivos, ósea, puedan adaptarse a los cambios no planeados de antemano a los cuales pueda enfrentarse el programador durante el desarrollo del software.

## 3.2 Metodología de Desarrollo Ágil [3] [13]

Con el desarrollo del internet y las telecomunicaciones, el mundo ha estado en contacto de una manera casi instantánea, los negocios han crecido a un nivel considerable, alcanzando a diferentes países alrededor del mundo, donde operan y entregan servicio a sus clientes. Esto genera que los negocios se encuentren en un ambiente global que cambia rápidamente y las empresas deben reaccionar a esto. El software es parte importante dentro de las organizaciones, por eso, es fundamental que el software cuente con la facilidad de adaptarse rápidamente a este ambiente cambiante. Por consiguiente, una de las primeras características que se pide por parte de las empresas a los programadores, es la rapidez y adaptabilidad de las características durante el desarrollo del software, incluso, algunas empresas dan prioridad a la velocidad del desarrollo del software por encima de la calidad.

La metodología de desarrollo ágil, como se mencionó anteriormente es una metodología basada en el desarrollo iterativo e incremental, pero con la diferencia de que permite la "evolución" de los requerimientos y soluciones, esto produce que el riesgo de los desarrollos de software se minimicen, desarrollando unidades de software de funcionalidad acotada en lapsos de tiempos breves, usualmente se utilizan ciclos de 1 a 4 semanas, a las cuales se les llama iteraciones. Al igual que las metodologías clásicas, la metodología de desarrollo ágil tiene etapas fundamentales, estas son las etapas que se deben realizar en cada iteración durante el desarrollo, y estas son: Planificación, Análisis de requerimientos, Diseño, Codificación, Revisión y Documentación. Después de realizada cada iteración, se vuelven a evaluar las prioridades del proyecto.

La principal diferencia existente entre las metodologías clásicas y la metodología de desarrollo ágil es la ideología en la que se orienta el desarrollo. Por un lado que las metodologías clásicas se basan en los estándares que han sido ampliamente aceptados por la industria, estas metodologías tienen alta resistencia a los cambios, ya que estas tienen un conjunto elevado de normas y políticas. Mientras que las metodologías de desarrollo ágiles tienen procesos menos controlados y pocas restricciones, lo que le permite estar preparados de antemano para abordar el cambio y evolucionar con él. La metodología de desarrollo ágil tiende a buscar las soluciones a través de la heurística, ósea, mediante métodos no rigurosos, que es lo que genera la gran diferencia entre las 2 ideologías. Además de esto también se realiza una diferencia de cómo se debe relacionar los programadores con los clientes. Para las metodologías clásicas existe un contrato formal y definido, además de que el trabajo con el cliente es basado en reuniones, mientras que en las metodologías de desarrollo ágil el contrato debe ser flexible, y el cliente debe ser parte del equipo de desarrollo.

## 3.3 Ventajas y desventajas de las Metodologías de Desarrollo Ágil [3] [13]

Para cada programador existe un tipo de solución que satisface más su forma de trabajo, por esa razón no existe un método universal para el desarrollo de software, sino que se debe utilizar el que acomode más a al estilo y forma de programar del desarrollador. Al igual que con las métodos de recomendación, las diferentes metodologías potencian diferentes áreas o aspectos del desarrollo del software. Las diferentes Con la diferencia de ideología planteada por la metodología de desarrollo ágil se potencian. A continuación se hablara sobre las principales ventajas de utilizar metodología de desarrollo ágil.

* La principal ventaja es la ya antes mencionada adaptabilidad y evolución de las soluciones de software hacia los cambios que ocurren durante el desarrollo, principalmente a su proceso iterativo, esto permite realizar un análisis de los requisitos en cada una de sus iteraciones, para así poder aplicar modificaciones evitando grandes pérdidas en cuanto a costes, motivación o tiempo.
* Como la metodología incita al cliente a ser parte del grupo de desarrollo, permite al cliente observar cómo está avanzando el proyecto y a participar (generalmente dando opinión), lo cual entrega tranquilidad, además que reduce el impacto de los cambios que desee hacer el cliente y reduce las malinterpretaciones respecto a los requerimientos que se den con los desarrolladores y el cliente, gracias a la integración de este al grupo y por la propiedad iterativa de esta metodología. Esto es un punto importante, ya que esto ayuda a no malgastar el tiempo y dinero del cliente desarrollando soluciones innecesariamente generales y complejas que en realidad no son requisitos del cliente.
* Atención continua a la excelencia técnica y al buen diseño.
* Mejora continua de los procesos y el equipo de desarrollo.
* Entrega continua y en plazos cortos de software funcional.
* Importancia de la simplicidad al eliminar trabajo innecesario.
* Cada componente del producto final ha sido probado y satisface los requerimientos.

