UNIVERSIDAD POLITÉCNICA SALESIANA
SEDE CUENCA

CARRERA DE INGENIERÍA DE SISTEMAS

Trabajo de titulación previo a la obtención del título de Ingeniera de Sistemas e Ingeniero de Sistemas

PROYECTO TÉCNICO:

"DESARROLLO DE UN SISTEMA DE RECOMENDACIÓN GENERALISTA BASADO EN FILTRADO COLABORATIVO PARA GRUPOS DE USUARIOS MEDIANTE UN ENFOQUE DE AGREGACIÓN DE FACTORES PROBABILÍSTICOS"

AUTORES:

KAREN TATIANA MÉNDEZ ZAPATA

FREDDY RICARDO QUINTUÑA BARBECHO

TUTOR:

REMIGIO ISMAEL HURTADO ORTIZ, PhD.

CUENCA - ECUADOR

2021

CESIÓN DE DERECHOS DE AUTOR

Nosotros, Karen Tatiana Méndez Zapata con documento de identificación N° 0705339646 y Freddy Ricardo Quintuña Barbecho con documento de identidad N° 0104756978, manifestamos nuestra voluntad y cedemos a la Universidad Politécnica Salesiana la titularidad sobre los derechos patrimoniales en virtud de que somos autores del trabajo de titulación: **"DESARROLLO DE UN SISTEMA DE RECOMENDACIÓN GENERALISTA BASADO EN FILTRADO COLABORATIVO PARA GRUPOS DE USUARIOS MEDIANTE UN ENFOQUE DE AGREGACIÓN DE FACTORES PROBABILÍSTICOS"**, mismo que ha sido desarrollado para optar por el título de: *Ingeniera de Sistemas e Ingeniero de Sistemas*, en la Universidad Politécnica Salesiana, quedando la Universidad facultada para ejercer plenamente los derechos cedidos anteriormente.

En aplicación a lo determinado en la Ley de Propiedad Intelectual, en nuestra condición de autores, nos reservamos los derechos morales de la obra antes citada. En concordancia, suscribimos este documento en el momento que hacemos entrega del trabajo final en formato digital a la Biblioteca de la Universidad Politécnica Salesiana.

Cuenca, febrero de 2021.

Karen Tatiana Méndez Zapata

C.I. 0705339646

Freddy Ricardo Quintuña Barbecho

C.I. 0104756978

CERTIFICACIÓN

Yo, declaro que bajo mi tutoría fue desarrollado el trabajo de titulación: **"DESARROLLO DE UN SISTEMA DE RECOMENDACIÓN GENERALISTA BASADO EN FILTRADO COLABORATIVO PARA GRUPOS DE USUARIOS MEDIANTE UN ENFOQUE DE AGREGACIÓN DE FACTORES PROBABILÍSTICOS"**, realizado por Karen Tatiana Méndez Zapata y Freddy Ricardo Quintuña Barbecho, obteniendo el *Proyecto Técnico*, que cumple con todos los requisitos estipulados por la Universidad Politécnica Salesiana.

Cuenca, febrero de 2021.

Remigio Ismael Hurtado Ortiz, Ph.D

C.I. 0104621388

DECLARATORIA DE RESPONSABILIDAD

Nosotros, Karen Tatiana Méndez Zapata con documento de identificación N° 0705339646 y Freddy Ricardo Quintuña Barbecho con documento de identificación N° 0104756978, autores del trabajo de titulación: **"DESARROLLO DE UN SISTEMA DE RECOMENDACIÓN GENERALISTA BASADO EN FILTRADO COLABORATIVO PARA GRUPOS DE USUARIOS MEDIANTE UN ENFOQUE DE AGREGACIÓN DE FACTORES PROBABILÍSTICOS"**, certificamos que el total contenido del *Proyecto Técnico*, es de nuestra exclusiva responsabilidad y autoría.

Cuenca, febrero de 2021.

Karen Tatiana Méndez Zapata

C.I. 0705339646

Freddy Ricardo Quintuña Barbecho

C.I. 0104756978

AGRADECIMIENTOS.

Primeramente, agradezco a Jehova, quien ha sido mi fortaleza en mis momentos de debilidad, quien me acompañado en cada paso que he dado, por guiarme y darme sabiduría y amor para culminar esta etapa de mi vida.

Por otro lado, agradezco, a mi madre por apoyarme incondicionalmente, por los valores que me ha enseñado y por la oportunidad de brindarme una excelente educación, por mostrarme como mujer a luchar por lo que deseo y nunca darme por vencida, a mis hermanas por estar siempre a mi lado, y a mi esposo que ha estado desde que comencé en esta travesía, por todo su apoyo y amor incondicional que me ha brindado.

Finalmente, y no menos importante expresar mis más sinceros agradecimientos al Ing. Remigio Hurtado, tutor del proyecto, quien ha demostrado su dedicación e interés, brindándonos su apoyo y conocimiento para poder culminarlo.

Karen Tatiana Méndez Zapata.

AGRADECIMIENTOS.

Agradezco a Dios por bendecirme la vida y la salud, por guiarme a lo largo de mi vida.

Gracias a mis padres: Miguel y Edelina, por confiar y creer en mis expectativas, por los consejos, valores y principios que me han inculcado. Por ser el apoyo y fortaleza en aquellos momentos de dificultad y de debilidad.

De manera especial, agradezco al Ing. Remigio Hurtado Ortiz tutor del proyecto de grado, por su dirección, paciencia, entrega y valiosos consejos que me permitieron alcanzar los objetivos de esta tesis.

Freddy Ricardo Quintuña Barbecho

DEDICATORIA.

Este triunfo es dedicado a las mujeres que nos vemos luchando día a día en medio de adversidades y muchas desventajas, que a pesar de tener el mismo nivel educativo no nos ven como una igual, a esas mujeres que son las primeras en levantarse y ultimas en acostarse para atender su hogar, a las mujeres que muchas veces terminan siendo padre y madre para sus hijos y aun si no suelen reconocer su valor.

Dedico este triunfo a ti mujer que no importa la condición ni el estatus social y aun así nunca te rindes.

Karen Tatiana Méndez Zapata.

El presente trabajo le dedico a Dios, por darme el conocimiento y la fuerza necesaria para terminar este proceso de obtener uno de mis anhelos más deseados.

A mis padres, por su dedicación, amor, trabajo y sacrificio en todos estos años, gracias a ustedes he logrado llegar hasta aquí y convertirme en la persona de bien que soy. Son los mejores padres. A mis hermanos (as) por estar siempre presentes, acompañándome y por el apoyo moral, que me brindaron a lo largo de esta etapa de mi vida.

A mi esposa por el apoyo incondicional durante el trabajo de titulación.

A todas las personas que me han apoyado y han hecho que el trabajo se realice con éxito en especial a aquellos que nos abrieron las puertas y compartieron sus conocimientos.

Freddy Ricardo Quintuña Barbecho.

Índice

Índice de Figuras **10**
Índice de Tablas **11**
RESUMEN **12**
ABSTRACT **13**
CAPÍTULO 1: INTRODUCCIÓN **14**
1.1. Problema 14
1.2 Justificación 14
1.3 Fundamentos de sistemas de recomendación 15
1.3.1 Generalidades 15
1.3.2 Clasificación de sistemas de recomendación 15
1.3.3 Filtrado Colaborativo 16
1.3.4 Sistemas de recomendación para grupos de usuarios 16
1.3.5 Clasificación de sistemas de recomendación para grupos de usuarios 17
1.3.6 Clustering aplicado a sistemas de recomendación 17
1.4 Tecnologías 18
1.4.1 Base de datos no relacional MongoDB 18
1.4.2 Angular 19
1.4.3 Ionic 19
1.4.4 NodeJS 19
1.4.5 Java EE 19
1.5 Trabajo relacionado 20
1.6 Objetivos 21
1.6.1 General 21
1.6.2 Específicos 22
1.7 Alcance y Limitaciones 22
1.8 Descripción General del Documento 22
CAPITULO 2: ARQUITECTURA, DISEÑO Y DESARROLLO DEL SISTEMA DE RECOMENDACIÓN PARA GRUPOS DE USUARIOS **23**
2.1 Arquitectura del sistema 23
2.1.1 Capa de Presentación 24
2.1.2 Capa de Negocio 25
2.1.3 Capa de Datos 26
2.1.4 Servidores 30
2.2 Diseño del Sistema de Recomendación a Grupos 30
2.3 Diseño del sistema Móvil y Web 33

2.3.1 Mockup Móvil 34
2.3.2 Mockup Web 36
2.3.3 Diagramas Casos de Usos 39
2.4 Desarrollo del sistema web 42
2.5 Desarrollo del sistema móvil 42
2.6 Integración y despliegue en la nube 43
CAPITULO 3: EXPERIMENTOS, VERIFICACIÓN DE REQUERIMIENTOS Y PRUEBAS FUNCIONALES 48
3.1 Calidad del sistema de recomendación 48
3.2 Verificación del sistema 52
3.3 Pruebas funcionales del sistema web 54
3.4 Pruebas funcionales del sistema móvil 60
CONCLUSIONES 76
RECOMENDACIONES 77
TRABAJO FUTURO 77

Índice de Figuras

Figura 1 Propuesta de Arquitectura 23
Figura 2 Capa de Presentación GRS 24
Figura 3 Capa de Negocio - GRS 25
Figura 4 Diseño Base de Datos GRS Cloud – Documentos 28
Figura 5 Diseño Base de Datos GRS Cloud - Documentos Simples 29
Figura 6 Diseño del Sistema de Recomendación a Grupos 31
Figura 7 Mockup Móvil Login 34
Figura 8 Mockup Móvil Registro 35
Figura 9 Mockup Móvil Home - Información del Item 35
Figura 10 Mockup Móvil Comentarios - Crear nuevo Grupo 36
Figura 11 Mockup Web – Login 36
Figura 12 Mockup Web - DASHBOARD 37
Figura 13 Mockup Web - Nuevo Libro 37
Figura 14 Mockup Web - Lista de Libros 38
Figura 15 Mockup Web - Usuario 38
Figura 16 Diagrama Caso de Uso: Administrador de Items 39
Figura 17 Diagrama de caso de uso: Gestión de Reportes 40
Figura 18 Diagrama Caso de Uso – Registro 40
Figura 19 Diagrama Caso de Uso - Aplicación Móvil 41
Figura 20 Distribución de componentes despliegue en la nube 44
Figura 21 Versión Angular 47
Figura 22 Comparación de técnicas de reducción de dimensionalidad. Figura tomada de (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020) 50
Figura 23 Distribución de usuarios en clusters. Figura tomada de (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020) 51
Figura 24 Calidad de predicción con RMSE. Figura tomada de (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020) 51
Figura 25 Calidad de recomendación con F1. Figura tomada de (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020) 52

Índice de Tablas

Tabla 1 Especificaciones caso de uso Administrador Web 39
Tabla 2 Especificaciones caso de uso Gestión de Reportes 40
Tabla 3 Especificaciones caso de uso: Registro 41
Tabla 4 Especificaciones caso de uso: Aplicación Móvil 41
Tabla 5 Características principales del Dataset usado en el proyecto 48
Tabla 6 Cross-validation values used in the experiments 48
Tabla 7 Lista de Verificación Sistema Web 53
Tabla 8 Lista de Verificación Sistema Móvil 53
Tabla 9 Prueba Funcional Sistema Web - Login 54
Tabla 10 Prueba Funcional Sistema Web - Crear y Editar Libro 55
Tabla 11 Prueba Funcional Sistema Web - Dashboard 57
Tabla 12 Prueba Funcional Sistema Web - Administrar Usuario 58
Tabla 13 Prueba Funcional Sistema Móvi - Login y Registrol 60
Tabla 14 Prueba Funcional Sistema Móvil - Perfil 64
Tabla 15 Prueba Funcional Sistema Móvil - Crear Grupo 65
Tabla 16 Prueba Funcional Sistema Móvil - Home 67
Tabla 17 Prueba Funcional Sistema Móvil - Grupos Recomendados 69
Tabla 18 Prueba Funcional Sistema Móvil - Comentarios 72
Tabla 19 Prueba Funcional Sistema Móvil - Rating 74

RESUMEN

En la actualidad una de las necesidades del ser humano es tomar una correcta elección en la toma de decisiones para ello es necesario implementar herramientas tecnológicas inteligentes. La información cada día crece a pasos agigantados, se vuelve cada vez más difícil encontrar información real o precisa. Dentro de las organizaciones depender de personas expertas en áreas específicas se vuelve menos rentable para una toma de decisiones futuras, no siempre los expertos estarán disponibles y no siempre tendrán el conocimiento actualizado. En la actualidad las organizaciones basan su stock de productos dependiendo de las ventas que se produzca cada cierto periodo analizado, ¿es correcto afirmar que el producto más vendido es el mejor? Ahora vamos a pensar como cliente al realizar una compra de algún producto, siendo más minuciosos vamos a tener varias dudas: ¿marca, color, precio?, ¿quién compró el producto?, ¿es recomendable comprar?, ¿en verdad cumple con las necesidades que busco?, ¿cómo seleccionar entre la gran variedad de productos similares?

Debido a las necesidades de la organización (en tener un stock correcto) y del cliente (en realizar una compra correcta y en menor tiempo posible), este proyecto crea una solución o producto que integra tres componentes que conforman un sistema de recomendación para grupos con el objetivo de brindar una solución de administración para la organización y recomendar productos para los clientes, además el sistema identifica grupos y recomienda productos con las mejores votaciones dependiendo del grupo conformado.