Como toda metodología, esta tiene sus ventajas, pero siempre están acompañada de desventaja que deben ser consideradas al momento de realizar un desarrollo de software, las cuales son las siguientes:

* Falta de documentación del diseño. Los comentarios y el mismo código se utilizan como documentación, en proyectos grandes se necesitara leer mieles de líneas de código para poder entender su funcionamiento.
* Problemas derivados de la comunicación oral. Ya que estas son muy difíciles de mantener en el tiempo y pueden generar ambigüedad.
* Fuerte dependencia de las personas. como no se tiene una buena documentación, se genera una fuerte dependencia entre las personas para poder entender lo que otro esta desarrollando
* Falta de reusabilidad derivada de la falta de documentación
* Restricciones en cuanto a tamaño de los proyectos
* Problemas derivados del fracaso de los proyectos ágiles. Si un proyecto ágil fracasa no hay documentación o hay muy poca; lo mismo ocurre con el diseño. La comprensión del sistema se queda en las mentes de los desarrolladores.

Al analizar las ventajas y desventajas correspondientes a la metodología de desarrollo ágil, podemos ver que tiene muchas ventajas a la hora de adaptarse a los cambios repentinos que puede abordar el desarrollo de software, pero encuentra un fuerte déficit en el área de documentación y exige una alta dependencia entre los desarrolladores. Teniendo en cuenta ambas áreas (ventajas y desventajas), se ha elegido utilizar metodología de desarrollo ágil ya que el proyecto a realizar en este trabajo de titulo no es de un gran tamaños, y como el equipo de desarrollo será de una sola persona, no se encontrara problemas en la dependencia de los miembros del equipo ni en la documentación del código.

# Capitulo 4 :

## 4.1 Diseño Conceptual del Proyecto

En este trabajo de titulo se implementara una aplicación web que permitirá recomendar a los usuarios una lista de ítems que acomode sus preferencias, para eso deberemos diseñar la forma como funcionara la aplicación web. para ello se determino que el sistema solo va interactuar con 1 entidad externa, que serán los usuarios, estos tendrán 2 roles, el de usuario normal que permite tener una cuenta en el sistema (que llamaremos sesión), que tendrá los mecanismos básicos del sistemas, tales como buscar, comentar y dejar nota (ranking) a los ítems del sistema, y el de administrador que tendrá las mismas características pero tendrá mecanismos más avanzados dentro del sistema como es crear y borrar ítem (esto se verá más adelante en los puntos 4.2.1 y 4.5).

El sistema utilizara el modelo de recomendaciones de tipo colaborativa, ósea, que recomienda ítems a usuarios con gustos similares, por ejemplo Usuario 1 disfruto del ítem A, y el Usuario 2 también disfruta del ítem A, en caso de que el Usuario 1 disfrute del ítem B, este ítem será recomendado al usuario 2. Para lograr esto el sistema poseerá un sistema de ranking, que permitirá a los usuarios dejar una nota entre 1 y 5, de tal forma se podrá medir el nivel de satisfacción del ítem para cada usuario, además de esto los mejores ítem también serán recomendados a todos los usuarios.

## 4.2 definición de los requerimientos

Los requerimientos de un sistema son la condición, necesidad o capacidad que se debe encontrar en un sistema o componente del sistema para alcanzar o satisfacer el estándar o especificación de este.

en este trabajo de titulo, no se trabaja con un usuario especifico, así que la asignación de los requerimientos estarán orientados a cumplir las necesidades de un sistema de recomendación.

## 4.2.1 requerimientos funcionales

los requerimientos funcionales son los que indican de forma detalladas el funcionamiento del sistema, explicando las entradas, salidas, condiciones que cumplen y sus excepciones, en otras palabras, lo que el sistema deba y no debe hacer .

para la implementación de trabajo los requerimientos funcionales del sistema estarán bajo las siguientes condiciones.

* roles de usuarios: para el caso de este sistema, solo existirán 2 roles de usuarios, el usuario que tendrá la opción de buscar ítems dentro del sistema, además de poder dejar un comentario y dar ranking a los ítems, por otro lado, existirá el rol de administrador, que tendrá las mismas características que el usuario, pero además permitirá crear y eliminar ítems del sistema.
* El sistema contara con una base de datos en donde se almacenara los ítems correspondiente a los lugares turísticos, los usuarios, los ranking y comentarios que tendrá cada ítem.
* Validación y autenticación: todos los usuarios podrán dejar ranking y comentario después de ingresar al sistema, el sistema validará sus datos, de forma que no hayan inconsistencias en la base de datos, así mismo, solo los usuarios con rol de administrador tendrán acceso a modificar los ítem.
* el sistema será capaz de generar un script con el cual entregara las recomendaciones a los usuarios.
* Los usuarios podrán elegir filtros favoritos para que las recomendaciones sean más personalizadas.

## 4.2.2 requerimientos no funcionales

los requerimientos no funcionales son lo que indican las características de funcionamiento del sistema, como lo son la estabilidad, portabilidad y rendimiento. los requerimientos no funcionales del sistema serán los siguientes

el sistema será desarrollado en una estructura tipo Modelo Vista Controlador o MVC, ya que este proporciona facilidades en su mantención, modificación y mejoras.

los lenguajes de programación a utilizar en este trabajo serán:

* html
* ruby 2.2.1
* El framework a utilizar será ruby on rails 4.2.6
* el motor de bases de datos será mysql
* el servidor será Apache 2.2.31
* el sistema operativo Fedora 24 ( Linux)

A demás se requiere que la interfaz sea amigable y de fácil manejo para los usuarios

## 4.2.3 requerimientos de hardware

El sistema será desarrollado en un notebook con las siguientes características:

|  |  |
| --- | --- |
| Procesador | Intel Core i3-4030U (1.9 GHz) |
| Memoria | de 4 GB DDR3L |
| Disco Duro | 1 TB |
| Tarjeta Grafica | Intel HD Graphics 4400 |
| Tarjeta de red | Red 802.11b/g/n |

## 4.3 Diagrama de Flujo de Datos

Para poder entender el funcionamiento del sistema utilizaremos un diagramas de contexto para mostrar cómo funciona el sistema, los diagramas de contexto nos indican cómo se relaciones las diferentes entidades que participan en un software entre sí, mostrando de donde se mueve la información y los datos, y hacia donde llegan dentro del sistema, en este diagrama se muestran las entradas y salidas del sistema (siendo las entradas todas las interacciones de parte del usuario y la salida toda la información entregada por el sistema).

en la tabla N°1 y N°2, se podrá ver las entradas y salidas de este sistema de manera detallada, explicando el origen y destino de cada una de estas, mientras que en la Figura N°1 se podrá ver de manera grafica estas interacciones.

## 4.3.1 Entradas y salidas del sistema

|  |  |  |  |
| --- | --- | --- | --- |
| **Entradas** | | | |
| **N°** | **Descripción** | **origen** | **destino** |
| E1 | Registrarse | Usuario | Sistema |
| E2 | Solicitud Ítem | Usuario | Sistema |
| E3 | Dejar comentario | Usuario | Sistema |
| E4 | Dejar Ranking | Usuario | Sistema |
| E5 | Envió datos recomendación | Usuario | Sistema |
| E6 | Modificar perfil | Usuario | Sistema |
| E7 | Solicitud ranking | Usuario | Sistema |
| E8 | Solicitud comentario | Usuario | Sistema |
| E9 | Solicitud usuarios | Usuario | Sistema |
| E10 | Crear ítem | Usuario | Sistema |
| E11 | Solicitar ítem | Usuario | Sistema |
| E12 | Eliminar ítem | Usuario | Sistema |
| E13 | Eliminar usuario | Usuario | Sistema |