El primer componente es la lógica de negocio o backend del sistema recomendador para grupos, encargado de realizar el procesamiento de la información, entrenamiento y recomendación de los grupos para cada cliente. Se aplica los siguientes conceptos: reducción de dimensionalidad con un modelo bayesiano (trabajar con una cantidad exuberante de datos es más pesado que trabajar con factores de ítems y usuarios), técnicas de datamining (clustering mediante kmeans), enfoque agregación de factores de usuarios del grupo (para las predicciones grupales). El segundo componente es la aplicación móvil para que los clientes puedan visualizar los ítems recomendados, gestionar su perfil, crear y visualizar sus grupos recomendados. El tercer componente es la aplicación web para la administración y graficas de grupos conformados. Los tres componentes se comunican a través de servicios web. Con estos componentes la organización puede tomar una mejor decisión con base a las gráficas mostradas en la página de administración, y el cliente puede seleccionar productos en menos tiempo, y saber con qué clientes tiene afinidad con base a los grupos recomendados.

ABSTRACT

Currently, one of human beings' necessities is to make a correct choice in decision-making, therefore it is necessary to implement smart technological tools. Information grows every day at a tremendous speed; it becomes increasingly difficult to find real or accurate information. Within organizations, relying on experts in specific areas becomes less profitable for future decision-making, experts will not always be available, and they will not always have up-to-date knowledge. In today's day, organizations base their stock products depending on sales that occur every certain period analyzed, is it correct to say that the best-selling product is the best? Now we are going to think as a client when making a purchase of a product by being more meticulous we are going to have several doubts: brand, color, price, who bought the product? Is it recommended to buy? Does it really meet the needs I am looking for? How do I select from the wide variety of similar products?

Due to the needs of the organization (to have a correct stock) and the client (to make a correct purchase and in the shortest time possible), this project creates a solution or product that integrates three components that make up a recommendation system for groups with the objective of providing a management solution for the organization and recommend products for customers; in addition, the system identifies groups and recommends products with the best votes depending on the group formed.

The first component is the business logic or backend recommender system for groups which in charge of processing information, training and recommending the groups for each client. The following concepts are applied: reduction of dimensionality of Bayesiano model (working with a full amount of data is heavier than working with the item' factors and users), techniques of datamining (clustering through kmeans), focusing in adding factors of group' users (for group predictions). The second component is the mobile application so that clients can view recommended items, manage their profile, create and view their recommended groups. The third component is the web application for the administration and graphics of groups formed. All three components communicate through web services. With these components, the organization can make a better decision based on the graphs shown on the administration page, and the customer can select products in less time, and know with which clients they have a relationship based on the recommended groups.

CAPÍTULO 1: INTRODUCCIÓN

1.1. Problema

El fácil acceso al internet y la implementación de servicios en la nube, hoy en día permite que exista gran variedad de información disponible en la web, lo cual deriva en un problema conocido como “sobrecarga de información” o “infoxicación” (Zhu, Hurtado, Bobadilla, & Ortega, 2018), (Ortega, Bobadilla, Guitiérrez, Hurtado , & Li, 2018). La mayor capacidad en las unidades de almacenamiento y el constante aumento de velocidad en las redes de comunicación permiten que la información almacenada en el universo digital se incremente a diario rápidamente, permitiendo que cualquier persona puede introducir información en la web de modo online implícitamente o explícitamente. Toda la información almacenada se puede reutilizar por empresas para mejorar sus negocios, implementando estrategias de negocio y estrategias de ventas. El problema se centra en cómo reutilizar la información para ganar clientes para el beneficio de la empresa. La infoxicación de la información provoca que los usuarios terminen saturados de ella, así internet termina bombardeando con tanta publicidad, elecciones, anuncios que provoca que los usuarios terminan confundidos y sin poder tomar una decisión clara, concisa sobre lo que desea ver, escuchar, leer, etc.

1.2 Justificación

Según (Bobadilla, Bojorque, Hernando, & Hurtado, 2017), en los últimos años la minería de datos y los sistemas de recomendación (RS, por sus siglas en inglés) han demostrado ser un apoyo a las empresas con el fin de tratar la gran cantidad de información, analizar datos y crear estrategias de negocio, tanto para ganar clientes, como para mejorar sus ingresos. Empresas como Amazon, YouTube, Netflix, Spotify y Facebook, desarrollan sistemas de recomendación para ofrecer a sus suscriptores ofertas como parte de las estrategias de negocio.

Los sistemas de recomendación han sido usados en diferentes ámbitos; la mayoría de los trabajos se centran en recomendaciones individuales, es decir, en sistemas que se enfocan en recomendar un producto a un usuario específico; sin embargo, queda por explorar el campo de los sistemas de recomendación a grupos de usuarios. Se pueden tener dos tipos de grupos de usuarios: grupos con distintas características (heterogéneos) y grupos con similares características (homogéneos). Un ejemplo de un grupo heterogéneo es una familia que desea por ejemplo un tour por parte de una agencia de viajes. Este tour debe satisfacer los gustos de todo el grupo. Un ejemplo de un grupo homogéneo es una comunidad de lectores que han sido agrupados por sus características similares. Este trabajo se centra en proporcionar un sistema de recomendación generalista basado en filtrado colaborativo para grupos de usuarios, es decir que pueda servir para grupos

heterogéneos y grupos homogéneos, y en distintos dominios, como comida en restaurantes, música, películas, lugares turísticos, etc (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020).

En el trabajo de (Ortega, Hurtado, Bobadilla, & Bojorque , 2018) se brinda una clasificación y el estado del arte de los enfoques reconocidos para recomendación a grupos de usuarios, entre estos enfoques están: a) enfoque de agregación de preferencias, b) enfoque de agregación de vecinos, c) enfoque de agregación de predicciones, y d) enfoque de agregación de recomendaciones. En este trabajo proponemos un enfoque novedoso para recomendación a grupos mediante un nuevo nivel de agregación de factores (información modelada mediante reconocimiento de patrones) de los usuarios

1.3 Fundamentos de sistemas de recomendación

1.3.1 Generalidades

Los sistemas de recomendación son herramientas de software y técnicas que proveen sugerencias para ítems que probablemente son de la preferencia de un usuario particular. Las sugerencias están relacionadas con procesos de decisión como qué ítem adquirir, qué película ver, qué música escuchar o qué libro leer. Un RS es básicamente un filtro de información, es decir, un algoritmo que recolecta la información más relevante para un usuario y elimina la menos relevante (Bobadilla, Bojorque, Hernando, & Hurtado, 2017).

1.3.2 Clasificación de sistemas de recomendación

Un RS puede ser clasificado de acuerdo con su tipo de filtrado de información:

1.3.2.1 Filtrado basado en contenido

Se basan en categorizar los ítems a recomendar, proporcionando resultados que tengan características similares a otros que han sido valorados anteriormente por el usuario o que tengan características similares a las que demanda el usuario (Leiva Olivencia & Vargas Pérez, 2016)

1.3.2.2 Filtrado Demográfico

Este tipo de filtrado clasifica a los usuarios en función de características demográficas como (edad, sexo, idioma, religión, profesión, estado civil, etc.) (Leiva Olivencia & Vargas Pérez, 2016). Aquellos que disponen de esa información la aprovechan. Un ejemplo, un RS para la publicidad en redes sociales.

1.3.2.3 Filtrado Colaborativo

Estos sistemas de recomendación presentan elementos que le han gustado a otros usuarios con gustos similares, con este propósito, calculan la similaridad entre usuarios. El filtrado colaborativo se basa en los votos que los distintos usuarios realizan sobre diferentes ítems, de tal manera que al contar con suficiente información almacenada en el sistema es posible emitir recomendaciones a los usuarios según la información proporcionada por otros usuarios de características similares (Bobadilla, Bojorque, Hernando, & Hurtado, 2017).

1.3.2.4 Filtrado Híbrido

Se trata de modelos de recomendación que combinan diferentes tipos de técnicas. Este filtrado intenta evitar las limitaciones existentes en los otros modelos. Se mezclan algunos de los tres filtros mencionados anteriormente para realizar recomendaciones, incluso se combina con alguna otra técnica de inteligencia artificial como puede ser la lógica difusa o la computación evolutiva (Bobadilla, Bojorque, Hernando, & Hurtado, 2017)

1.3.3 Filtrado Colaborativo

Las técnicas de Filtrado Colaborativo (*CF, por sus siglas en inglés)*, son muy utilizadas en los Sistemas de Recomendación, ya que en la literatura demuestra que es el tipo de filtrado más utilizado por su calidad de predicciones y recomendaciones (Galán Nieto, 2007) porque utilizan las valoraciones de los usuarios sobre ciertos elementos del conjunto total para predecir valoraciones en el resto de los elementos y recomendar los de mayor valoración predicha. Por ello el tipo de filtrado que se utilizará en este proyecto es el filtrado colaborativo.

Está técnica se basa en los gustos similares de las personas, según (TORRES RUDLOFF, 2017) al tener gustos similares pueden entregar mejores y más precisas recomendaciones sobre productos. Como principal representante de CF está ***User*-based**, que a través de una métrica especifica calcula la similaridad y retorna los ***k*** usuarios más cercanos.

1.3.4 Sistemas de recomendación para grupos de usuarios

Los sistemas de recomendación para usuarios han tenido un gran avance en la investigación y en la implementación de técnicas, descuidando otro enfoque de los RS qué son los Sistemas de Recomendación para Grupos de usuarios (*GRS, por sus siglas en inglés)* que permiten identificar características, personalidades, patrones de grupos de usuarios y de esta manera se puede dar recomendaciones al grupo y no a un usuario específico.

Para (Kyoungsoo , Inbae , Jongtae , & Jaesoo , 2019), las preferencias de los miembros del grupo pueden ser similares o diferentes. En todo caso, se requiere obtener la preferencia común de los miembros del grupo, aliviar el conflicto entre los miembros del grupo y hacer que la recomendación tenga la mayor precisión para satisfacer las necesidades de todos los miembros del grupo.

Los GRS representan un desafío por su complejidad de identificar las preferencias de los usuarios y generar recomendaciones que compensen los gustos de todos los usuarios que integran al grupo sin dejar los gustos de ningún miembro.

1.3.5 Clasificación de sistemas de recomendación para grupos de usuarios

Según (Hurtado Ortiz, Recomendación a grupos de usuarios usuando conceptos de singularidades, 2020) un sistema puede generar recomendaciones a grupos utilizando tres enfoques diferentes para predecir votos:

1. **Construcción de modelos de preferencias grupales**: se realiza la predicción de los votos faltantes para cada grupo utilizando el modelo. Enfoque también conocido como agregación de preferencias o perfiles de usuarios del grupo, o enfoque relacionado a usuarios.
2. **Agregación de recomendaciones:** se hace la predicción de los votos para los ítems no calificados por cada usuario y luego la fusión de las recomendaciones individuales hechas para los miembros de un grupo.
3. **Agregación de predicciones**: se agregan las predicciones creadas para cada usuario en una preferencia de grupo.

1.3.6 Clustering aplicado a sistemas de recomendación

El Clustering es una técnica de minería de datos que permite agrupar y generar colecciones de objetos de datos denominadas clúster, de tal forma que los objetos de un clúster son similares entre sí dentro del mismo grupo, pero diferentes a los objetos de otros grupos. Existen varios tipos de clustering, entre los cuales están: aglomerativos, divisivos, exclusivos, probabilísticos, jerárquicos, etc. (Inga Chalco, Bojorque Chasi, & Hurtado Ortiz, 2018)

Según (TORRES RUDLOFF, 2017) el procesamiento de clustering se divide en las siguientes etapas:

1. Extraer y seleccionar las características más representativas del conjunto de datos original.

2. Diseñar el algoritmo de agrupación de acuerdo a las características del problema.
3. Evaluar el resultado de la agrupación y validar el algoritmo.
4. Entregar una explicación práctica acerca del resultado obtenido.

Los algoritmos de recomendación basados en clustering han demostrado ser eficientes en data set reales porque pueden ser utilizados como etapa previa en el preprocesamiento de los datos, agrupándolos por usuarios/ítems en clústers. El objetivo de aplicar clustering en sistema de recomendación a grupos, es que los usuarios e ítems puedan ser recuperados y aprovechados para generar recomendaciones con resultados más precisos (TORRES RUDLOFF, 2017).

1.4 Tecnologías

1.4.1 Base de datos no relacional MongoDB

Las bases de datos NoSQL a diferencia de las bases de datos relacionales no tienen un identificador único “id”, que permite la relación entre un conjunto de datos “tablas”, para (Amazon, s.f.) estos tipos de base de datos están optimizados específicamente para aplicaciones que requieren grandes volúmenes de datos, baja latencia y modelos de datos flexibles, lo que se logra mediante la flexibilización de algunas restricciones de coherencia de datos en otras bases de datos.

MongoDB es una base de datos orientada a documentos. Esto quiere decir que, en lugar de guardar los datos en registros, guarda los datos en documentos. Estos documentos son almacenados en BSON, que es una representación binaria de JSON. (COBA DE LA TORRE & GÓMEZ GÓMEZ, 2014)

1.4.1.1 Ventajas de la base de datos no relacional

- Una de las ventajas más importantes frente a las bases de datos relaciones, es que no se sigue un esquema.
- Base de datos NoSQL es sumamente útil en proyectos escalables.
- Optimización de consultas.
- Realizar cambios en el esquema sin detener la base de datos.
- No necesita una máquina con muchos recursos para ser ejecutada.