Tabla N°1: Entradas del Sistema

|  |  |  |  |
| --- | --- | --- | --- |
| **Salidas** | | | |
| **N°** | **Descripción** | **origen** | **destino** |
| S1 | Aceptar Usuario | Sistema | Usuario |
| S2 | Listar ítem | Sistema | Usuario |
| S3 | Validación comentario | Sistema | Usuario |
| S4 | Validación Ranking | Sistema | Usuario |
| S5 | Listar Recomendación | Sistema | Usuario |
| S6 | Validar modificación | Sistema | Usuario |
| S7 | Listar ranking | Sistema | Usuario |
| S8 | Listar comentario | Sistema | Usuario |
| S9 | Listar usuarios | Sistema | Usuario |
| S10 | Validar Creación Usuario | Sistema | Usuario |
| S11 | Listar ítem | Sistema | Usuario |
| S12 | Validar eliminación | Sistema | Usuario |
| S13 | Confirmación de sistema | Sistema | Usuario |

Tabla N°2: Salidas del Sistema

## 4.3.2 Diagrama de contexto

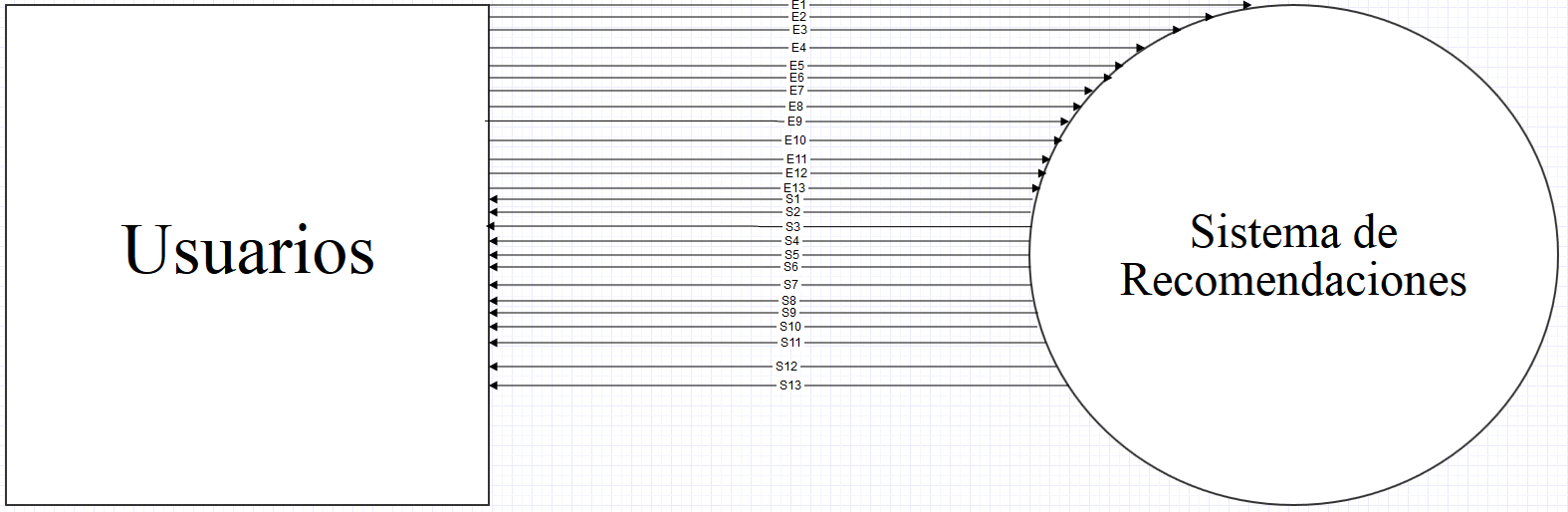


Figura N°1: Diagrama de Contexto

## 4.4 diagrama de casos de uso

Los diagramas de casos de usos nos muestran la interacción típica que ocurre entre el usuario y el sistema, mostrando así como se desea que trabaje el sistema de manera visual y sin especificar de manera detallada el comportamiento o implementación de este, tal como se muestra en la figura N°2

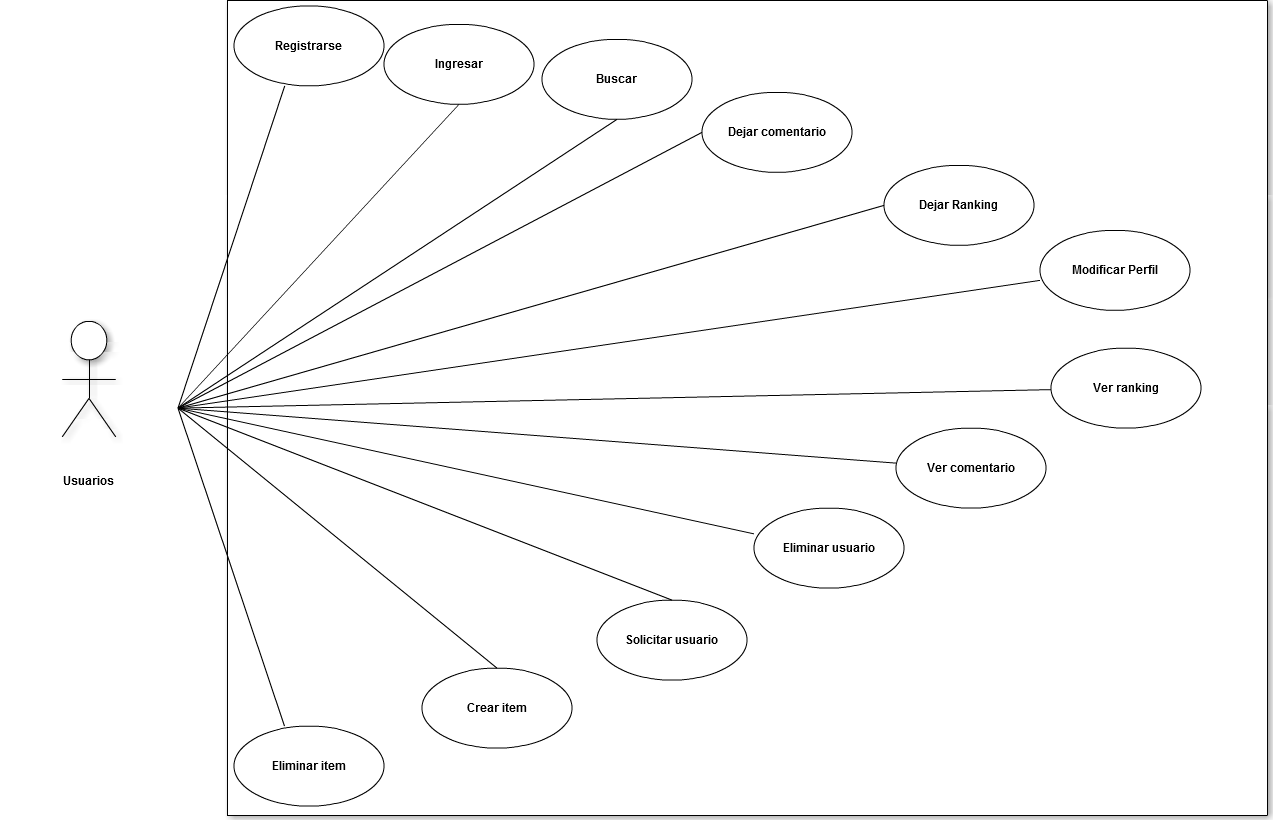


Figura N°2: diagrama de casos de uso

## 4.5 descripción de casos de uso

|  |  |
| --- | --- |
| Caso de uso | Registrarse |
| descripción | Petición del cliente para ser usuario del sistema |
| gatillante | Petición del usuario |
| actores | Usuario |
| precondiciones | Que el usuario no se haya creado ya, el usuario no puede utilizar un mail que ya este registrado |
| metas | Asegurar la información privada del cliente |
| Condiciones de fallo | Usuario existente, símbolos prohibidos para nombre de usuario |
| Caso esperado | El cliente al ingresar al sistema, entrega sus datos en la ventana de registro, los datos son guardados en la base de datos |