1.4.1.2 ¿Por qué trabajar con una base de datos no relacional?

.

Porque una base de datos no relacional es altamente flexible, escalable y robusta, se basada en documentos; la información que se genere se guarda en documentos y no en registros como se haría con una base de datos tradicional.

BSON representa los documentos JSON de forma binaria y permite la representación de tipos de datos que no hacen parte de la especificación JSON, como datos de tipo fecha y datos de tipo binario (arreglo de bytes). (Galvis, 2015).

Debido a estas ventajas, en este proyecto se ha decido trabajar con una base de datos NoSQL MongoDB por su flexibilidad y escalabilidad que nos permitirá trabajar con el dataset de ***MovieLens*** (GroupLens, 2000)***.***

1.4.2 Angular

Angular es una plataforma y un marco para crear aplicaciones cliente de una sola página utilizando HTML y TypeScript, está escrito en typescript. La arquitectura de una aplicación Angular se basa en ciertos conceptos fundamentales. Los bloques de construcción básicos son NgModules, que proporcionan un contexto de compilación para los componentes. NgModules recopila código relacionado en conjuntos funcionales; una aplicación angular está definida por un conjunto de NgModules. Una aplicación siempre tiene al menos un módulo raíz que permite el arranque y, por lo general, tiene muchos más módulos de funciones (Angular, s.f.).

1.4.3 Ionic

Es un Framework que permite crear aplicaciones móviles para Android y IOS de alta calidad y rendimiento utilizando tecnologías web (HTML, CSS y JavaScript), con integraciones para marcos populares como Angular y React.

Ionic Framework se centra en el frontend UX y la interacción UI de una aplicación: controles UI, interacciones, gestos, animaciones. (IONIC FRAMEWORK, 2020)

1.4.4 NodeJS

NodeJS es una plataforma basada en tiempo de ejecución de JavaScript de Chrome para crear fácilmente aplicaciones de red rápidas y escalables. NodeJS utiliza un modelo de E/S sin bloqueo controlado por eventos que lo hacen liviano y eficiente, perfecto para aplicaciones en tiempo real con uso intensivo de datos que se ejecutan en dispositivos distribuidos. NodeJS ofrece una biblioteca amplia de módulo de JavaScript que permiten simplificar el desarrollo de aplicaciones web que usen esta tecnología. (OPEN JS FUNDATION , s.f.)

1.4.5 Java EE

La tecnología Java es tanto un lenguaje de programación como una plataforma. El lenguaje de programación Java es un lenguaje orientado a objetos de alto nivel. La plataforma Java EE está construida sobre la

plataforma Java SE. La plataforma Java EE proporciona una API y un entorno de ejecución para desarrollar y ejecutar aplicaciones de red seguras, escalables, confiables y de múltiples niveles (Oracle, s.f.).

1.5 Trabajo relacionado

Actualmente existe un abanico de investigaciones que tratan el tema de sistemas de recomendación, filtrado colaborativo y sistemas de recomendación a grupos, a continuación se destacan los trabajos más relevantes del estado del arte; por ejemplo el trabajo de (Ghobar, 2017) se caracteriza por estar basada en la generación de dos filtros, uno colaborativo y otro por contenido permitiendo usar los datos ya existentes sobre las preferencias de los usuarios, y aplicar técnicas de Machine Learning para desarrollar un sistema para hacer recomendaciones, sugiriendo nuevos ítems ajustados a los gustos de los usuarios a partir de perfiles de usuario o de productos generados con este fin.

En el trabajo de (Jesús, 2016) se propone un sistema diseñado principalmente para generar paquetes turísticos a partir de la opinión o gustos de los distintos usuarios que utilizan el mismo, y de este modo, finalmente, realizar un análisis de cuán eficiente es el sistema a la hora de recomendar paquetes a cada uno de los usuarios luego de haber puntuado cada uno de los ítems que lo componen.

En (Zhu, Hurtado, Bobadilla, & Ortega, 2018) se genera recomendaciones eficientes basándose en el framework CF4J desarrollado en JAVA que permite incorporar información estructural e información no estructural para calcular las similaridad entre usuarios.

Para los autores (Ortega, Hurtado, Bobadilla, & Bojorque , 2018) la recomendación a un grupo de usuarios es un gran desafío para el filtrado colaborativo. Las recomendaciones a grupos de usuarios surgen de la comodidad de poder recomendar un grupo de usuarios sobre productos o servicios que satisfagan al grupo, los autores proponen la medida de similitud SMGU, diseñada para recomendaciones de filtrado colaborativo para grupos de usuarios. Esta medida de similitud combina información numérica y no numérica. La información numérica se pondera atendiendo a la singularidad de calificación de los miembros del grupo.

(Bobadilla, Bojorque, Hernando, & Hurtado, 2017) indica que una matriz dispersa en un sistema de recomendación con filtrado colaborativo hace difícil comparar elementos, obtener modelos precisos, obtener predicciones precisas y agrupar adecuadamente los elementos, por ello proponen el uso de un método de factorización matricial no negativa bayesiana (BNMF) para mejorar los resultados actuales de agrupamiento

en el área de filtrado colaborativo; otro tipo de sistema de recomendación son los basados en documentos científicos.

(Ortega, Bobadilla, Guitiérrez, Hurtado , & Li, 2018) se centran en explorar la información de citas y referencias incluida en cada artículo de investigación y también las listas de coautores. De esta forma, se puede abordar la recomendación de artículos relacionados e incluso de autores relacionados.

(Bojorque, Hurtado, & Inga, 2019) muestran el comportamiento de las métricas de similitud en datos escasos para sistemas de recomendación. La agrupación en clústeres en RS es una técnica importante para realizar grupos de usuarios o elementos con el propósito de recomendaciones de personalización y optimización.

(Hurtado Ortiz, Bojorque Chasi, & Inga Chalco, 2018) muestra los resultados de un sistema de recomendación (RS) que sugiere paquetes de elementos a un usuario o una comunidad de usuarios. Hoy en día, existen varios RS que realizan sugerencias de un artículo único considerando las preferencias de un usuario. Sin embargo, estos RS no son escalables y, en ocasiones, las sugerencias que hacen están lejos de las preferencias del usuario.

(Inga Chalco, Bojorque Chasi, & Hurtado Ortiz, 2018) indican que, en la actualidad, los Sistemas de Recomendación (RS) que utilizan Filtrado Colaborativo (CF) son objetos de interés y desarrollo. CF permite a RS tener un filtrado escalable, variar métricas para determinar la similitud entre usuarios y obtener recomendaciones muy precisas al utilizar datos dispersos.

Este trabajo se enfoca en sistemas de recomendación a grupos de usuarios y para ello se incorpora un modelo bayesiano para reducir la dimensionalidad de los datos y mediante un enfoque de agregación de predicciones se guía este modelo a la recomendación para grupos de usuarios conformados por el usuario en la aplicación y grupos detectados mediante clustering.

1.6 Objetivos

1.6.1 General

Desarrollar un sistema de recomendación generalista basado en filtrado colaborativo para grupos de usuarios mediante un enfoque de agregación de factores probabilísticos.

1.6.2 Específicos

OE1. Estudiar y conocer los fundamentos de los sistemas de recomendación basados en filtrado colaborativo y los enfoques de recomendación para grupos de usuarios.

OE2. Diseñar y desarrollar módulos de a) reducción de dimensionalidad utilizando un modelo probabilístico bayesiano para aliviar el problema de dispersión de datos en el filtrado colaborativo, b) clustering para detectar automáticamente grupos de usuarios, c) método de recomendación a grupos mediante un enfoque de factores de usuarios de grupos.

OE3. Diseñar, desarrollar e implementar una aplicación web desplegada sobre una plataforma en nube privada que permita la gestión de usuarios, votos e ítems.

OE4. Diseñar y desarrollar una aplicación móvil que permita la interacción con los usuarios del sistema recomendador para grupos de usuarios.

OE5. Diseñar y ejecutar un plan de experimentación que permita validar el sistema de recomendación para grupos de usuarios.

1.7 Alcance y Limitaciones

Se pretende realizar un sistema de recomendación para grupos de usuarios generalista. Sim embargo, por temas prácticos tiene como alcance integrar un GRS en un sistema informático para recomendar ítems “Libros” a grupos de usuarios, y contará con ciertas limitaciones como el dataset de usuarios e ítems con el fin de realizar el entrenamiento del GRS.

1.8 Descripción General del Documento

Lo que sigue del documento está estructurado de la siguiente manera:

a. Arquitectura, diseño y desarrollo del GRS: en esta sección se explicará la arquitectura, diseño y desarrollo del sistema informático, y el modelo del sistema de recomendación para grupos de usuarios.
b. Experimentos y pruebas funcionales: en esta sección se presentará los resultados del modelo GRS para demostrar su precisión en las recomendaciones, y finalmente se presentará las pruebas funcionales del GRS para la recomendación a grupos de usuarios conformados manualmente, y para la recomendación a grupos de usuarios descubiertos mediante técnicas de clustering.
c. Finalmente, se presentará las conclusiones y trabajo futuro.

CAPITULO 2: ARQUITECTURA, DISEÑO Y DESARROLLO DEL SISTEMA DE RECOMENDACIÓN PARA GRUPOS DE USUARIOS

En este capítulo se presenta primeramente la arquitectura del sistema con la explicación de cada capa y componente. Posteriormente, se explica el diseño del modelo del sistema de recomendación. Luego, se presenta la documentación del diseño, desarrollo y despliegue del sistema web y móvil.

2.1 Arquitectura del sistema

En los últimos años el cambio en la arquitectura del desarrollo de software ha evolucionado desde la programación estructurada a una programación basada en eventos y actualmente la programación orientada a objetos, esto conlleva a la programación N Capas (Moquillaza Henríquez, Vega Huerta, & Guerra Grados, 2010). El objetivo de esta arquitectura es separar la lógica de negocios de la lógica de diseño, esto permite separar el código en varios niveles y mantener una estructura más ordenada, si una capa se ve afectada, se puede trabajar sobre ella si afectar a las otras capas.

En este trabajo proponemos el desarrollo e implementación de un sistema de recomendación generalista basado en filtrado colaborativo para grupos de usuarios mediante un enfoque de agregación de factores probabilísticos.

En la figura 2 se presenta la arquitectura del sistema mediante una vista por capas.

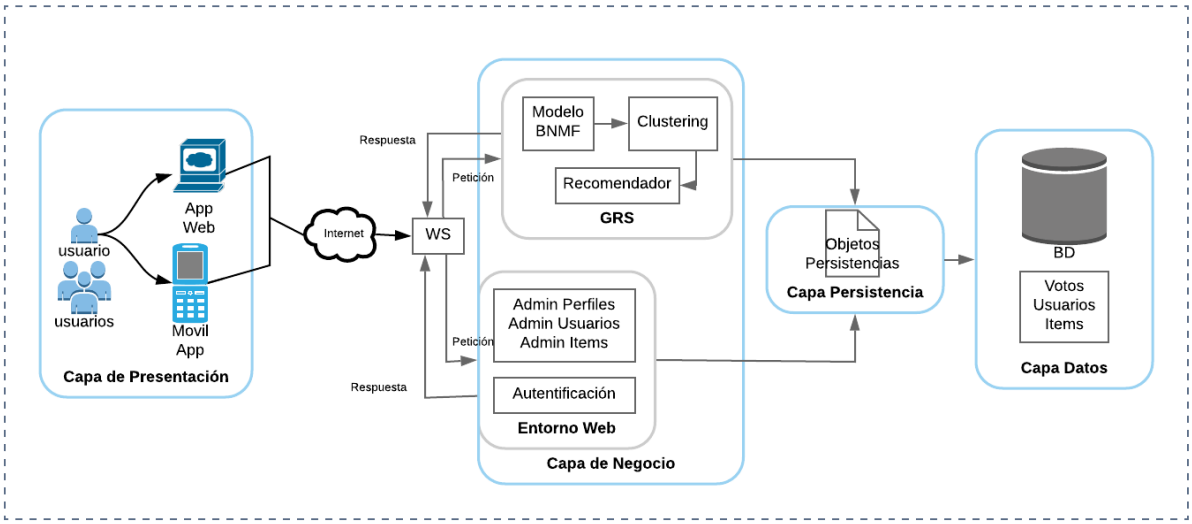


Figura 1 Propuesta de Arquitectura

A continuación, se explica cada una de las capas, en primer lugar, se explica la capa de presentación ya sea en un entorno web o móvil, en segundo lugar, la capa de negocio que incorpora el módulo GRS y la lógica de backend para la

web y para el móvil. Luego, se explica el diseño de la capa de datos y la gestión de la base de datos no relacional. Finalmente, se explica los servidores necesarios para el sistema.

2.1.1 Capa de Presentación

La capa de presentación permite la interacción del administrador contra la aplicación web y el usuario “estudiante, docente” contra la aplicación móvil.

Figura 2 Capa de Presentación GRS

2.1.1.1 Aplicación Móvil

La implementación de una aplicación móvil desarrollada en el framework ionic que permite el desarrollo de aplicaciones móvil híbridas, y el framework angular para crear y mantener aplicaciones en una sola página, por lo que permite mantener separado el Backend y Frontend para tener un mejor control de código que se está implementando, está aplicación tendrá las siguientes funciones: crear grupo, crear comentarios, calificar un ítem, revisar información del ítem, revisar grupos recomendados, ver lista de ítems recomendados.