|  |  |
| --- | --- |
| Caso de uso | Ingresar |
| Descripción | Petición del usuario para acceder al sistema |
| Gatillante | Petición del usuario |
| Actores | Usuarios |
| Precondiciones | Que el usuario no esté utilizando alguna sesión |
| Metas | Asegurar la password del cliente |
| Condiciones de fallo | Usuario o password incorrecto |
| Caso esperado | El cliente ingresa a la ventana de inicio de sesión, ingresa usuario y password correctamente, se realiza una consulta a la base de datos para verificar que los datos existan, si existen permite el inicio de sesión, si no existen lanzara un mensaje de error |

|  |  |
| --- | --- |
| Caso de uso | Dejar comentario |
| Descripción | Usuario deja un pequeño texto sobre sus preferencias en cierto ítem |
| Gatillante | Petición del usuario |
| Actores | Usuarios |
| Precondiciones | Que no exista un comentario del mismo usuario en el mismo ítem |
| Metas | Asignar comentario al ítem correspondiente |
| Condiciones de fallo | Que el usuario intente comentar sin iniciar sesión |
| Caso esperado | Usuario utilizando su sesión, ingresa a la página de ítem, en la sección designada escribirá el comentario deseado, el sistema validara al usuario, si la sesión es validada, el mensaje será asignado al ítem y este será guardado en la base de datos |

|  |  |
| --- | --- |
| Caso de uso | Dejar ranking |
| Descripción | Usuario deja una puntuación sobre sus preferencias en cierto ítem |
| Gatillante | Petición del usuario |
| Actores | Usuarios |
| Precondiciones | Que no exista un ranking del mismo usuario en el mismo ítem |
| Metas | Asignar ranking al ítem correspondiente |
| Condiciones de fallo | Que el usuario intente dejar ranking sin iniciar sesión |
| Caso esperado | Usuario utilizando su sesión, ingresa a la página de ítem, en la sección designada dejara el ranking deseado, el sistema validara al usuario, si la sesión es validada, el ranking será asignado al ítem y este será guardado en la base de datos. |

|  |  |
| --- | --- |
| Caso de uso | Buscar |
| Descripción | El usuario genera una consulta a la bases de datos para encontrar un ítem especifico |
| Gatillante | Petición del usuario |
| Actores | Usuarios |
| Precondiciones | Conexión correcta entre el sistema y al base de datos |
| Metas | Generar consulta a la base de datos |
| Condiciones de fallo | Que el ítem no exista en la base de datos |
| Caso esperado | El usuario ingresara el nombre del ítem que desea buscar en la sección de búsqueda, esto generara la consulta a la bases de datos, luego el sistema entregara la lista con los resultados encontrados en la base de datos. |

|  |  |
| --- | --- |
| Caso de uso | Modificar perfil |
| Descripción | El usuario cambiara la información privada que utilizo para registrarse |
| Gatillante | Petición del usuario |
| Actores | Usuarios |
| Precondiciones | Que el usuario este registrado y que haya iniciado sesión |
| Metas | Asegurar la información privada del cliente |
| Condiciones de fallo | Que no cumpla con las características para registrarse (por ejemplo cambio de clave, no cumple con los requisitos de las password del sistema), o que utilice exactamente la misma información que estaba anteriormente |
| Caso esperado | El usuario inicie sesión, ingrese a la ventana de modificación de usuario, ingrese sus nuevos datos dentro del sistema, el sistema enviara la información a la base de datos para ser guardada |

|  |  |
| --- | --- |
| Caso de uso | Ver ranking |
| Descripción | El usuario genera una consulta donde muestre todos los ranking realizados por el |
| Gatillante | Petición del usuario |
| Actores | Usuarios |
| Precondiciones | Que el usuario este en su sesión |
| Metas | Realizar una consulta exitosa a la base de datos |
| Condiciones de fallo | Que el usuario no haya dejado ningún ranking en ningún ítem |
| Caso esperado | El usuario este dentro de su sesión, luego entra a la ventana perfil de usuario, donde existirá un botón "ver ranking" que generara la consulta a la base de datos, luego el sistema entregara el listado de los ranking realizados por el usuario |

|  |  |
| --- | --- |
| Caso de uso | Ver comentario |
| Descripción | El usuario genera una consulta donde muestre todos los comentarios realizados por el |
| Gatillante | Petición del usuario |
| Actores | Usuarios |
| Precondiciones | Que el usuario este en su sesión |
| Metas | Realizar una consulta exitosa a la base de datos |
| Condiciones de fallo | Que el usuario no haya realizado ningún comentario |
| Caso esperado | El usuario este dentro de su sesión, luego entra a la ventana perfil de usuario, donde existirá un botón "ver comentarios" que generara la consulta a la base de datos, luego el sistema entregara el listado de los comentarios realizados por el usuario |

|  |  |
| --- | --- |
| Caso de uso | Eliminar usuario |
| Descripción | El usuario desea eliminar su cuenta |
| Gatillante | Petición del usuario |
| Actores | Usuarios |
| Precondiciones | Que el usuario tenga una sesión en el sistema |
| Metas | Eliminar correctamente el usuario del sistema |
| Condiciones de fallo | El usuario no esté dentro del sistema |
| Caso esperado | El usuario en la ventana de perfil, envié petición al sistema de eliminar su usuario, después de validar la sesión y confirmar la petición, se enviara una consulta a la base de datos para eliminar al usuario |

|  |  |
| --- | --- |
| Caso de uso | Crear ítem |
| Descripción | El usuario con rol de administrador genera petición a la base de datos para ingresar un nuevo ítem |
| Gatillante | Petición del usuario con rol de administrador |
| Actores | Usuarios con rol de administrador |
| Precondiciones | El usuario acceda con una sesión de administrador |
| Metas | Asegurar la consulta enviada a la base de datos |
| Condiciones de fallo | Que el ítem que se intenta crear ya exista dentro de la base de datos |
| Caso esperado | El usuario con rol de administrador ingresara a la ventana de creación de ítem donde ingresara todos los datos necesarios para la creación de este, el sistema enviara una consulta a la base de datos donde este será guardado |

|  |  |
| --- | --- |
| Caso de uso | Eliminar ítem |
| Descripción | El usuario con rol de administrador genera petición a la base de datos para eliminar un ítem existente |
| Gatillante | Petición del usuario con rol de administrador |
| Actores | Usuarios con rol de administrador |
| Precondiciones | El usuario acceda con una sesión de administrador |
| Metas | Asegurar la consulta enviada a la base de datos |
| Condiciones de fallo | Que el ítem que se intenta crear no exista dentro de la base de datos |
| Caso esperado | El usuario con rol de administrador ingresara a la ventana de creación de ítem donde ingresara todos los datos necesarios para la creación de este, el sistema enviara una consulta a la base de datos donde este será guardado |

|  |  |
| --- | --- |
| Caso de uso | Solicitar usuario |
| Descripción | El usuario con rol de administrador genera petición a la base de datos para ver los usuarios registrados en el sistema |
| Gatillante | Petición del usuario con rol de administrador |
| Actores | Usuarios con rol de administrador |
| Precondiciones | Conexión correcta entre el sistema y al base de datos |
| Metas | Generar consulta correcta a la base de datos |
| Condiciones de fallo | Que no existan usuarios dentro del sistema |
| Caso esperado | El usuario con rol de administrador ingresara a la ventana de Usuarios , donde existirá un botón que genere la consulta a la base de datos, luego el sistema devolverá una lista con todos los usuarios dentro del sistema. |