2.1.1.2 Aplicación Web

La aplicación web, la cual está desarrollada en el framework de desarrollo Angular para JavaScript creado por Google, y Bootstrap que es una biblioteca de multiplataformas de código abierto para la creación de sitios y aplicaciones web, está aplicación tendrá las siguientes funciones: crear usuarios, crear nuevo ítem, editar ítem, revisar gráficas de ítems recomendados.

2.1.2 Capa de Negocio

La capa de negocios contiene la lógica principal de procesamiento, entrenamiento e integración del sistema de recomendación a grupos de usuarios, y la lógica del backend web y móvil.

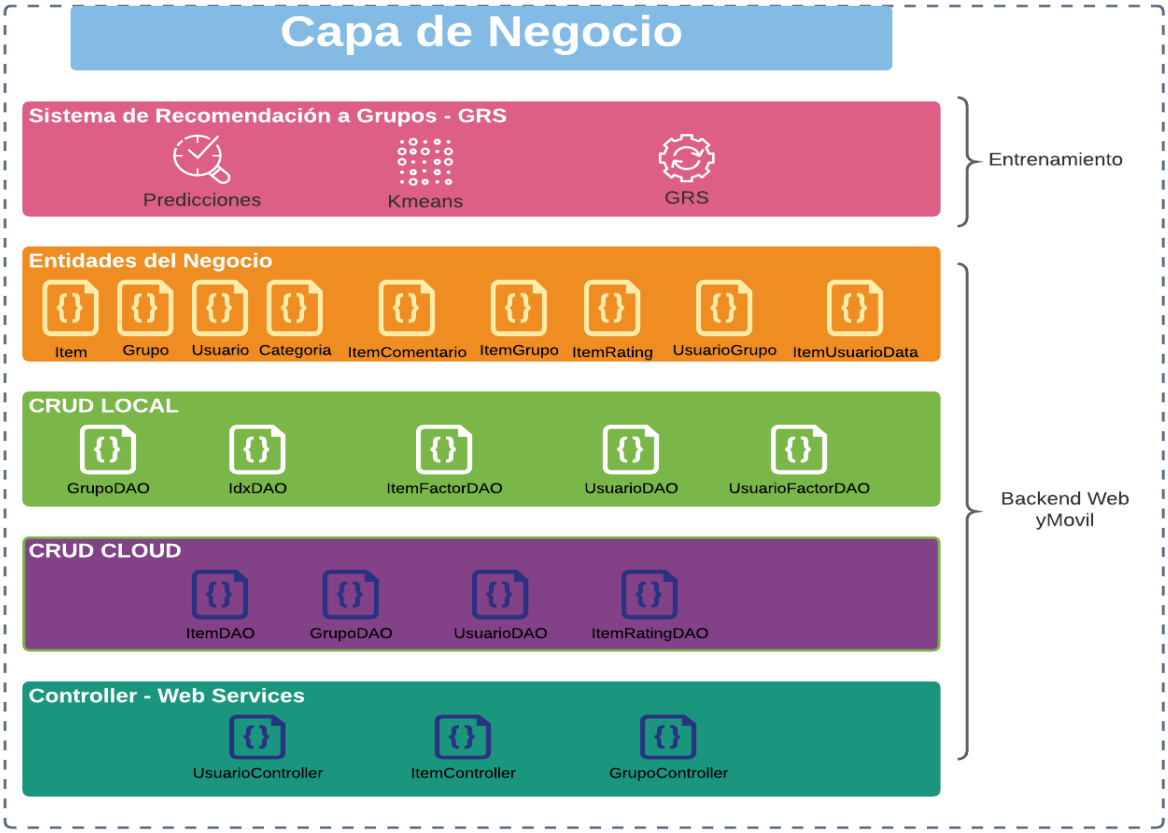


Figura 3 Capa de Negocio - GRS

2.1.2.1 Módulo GRS

En términos generales, este módulo se encarga de reducir la dimensionalidad de los datos, realizar un clustering de usuarios y realizar las predicciones y recomendaciones a grupos de usuarios detectados mediante del clustering, y también permite recomendar a grupos de usuarios que han sido conformados manualmente a través de la aplicación. Más adelante, en la subsección del diseño del sistema de recomendación se explica a detalle el modelo de aprendizaje.

2.1.2.2 Módulo de entorno Web y Móvil – Backend

En el módulo de entorno web y móvil se específica tres secciones importantes que permiten el procesamiento lógico para la comunicación con la capa de presentación.

a) Entidades de negocio, representa los objetos ítem, Usuario, Grupos entre otros objetos secundarios.
b) Métodos CRUD Local y Cloud para ítem, Usuarios, Grupos y otros documentos permiten la comunicación hacia la base de datos no relacional.
c) Los Controladores contiene la lógica necesaria para dar respuesta a las peticiones que se realicen en la vista "Aplicación Móvil y Aplicación Web, por ejemplo, la creación de un nuevo grupo o la edición del perfil del usuario en la vista móvil y la creación de un nuevo ítem o la edición del mismo en la vista web.

2.1.3 Capa de Datos

2.1.3.1 Gestión de la base de datos

Para que el Sistema de Recomendación a Grupos – GRS sea capaz de realizar las recomendaciones de forma correcta necesita un entrenamiento con un conjunto de datos, para el entrenamiento se utilizara MovieLens 1M. Estos archivos contienen 1000209 clasificaciones anónimas de aproximadamente 3900 películas creado por 6040 usuarios de MovieLens en el año 2000 (GroupLens, 2000).

La información se reutilizará en futuras recomendaciones, por lo que es necesario almacenar el conjunto de datos en la base de datos no relacional MongoDB (migración de datos) en la que están implementadas los distintos documentos de almacenamiento de datos. Véase el diseño de la base de datos no relacional ***Figura 5***.

Para el diseño del núcleo se ha diseñado una clase por cada documento de la base de datos en formato JSON. Para los métodos inserción, modificación o recuperación de datos, se utilizará una herencia de la clase MongoRepository.java librería spring-data-mongodb-2.2.0.RELEASE.jar. Los métodos son implementados en WS publicados en el núcleo, para que la aplicación móvil y la aplicación WEB las invoquen según las funcionalidades que se necesite implementar.

Dentro del núcleo se maneja dos implementaciones: una base de datos local para el entrenamiento y la base de datos Cloud para la aplicación móvil y aplicación web. La clase para la configuración con

la base de datos local se llama PrimaryMongoConnection.java que contiene los datos de conexión a la base de datos (usuario, servidor y puerto). La clase para la configuración con la base de datos Cloud se llama SecondaryMongoConnection.java que contiene los datos de conexión (usuario, servidor y puerto).

Las clases de configuración proporcionan la conexión entre cada documento y su entrada en la base de datos. De esta forma, cada clase tienen como atributos los elementos de los documentos de la base de datos para realizar las respectivas consultas.

2.1.3.1.1 Migración

El entrenamiento del GRS se centra en la información del archivo MovieLens “rating.dat”, con el siguiente formato IDUser;IDItem;Rating. El entrenamiento del modelo se realza mediante una validación cruzada, para lo cual se ha dividido el archivo en dos archivos train.dat (80% de la información para el entrenamiento) y test.dat (20% de la información para el testing del GRS).

Para almacenar la información en la base de datos local se utiliza el siguiente WS:

- **“.../usuario/migrarlocal?nombredataset=ML1M&separator=;”**

Después de la migración se almacenará los objetos usuario.java, item.java necesarios para poder realizar el entrenamiento del GRS.

Para almacenar la información en la base de datos Cloud se utiliza el siguiente WS:

- **“.../usuario/migrarcloud?nombredataset=ML1M&separator=;”**

Para almacenar la información de los usuarios y de los ítems, se ejecutará los siguientes WS:

- **“.../usuarioData/migrarUsuarioData?nombredataset=ML1M&separator=,”**
- **“.../itemData/migrarItemData?nombredataset=ML1M&separator=$”**

Para completar la información de los usuarios e ítems se ejecutarán los siguientes WS:

- **“.../usuario/completarmigracion/”**
- **.../item/completarmigracion/”**

- **“.../item/migrarCategoria?nombredataset=ML1M&separator=;**

2.1.3.2 Diagrama de base de datos no relacional GRS

A continuación, se especifica el diagrama de base de datos Cloud no relacional del Sistema Recomendador a Grupos de Usuarios.

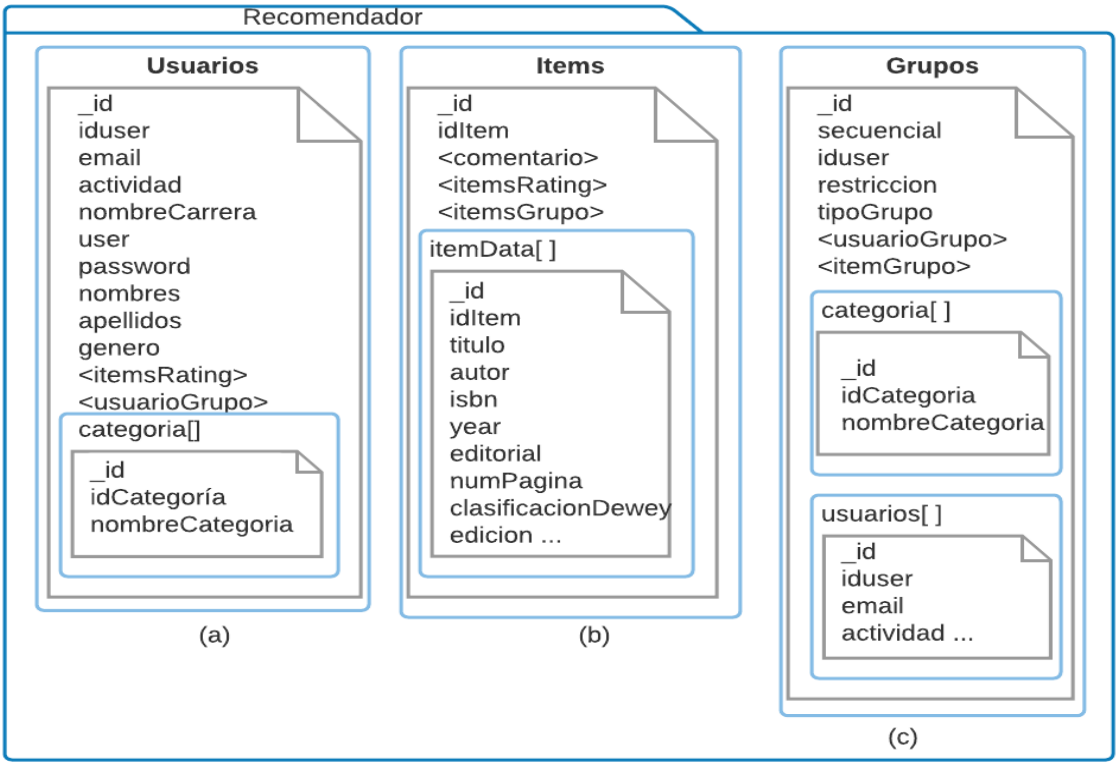


Figura 4 Diseño Base de Datos GRS Cloud – Documentos

En la ***Figura 4*** se muestra la primera sección del esquema de base de datos no relacional del Sistema de Recomendación a Grupos, a continuación, se explica la funcionalidad de los tres documentos principales del sistema:

a) Documento ***Usuarios***, almacena la información de registro del estudiante o docente que se digita en la aplicación móvil como idUser, email, actividad, nombreCarrera, user, password, nombres, apellidos, genero, <itemRating>, <usuarioGrupo>, de igual manera contiene un sub documento de categorías[ ] que almacena el idCategoria y nombre de la categoría seleccionadas al realizar el registro de un nuevo usuario.

b) Documento ***ítems***, almacena toda la información del ítem que es ingresada por el administrador y que posteriormente será recomendado aún usuario o grupos de usuarios.

c) Documento **Grupos**, almacena la información del grupo como: idUser, restricción, tipoGrupo, <usuarioGrupo>, <itemGrupo>, del usuario que ha creado un nuevo grupo o el grupo recomendado al usuario, contiene dos sub documentos categoria[ ] que permite

almacenar la categoría seleccionada al crear un nuevo grupo y usuario[ ] almacena los usuarios que pertenecen al nuevo grupo creado o grupo recomendado.

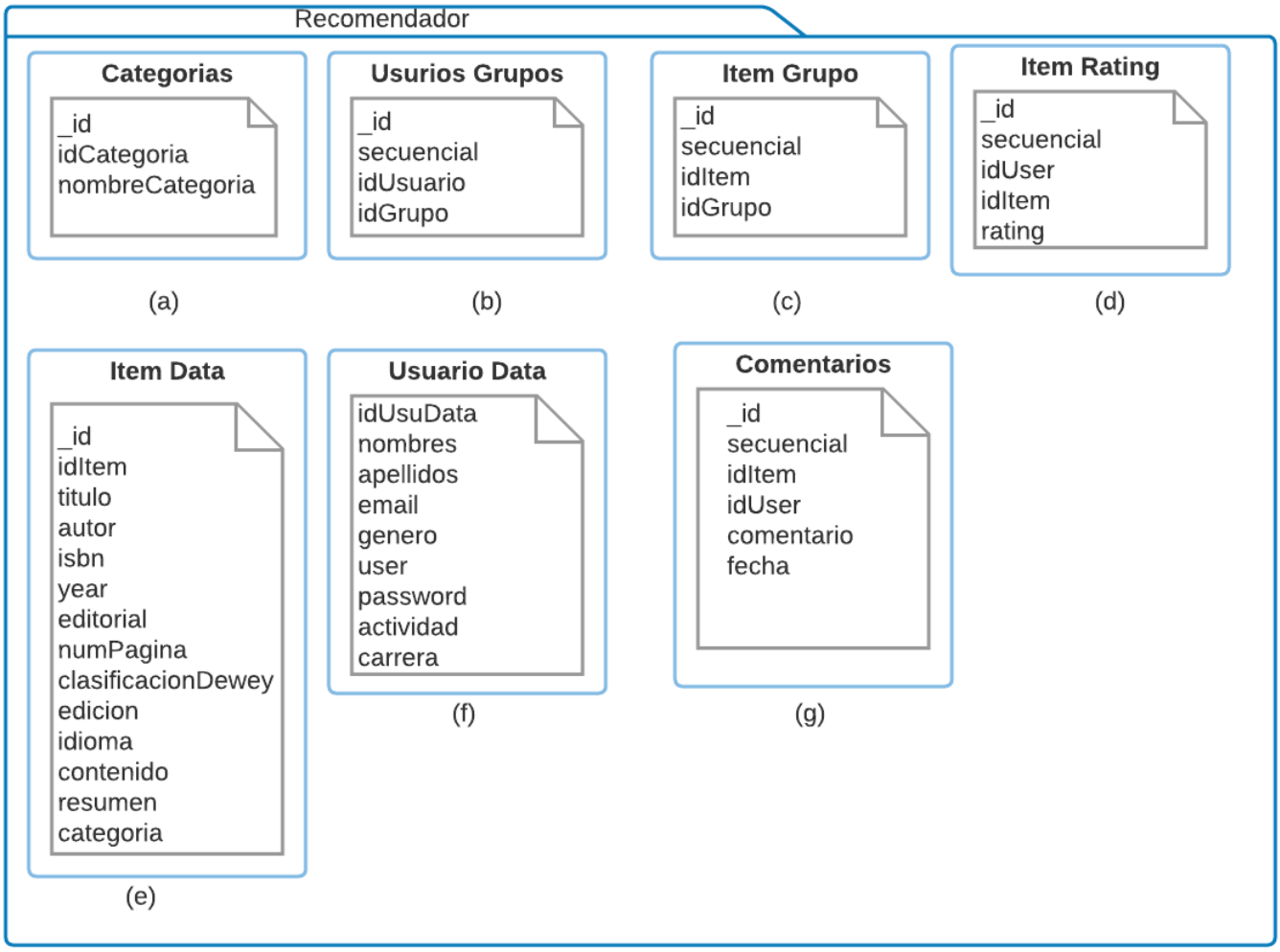


Figura 5 Diseño Base de Datos GRS Cloud - Documentos Simples

La ***Figura 5*** representa los documentos simples es decir no contiene subdocumentos.

a) Documento ***Categorías***, almacena el idCategoría y nombre de Categorías que se migra desde un archivo plato txt.
b) Documento ***Usuarios Grupos***, almacena el idUsuario y IdGrupo.
c) Documento ***Item Grupo***, almacena el idItem y idGrupo para identificar que items que se han recomendado al grupo.
d) Documento ***Item Rating***, almacena la calificación que un usuario da al ítem.
e) Documento ***Item Data***, almacena toda la información del ítem, desde el título, autor, ISBN, año, editorial, número de página, clasificación Dewey, edición, contenido, resumen, idioma e índice.
f) Documento ***Usuario Data***, almacena la información del usuario que son migrados desde un archivo plato txt.

g) Documento ***Comentarios***, almacena el idItem, idUsuario, comentario y fecha para identificar que usuario agregado un nuevo comentario a un ítem específico.

2.1.4 Servidores

Los servidores adquiridos están bajo una distribución CentOS y cuentan con la siguiente estructura.

- **Servidor de Backend y Frontend**

  Backend este servidor contiene toda la lógica de negocio, es decir está el algoritmo del Sistema de Recomendación a Grupos de Usuarios, los métodos CRUD que por medio de web services permite realizar todas las funcionalidades. **Frontend** es toda la implementación móvil y web que permite al usuario generar información que será utilizada para el entrenamiento.

- **Servidor de Base de datos Local**

  La Base de Datos Local se implementa un proceso manual para procesar la información ingresada y realizar el entrenamiento del sistema de recomendación. El proceso de entrenamiento se ejecuta cada hora para alimentar el sistema de recomendación. El sistema se provee con la retroalimentación de los votos de los usuarios que se almacenan en el servidor que contiene la base de datos Cloud.

- **Servidor de Base de Datos Cloud**

  La Base de Datos Cloud permite almacenar toda la información que se genera con el uso de la aplicación móvil y web, es decir nuevos grupos de usuarios, calificaciones, comentarios, nuevos registros de usuarios e ítems.

2.2 Diseño del Sistema de Recomendación a Grupos

En la ***figura 6*** se presenta el modelo del sistema de recomendación. En esta subsección se explica los pasos del proceso y las técnicas aplicadas.

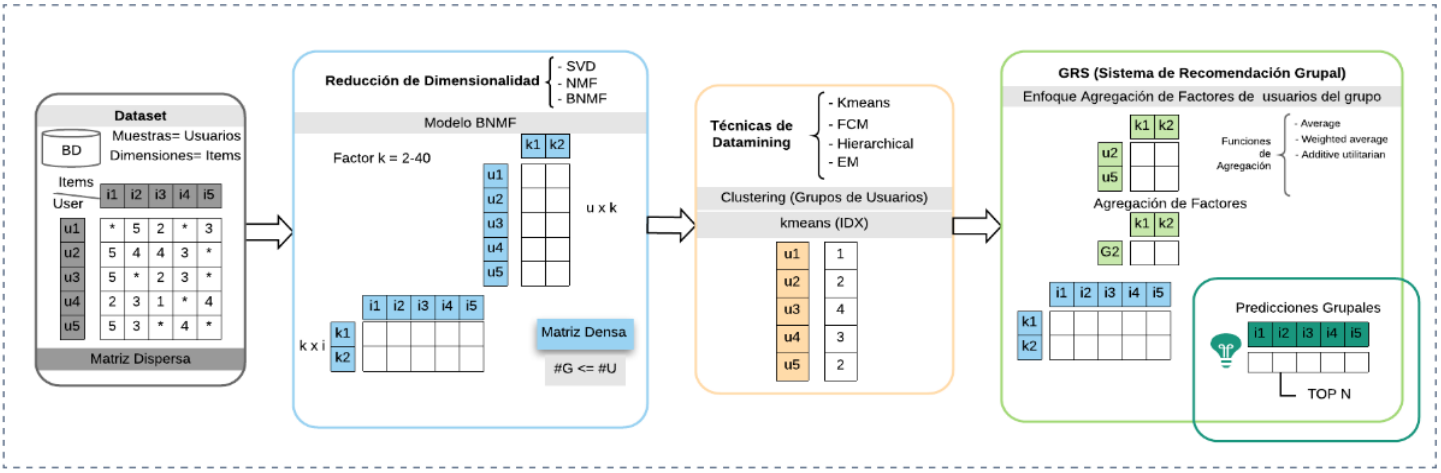


Figura 6 Diseño del Sistema de Recomendación a Grupos

El sistema sigue dos fases: 1. Entrenamiento, y 2. Recomendación. A continuación, es detalla el proceso de cada fase.

Entrenamiento

Paso 1: Con el conjunto de entrenamiento (80%) iniciamos la reducción de dimensionalidad de una matriz NxN con el modelo BNMF (Hernando, Bobadilla, & Ortega, 2016) esto permitirá la reducción de dimensionalidad de una matriz dispersa a una matriz densa de menor dimensionalidad de Usuarios e ítems.

Paso 2: A partir del modelo reducido y por medio de Clustering "Kmeans" agrupamos a los usuarios por el número de grupos que deseamos, de esta manera tendremos similitudes dentro del mismo grupo, y diferencias a otros grupos; posteriormente este proceso permitirá recomendar ítems a grupos de usuarios.

Paso 3: Obtenemos los grupos de usuarios en un vector de índices que denominamos ***IDX***, donde el valor para cada usuario es el grupo al que pertenece el usuario

Paso 4: Realizamos la distribución de usuarios en los grupos, esto nos permitirá saber cuántos usuarios tiene el grupo.

Paso 5: A partir de las pertenencias por grupo se podrá obtener los datos de cada usuario y su grado de pertenencia a cada grupo.

Paso 6: Por último, a partir de las pertenencias y de los factores del modelo reducido se podrá obtener las predicciones. Más adelante se explica el modelo de predicción.

Paso 7: Con los votos reales del conjunto de testing (20%) se mide la calidad del sistema de recomendación**.**

Recomendación

Paso 1: Ordenamos las predicciones en orden descendente.

Paso 2: Presentamos las n predicciones más altas que serán las n recomendaciones de los ítems a presentar a cada grupo y a cada usuario.

Modelo de predicción

La explicación a detalle del modelo de aprendizaje y predicción que se ha incorporado se presenta en el trabajo de (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020)Este modelo se ha incorporado en este trabajo de titulación y se ha desarrollado bajo frameworks de desarrollo empresarial para llevar a fase de producción.

A continuación, se explica brevemente el modelo:

Inicialmente tenemos un matriz cuyas muestras son los usuarios representados por $\boldsymbol{u}$, y las dimensiones son los ítems representados por $\boldsymbol{i}$. El filtrado colaborativo presenta el inconveniente de la escasez de votos por parte de los usuarios. Esto da lugar a una matriz demasiado grande con muchos vacíos de información, esta matriz también se conoce como una matriz dispersa. Trabajar con una matriz dispersa con gran dimensionalidad requiere un alto consumo de recursos computacionales; por ejemplo, el dataset público de películas ML1M da una matriz de 6040x3768 con la cual podríamos tener 22758720 votos, sin embargo, ML1M presenta aproximadamente sólo 1000000 votos, es decir, existe una escasez del 95.6%. Por lo tanto, surge la necesidad de realizar una transformación de los datos mediante modelos o técnicas de reducción de dimensionalidad.

En investigaciones recientes se ha utilizado SVD y NMF como técnicas de reducción de dimensionalidad (Bobadilla, Bojorque, Hernando, & Hurtado, 2017), en este trabajo se utiliza BNMF (Bobadilla, Bojorque, Hernando, & Hurtado, 2017) para convertir una matriz dispersa en una matriz densa de menor dimensionalidad. Este modelo brindará una característica en especial que se denominará factores ocultos, es decir, al tener una matriz $\boldsymbol{uxi}$ ahora tendremos una matriz de factores ocultos para usuarios $\boldsymbol{uxk}$ y una matriz de factores ocultos para ítems $\boldsymbol{kxi}$, donde $\boldsymbol{k}$ es el factor oculto. Según (Bobadilla, Bojorque, Hernando, & Hurtado, 2017), el factor $\boldsymbol{k}$ puede ser entre 2 y 40, es decir ahora trabajaremos en una dimensión $\boldsymbol{k}$ que es mucho más pequeña a la dimensión original $\boldsymbol{i}$. Al trabajar con la matriz densa, tendremos un mejor rendimiento en tiempo y costo computacional. Los factores ocultos calculados permitirán dos funciones:

a. Realizar un clustering con menos uso de recursos.

b. Realizar las predicciones a grupos detectados mediante clustering y a grupos conformados por los mismos usuarios.

Para detectar grupos de usuarios necesitamos agrupar usuarios, para ello se puede usar varias técnicas de clustering de data mining como: K Means, Fuzzy C means, clustering jerárquico, Expectation Maximization, entre otros. Según (Bobadilla, Bojorque, Hernando, & Hurtado, 2017), la técnica de clustering con la cual se ha logrado mejores resultados en los sistemas de recomendación es Kmeans. Con la técnica de agrupación, obtendremos los grupos de usuarios en un vector de índices que denominamos ***IDX***, donde el valor para cada usuario es el grupo al que pertenece el usuario.

El **descubrimiento de grupos de usuarios** permitirá predecir y recomendar ítems a grupos de usuarios detectados automáticamente, este escenario podrá servir por ejemplo para segmentación y recomendación a mercados de usuarios. El proceso de predicción en este escenario de grupos descubiertos se realiza de la siguiente manera:

- Se toman los factores ocultos (en términos de probabilidades) de todos los usuarios del grupo detectado automáticamente mediante Kmeans.
- Se realiza una agregación de los factores del grupo mediante una función promedio, de tal manera que se obtiene los factores de probabilidad que representan al grupo.
- Se realiza el producto punto entre los factores del grupo y los factores de ítems calculados previamente a través de BNMF. Esto genera las predicciones de un grupo para cada ítem.

Para la predicción a **grupos de usuario conformados** manualmente por un usuario a través de la aplicación se realiza el siguiente proceso:

- Se toman los factores ocultos (en términos de probabilidades) de todos los usuarios del grupo conformado manualmente.
- Se realiza una agregación de los factores del grupo mediante una función promedio, de tal manera que se obtiene los factores de probabilidad que representan al grupo.
- Se realiza el producto punto entre los factores del grupo y los factores de ítems calculados previamente a través de BNMF. Esto genera las predicciones de un grupo para cada ítem.

En el capítulo 3, se presenta los resultados de calidad del modelo del sistema de recomendación.

2.3 Diseño del sistema Móvil y Web

El diseño del sistema móvil y web se basan en los requerimientos para un sistema recomendador a grupos de usuarios de una biblioteca. Para ello se realizan los mockups de la aplicación móvil y aplicación web utilizando la

herramienta Balsamiq que nos permite tener una estructura base y proceder con el desarrollo.