# capitulo 5

## 5.1 modelo entidad-relación

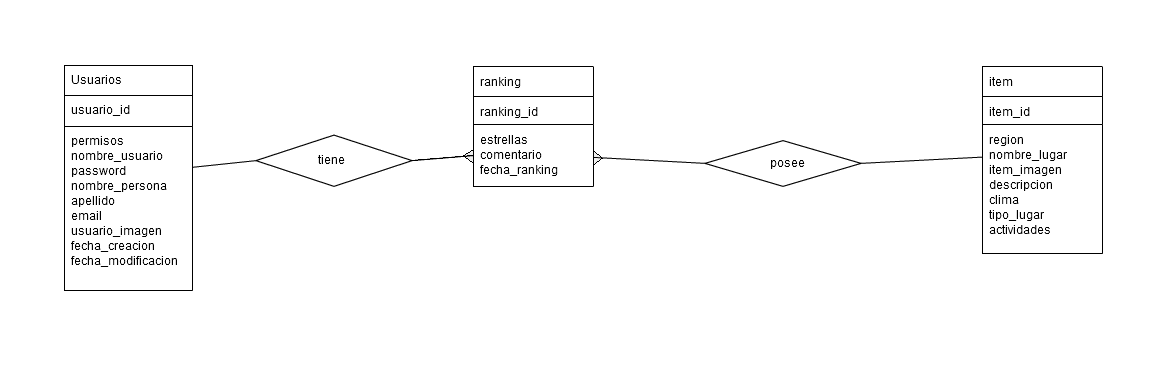


Figura N°3: modelo entidad-relación

## 5.2 Modelo Físico Relacional

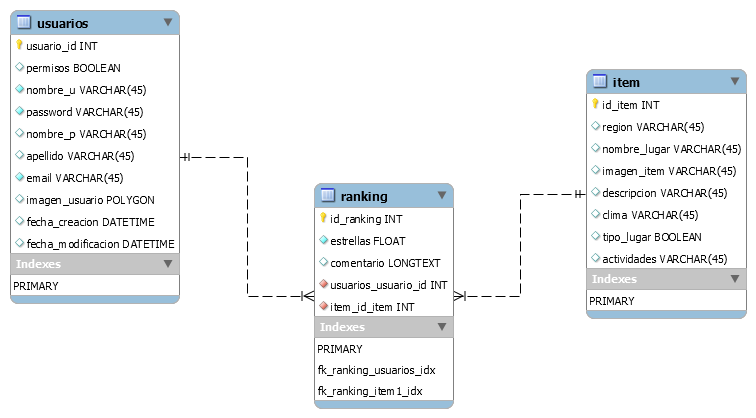


Figura N°4: modelo físico relacional

# Bibliografía

[1] Albín Rodriguez, Antonio Pedro. «Sistema de Recomendación Colaborativo Basado en Algoritmos de Filtrado Mejorado.» Jaén.

[2] Álvarez Vidales, José Borja. «Sistemas de recomendación de servicios turísticos personalizados en el destino turístico inteligente.» Máster Universitario en Dirección y Planificación del Turismo, Málaga, 2014.

[3] Cendejas Valdéz, José Luis. «modelo de desarrollo de software integral colaborativo.» Tesis Doctoral, Puebla, Mexico , 2014.

[4] Cendejas Valdéz, José Luis. «Modelo para el Desarrollo de Software Integral Colaborativo.»

[5] Francesco Ricci, Lior Rokach, Bracha Shapira, Paul B. Kantor. *Recommender Systems Handbook.* Springer, 2011.

[6] González Calero, Pedro Antonio. «Recomendación Social con Aplicaciones a Turismo.» Memoria Máster, Madrid, 2014.

[7] Guzman Álvarez, Cesar Augusto. «Sistema de Recomendación y planificación turística de la ciudad de valencia vía web.» Memoria, Valencia, 2009.

[8] Leiva Olivencia, José Luis. «Realidad aumentada bajo Tecnología Móvil basada en el contexto Aplicada a destinos Turísticos.» Tesis Doctoral, Málaga, 2014.

[9] Martínez Mimbrera, Fransisco Jesús. «Sistema de Recomendación Actualizable y con Gestión de usuarios.» Trabajo de titulo, Jaén, 2008.

[10] Navarro Benites, Karlen. «Recomendación social con aplicaciones a turismo.» Memoria de Master, madrid, 2014.

[11] Pérez Cordón, Luis Gonzaga. «Modelos de Recomendación con falta de información. Aplicaciones al sector trístico.» Memoria de Tesis, Jaén, 2008.

[12] Ruiz Iniesta, Almudena. «Estrategias de recomendación aplicadas a repositorios de recursos educativos.» Proyecto Fin de Máster, Madrid, 2009.

[13] Salazar Molina, Sebastián. «Ingeniería de Software, Metodologías de Desarollo.» Informe, Santiago, 2014.