A continuación, se presentan las ilustraciones base que se utilizó como guía para el desarrollo de estas dos aplicaciones.

2.3.1 Mockup Móvil

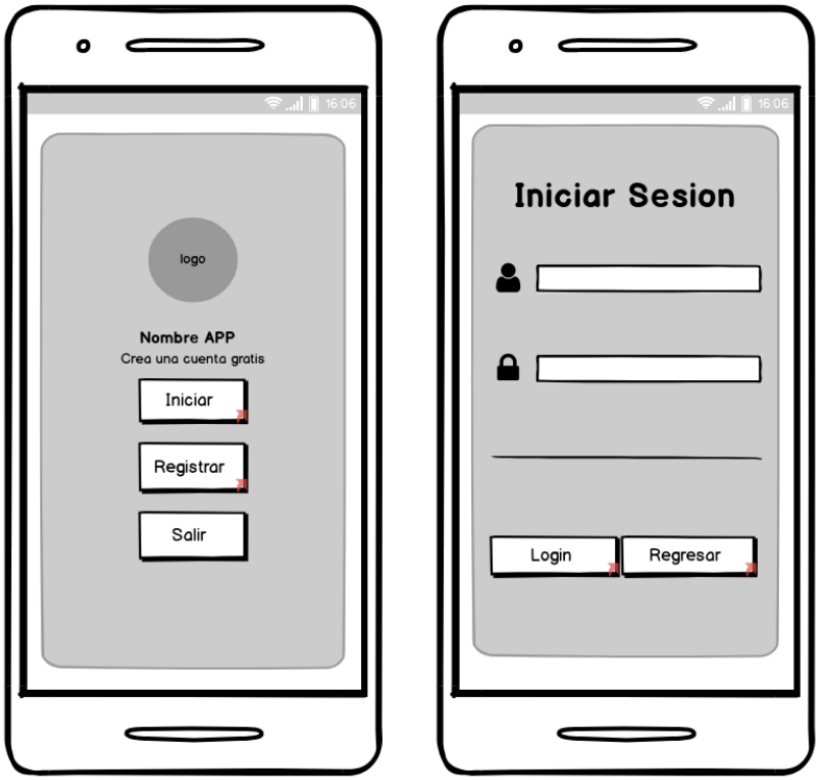


Figura 7 Mockup Móvil Login

Figura 8 Mockup Móvil Registro

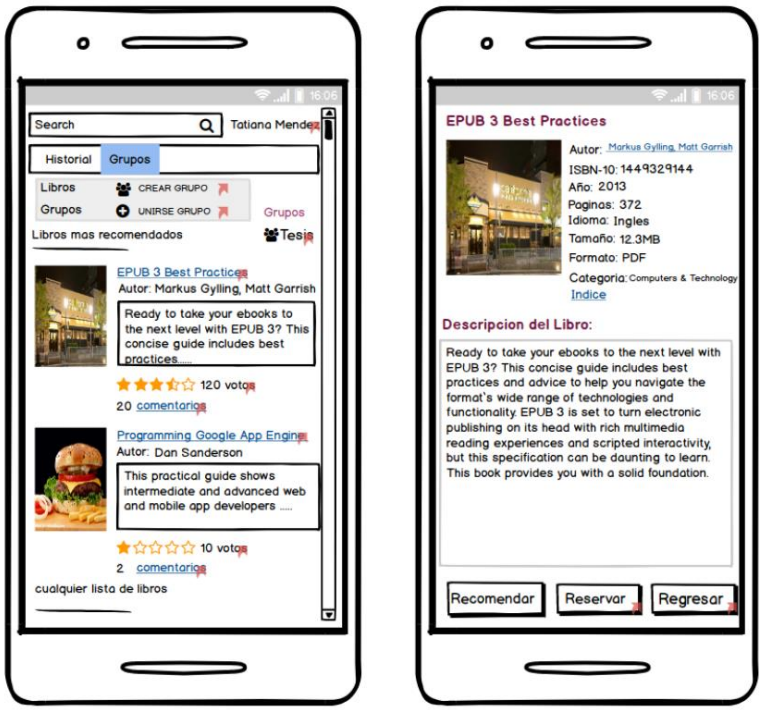


Figura 9 Mockup Móvil Home - Información del Item

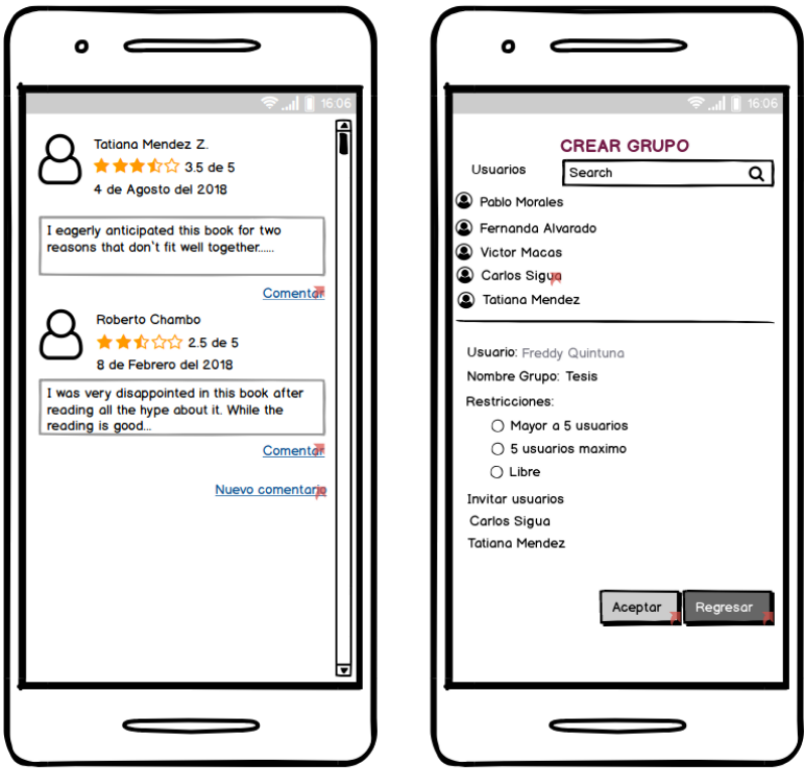


Figura 10 Mockup Móvil Comentarios - Crear nuevo Grupo

2.3.2 Mockup Web

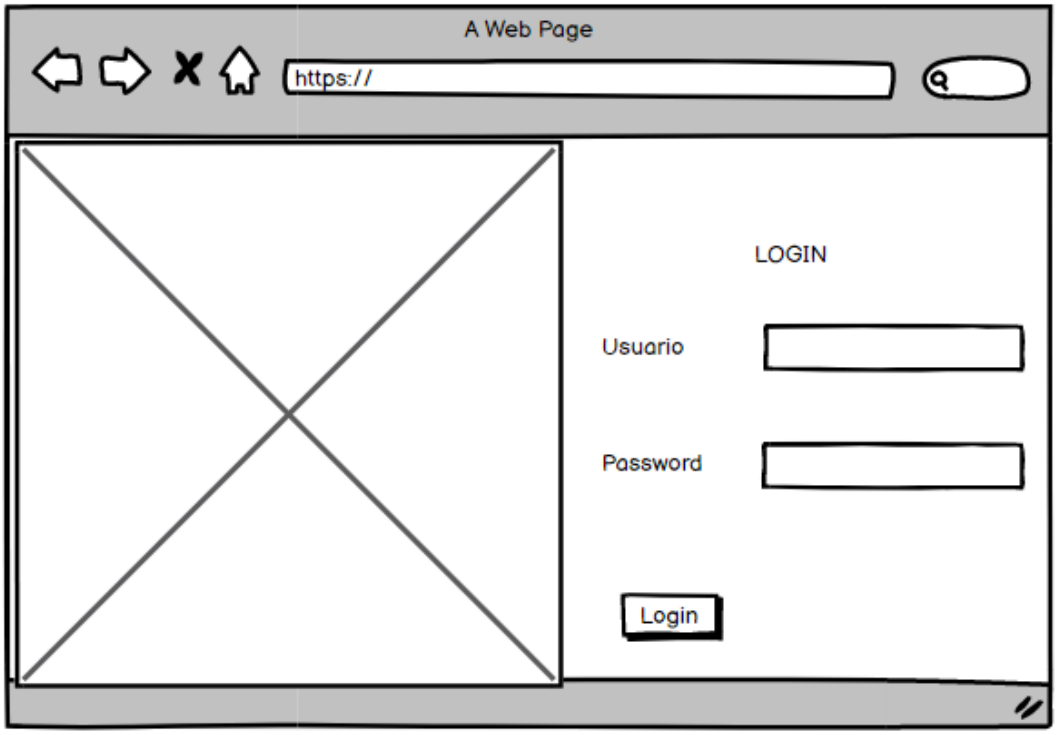


Figura 11 Mockup Web – Login

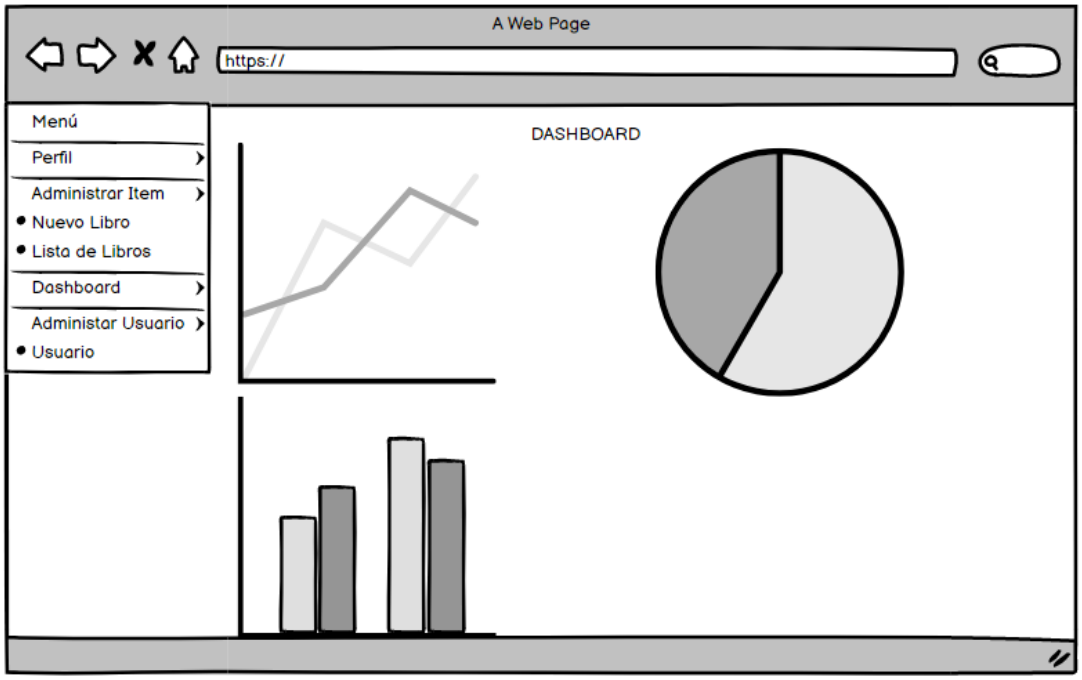


Figura 12 Mockup Web - DASHBOARD

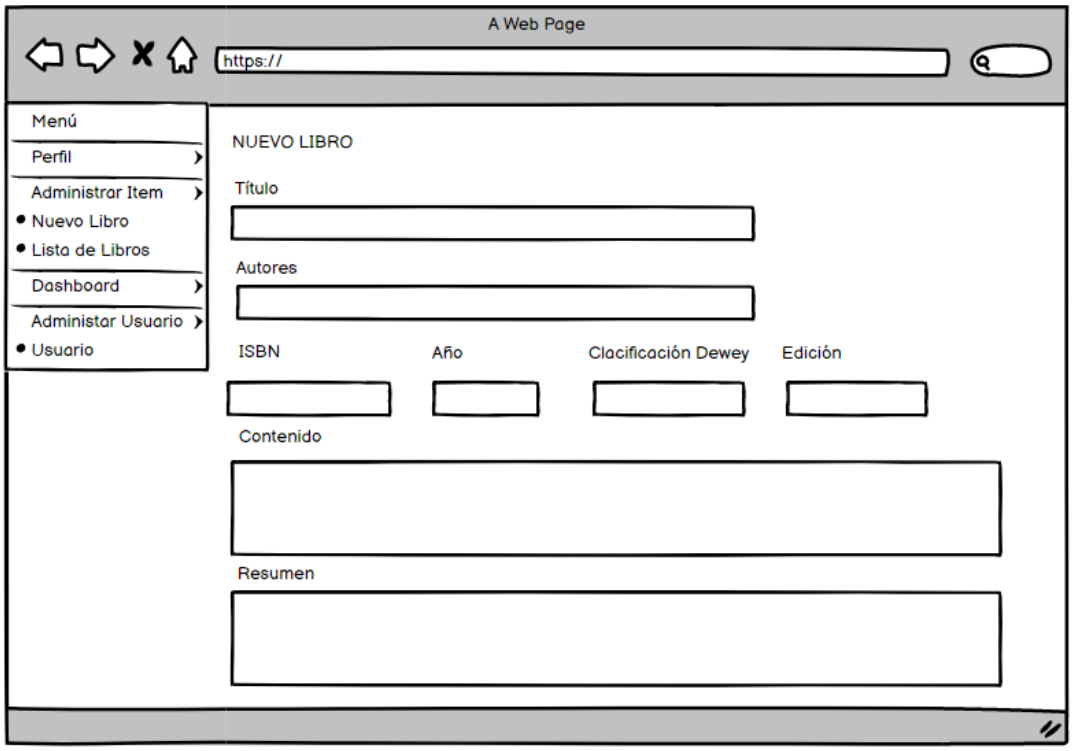


Figura 13 Mockup Web - Nuevo Libro

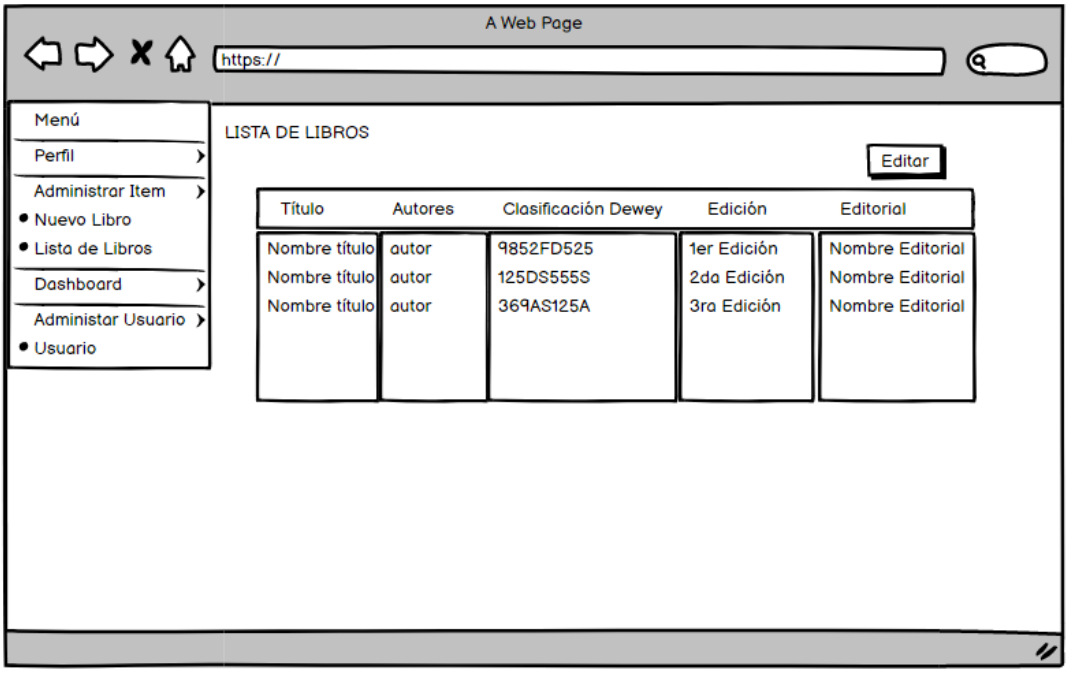


Figura 14 Mockup Web - Lista de Libros

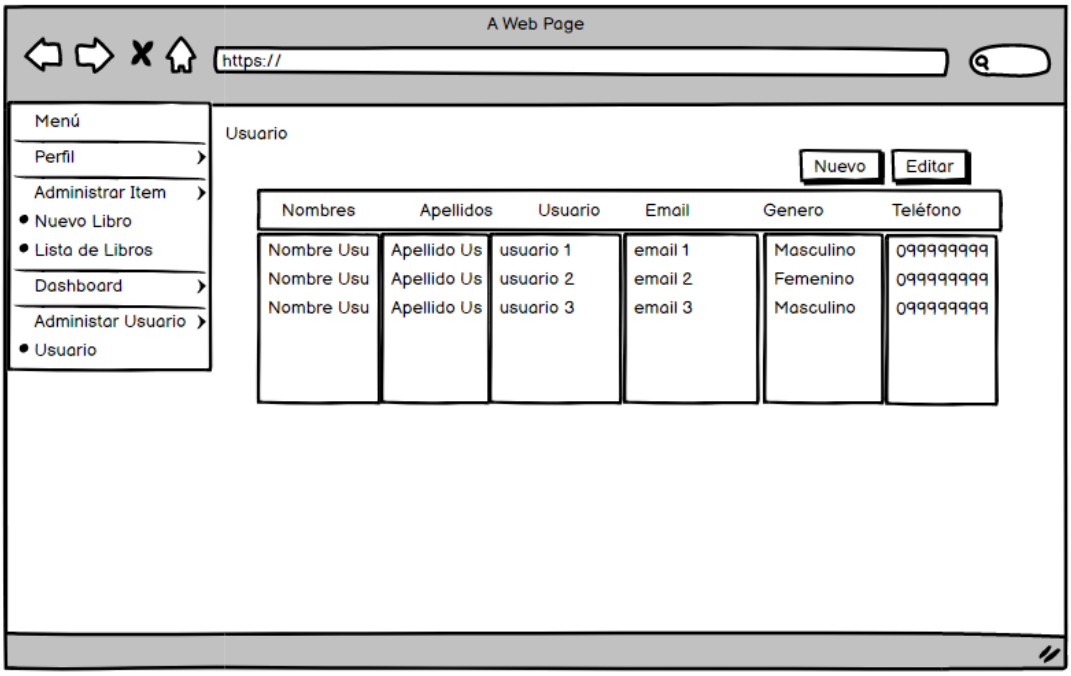


Figura 15 Mockup Web - Usuario

2.3.3 Diagramas Casos de Usos

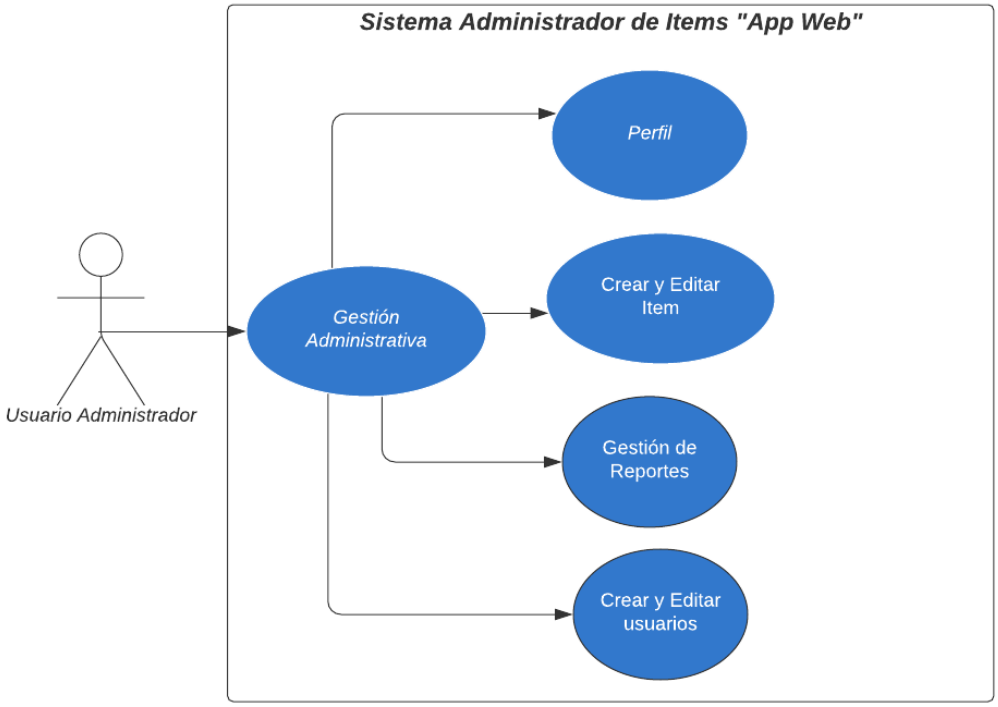


Figura 16 Diagrama Caso de Uso: Administrador de Items

Tabla 1 Especificaciones caso de uso Administrador Web

Caso de Uso: Administrador de ítems	
Actor	Usuario Administrador
Descripción	El administrador iniciara sesión con el usuario y password para acceder al back office web.
Actividades	• Ver Perfil de Usuario • Crear un nuevo Usuario • Editar un usuario existente • Crear un nuevo ítem (Libro) • Editar un ítem existente (Libro) • Gestionar Reportes

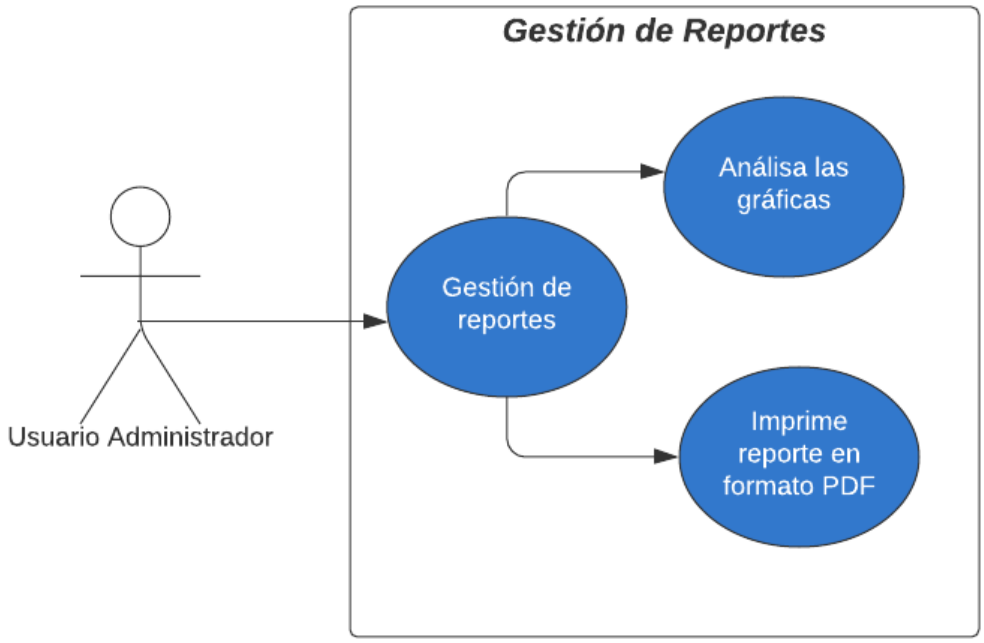


Figura 17 Diagrama de caso de uso: Gestión de Reportes

Tabla 2 Especificaciones caso de uso Gestión de Reportes

Caso de Uso: Gestión de Reportes	
Actor	Usuario Administrador
Descripción	El administrador analizará las gráficas presentadas en la ventana Dashboard.
Actividades	• Analizar gráficas • Imprimir reportes

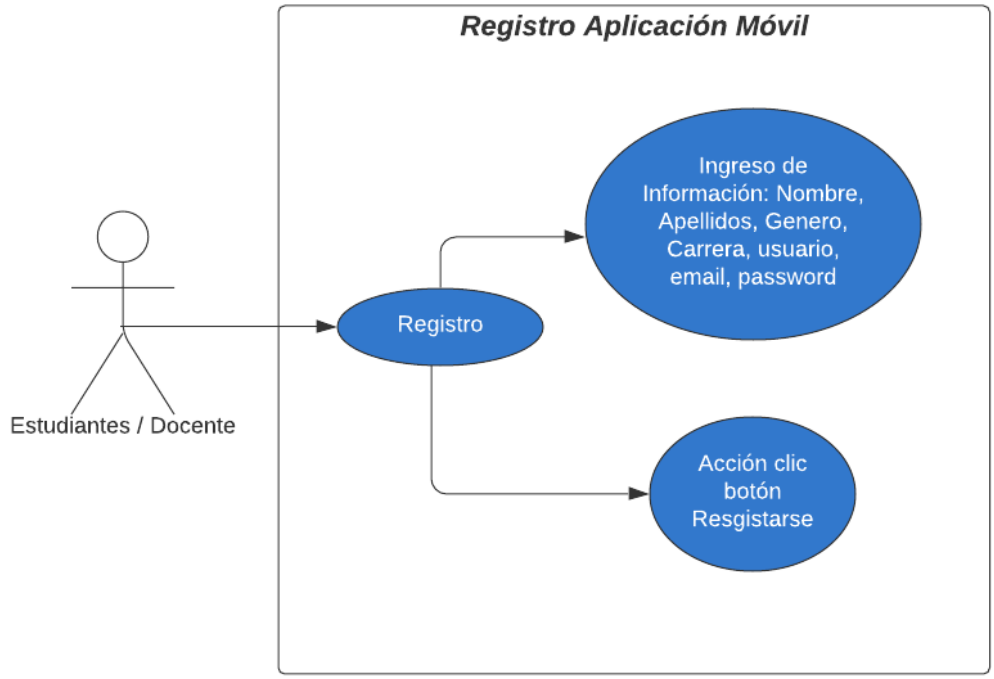


Figura 18 Diagrama Caso de Uso – Registro

Tabla 3 Especificaciones caso de uso: Registro

Caso de Uso: Registro	
Actor	Estudiante / Docente
Descripción	El nuevo usuario creará una cuenta para acceder a la aplicación móvil.
Actividades	• El usuario ingresa sus Nombre, Apellidos, Genero, Carrera, Nombre de Usuario, email y Password. • Hace clic en el botón registrar para culminar el proceso.

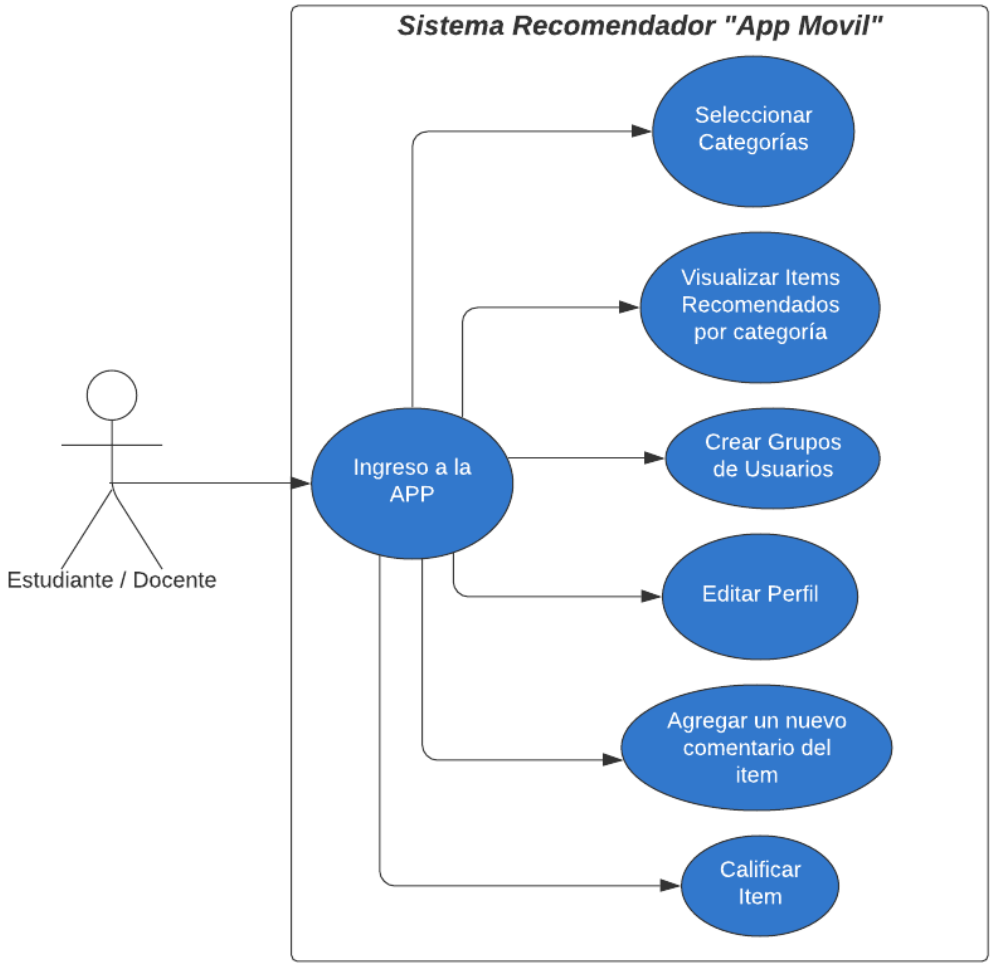


Figura 19 Diagrama Caso de Uso - Aplicación Móvil

Tabla 4 Especificaciones caso de uso: Aplicación Móvil

Caso de Uso: Aplicación Móvil	
Actor	Estudiante / Docente
Descripción	El usuario crea grupos, agrega comentarios, califica ítems
Actividades	• El usuario ingresa con el nombre de usuario y password registrados.

	• El usuario nuevo seleccionará una lista de categorías de interés.
	• Visualizará los ítem recomendados por las categorías seleccionadas.
	• El usuario creará nuevos grupos
	  ○ Asignará usuarios
	  ○ Seleccionará categoría de interés
	  ○ Ingresará nombre del nuevo grupo
	• El usuario podrá editar el perfil.
	• Crear un nuevo comentario al ítem
	• Calificar al ítem del interés

2.4 Desarrollo del sistema web

Para el desarrollo de la aplicación web se trabajó con el framework Angular que permite el desarrollo de aplicaciones web. El código del sistema web se encuentra en el repositorio: https://github.com/fquintuna/grsfrontend.git

Se presenta la siguiente estructura de componentes:

- Login
  - Ws usuario/loginAdmin
- Dashboard
  - Implementa las gráficas con la librería Chartlist
- New-book
  - WS: item/nuevoItem
  - WS: item/listaCategoria
- List-book
  - WS: item/listaCategoria
  - WS: item/listItemNombreCategoria
  - WS: item/acualizarItem
- New-user
  - WS: usuario/registroAdmin
- Perfil
  - WS: usuario/perfilAdmin
- Servicios

2.5 Desarrollo del sistema móvil

Las tecnologías utilizadas para el desarrollo de la aplicación móvil es el Framework angular para implementar la lógica del Frontend y el framework ionic para el despliegue de la interfaz. El código del sistema web se encuentra en el repositorio: https://github.com/fquintuna/grsmovil.git

Se presenta la siguiente estructura de componentes:

- Login
  - WS: usuario/login
  - WS: usuario/listausuario
- Registro
  - WS: usuario/registro
- Home
- Menu
- Page-principal
  - WS: usuario/obtenerTopCategoriaItemUser
- Crear-grupo
  - WS: grupo/grupo
  - WS: usuario/getUsersNombres
  - WS: usuario/getAllUsersRandom
  - WS: grupo/listaCategoria
- Informacion-libro
  - WS: item/itemsIdItem
- Mantenimiento-grupo
  - WS: grupo/informacionGrupo
  - WS: grupo/informacionGrupoItems
- Informacion-comentarios
  - WS: item/itemsIdItemComentario
  - WS: item/guardarItemComentario
- Modal-categorias
  - WS: usuario/listaCategoria
  - WS: usuario/actualizarCategoria
- Popover-grupos
  - WS: grupo/listagrupousr
- Servicios

2.6 Integración y despliegue en la nube

El GRS está conformado por 3 componentes GRSBACKEND, GRSFRONTEND, GRSMOVIL.

Componente GRSBACKEND: manejo de la lógica de negocio, donde se realiza el procesamiento de información. Desarrollado en lenguaje JAVA, para la gestión de librerías se maneja el software MAVEN. El desarrollo se lo realizo IDE Springboot 4.

Componente GRSMOVIL: aplicación móvil maneja la interfaz para el usuario, se presenta los ítems y grupos recomendados para el usuario. La aplicación fue desarrollada en framework IONIC. El desarrollo se lo realizo en IDE Atom.

Componente GRSFRONTEND: aplicación web maneja la interfaz para los usuarios administradores, se realiza mantenimiento de usuarios

administradores, se muestra graficas del sistema de recomendación para la toma de decisiones.

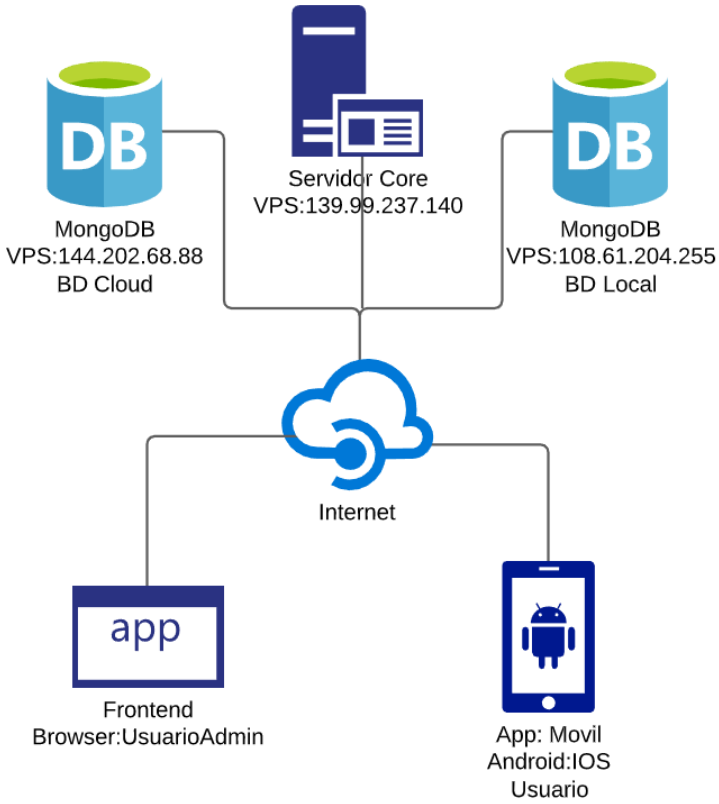


Figura 20 Distribución de componentes despliegue en la nube

La descripción de la integración se realizará en el siguiente orden:

1. Servidor DB Mongo.
2. Servidor Core (Backend), Servidor Apache (Frontend)
3. Frontend (Angular, Bootstrap)

Servidor Mongo DB, DB Cloud

Se contrata VPS (Servidor Virtual Privado) en Vultr; "Vultr, fundada en 2014, tiene la misión de empoderar a los desarrolladores y empresas simplificando la implementación de la infraestructura a través de su avanzada plataforma en la nube."

Características:

- Sistema Operativo: Linux Centos 8.2.
- IP: 144.202.68.88.
- Hostname: systemavatar.
- Memoria disco: 20 GB.
- Memoria RAM: 1024 MB.

Configuración base de datos Cloud:

1. Ingresar BD Mongo
   - # mongo

2. Crear base de datos
    - use databasecloud
3. Crear usuario con password
    - db.createUser({user:'usercloud',pwd:'warcrafthorda',roles:[{role:'db Owner', db:'databasecloud'}]})
4. Para habilitar autentificación, puerto de escucha de la base de datos
    - Configurar archivo mongodb.conf
    - Habilitamos parámetro
        - security:
            - authorization: “enabled”
    - El puerto por defecto es 27017

Servidor Mongo DB, DB Local

Se contrata VPS (Servidor Virtual Privado) en Vultr.
Características:

- Sistema Operativo: Linux Centos 8.2.
- IP: 108.61.204.255.
- Hostname: systemlocal.
- Memoria disco: 20 GB.
- Memoria RAM: 1024 MB.

La instalación de realiza según manual Configuración Instalación MongoDB.
Configuración base de datos local:

1. Ingresar BD Mongo
    - # mongo
2. Crear base de datos
    - use databaselocal
3. Crear usuario con password
    - db.createUser({user:'userlocal',pwd:'warcrafthorda',roles:[{role:'db Owner', db:'databaselocal'}]})
4. Para habilitar autentificación, puerto de escucha de la base de datos
    - Configurar archivo mongodb.conf
    - Habilitamos parámetro
        - security:
            - authorization: “enabled”
    - El puerto por defecto es 27017

Servidor Core Backend, Servidor Apache

Se contrata VPS (Servidor Virtual Privado) en OVH; “Estamos en el corazón de la revolución de los datos que está abriendo las puertas a nuevas y amplias posibilidades de aplicaciones y tecnología.”
Características:

- Sistema Operativo: Linux Centos 8.2.
- IP: 139.99.237.140.
- Hostname: systemcore.

- Memoria disco: 100 GB.
- Memoria RAM: 8192 MB.

Se configura Java Virtual Machine, para ejecutar la aplicación Backend, con los siguientes comandos:

- #curl -O https://download.java.net/java/GA/jdk13.0.1/cec27d702aa74d5a8630c65ae61e4305/9/GPL/openjdk-13.0.1_linux-x64_bin.tar.gz
- #dnf install tar –y
- #tar xvf openjdk-13.0.1_linux-x64_bin.tar.gz
- #alternatives --install /usr/bin/java java /opt/jdk-13.0.1/bin/java 3
- #alternatives --config java
- #nano /etc/profile
    - export JAVA_HOME=/opt/jdk-13.0.1
    - export PATH=$PATH:/opt/jdk-13.0.1/bin

Revisión de conectividad para acceso a base de datos:

- telnet 108.61.204.255 27017
    - conexión satisfactoria
- telnet 144.202.68.88 27017
    - conexión satisfactoria

Proceso para levantar Backend:

- Crear carpeta donde se alojará jar con la lógica entrenamiento
    - mkdir /home/core/grs/
- Desde la máquina de desarrollo, copiamos la app dentro de la dirección del servidor
    - scp –r grscore.jar root@139.99.237.140:/home/core/grs/
- Para tener funcional el Backend, creamos un ejecutable .sh
    - nano /home/core/grs/ejecutable/rungrs.sh
    - #bin bash
    - Java –jar grscore.jar
- Permisos de ejecución
    - chmod 777 rungrs.sh
- Creamos crontab para ejecutar archivo de ejecución. Cada vez q se inicie el servidor el proceso Backend estará ejecutándose.
    - crontab –e
        - @reboot /home/core/ejecutable/rungrs.sh

Proceso para levantar Frontend: Servidor apache

- Instalamos servicio apache httpd.
- Luego de la instalación por defecto se crea las carpetas /var/www/html
- Para desplegar la app Frontend en el servidor
    - copiamos la carpeta generada en la máquina de desarrollo **dist/** dentro de nuestro servidor, dirección /var/www/html/
    - reiniciamos servicio httpd y realizamos la prueba de funcionamiento de Frontend.

Frontend (Angular)

Las versiones para el desarrollo y compilación son las siguientes:

Figura 21 Versión Angular

Comando para deployar aplicación Frontend, se creará la carpeta /dist/ para desplegar en el servicio apache.

- ng build –prod

CAPITULO 3: EXPERIMENTOS, VERIFICACIÓN DE REQUERIMIENTOS Y PRUEBAS FUNCIONALES

En este capítulo vamos a demostrar la calidad del sistema, primeramente, se medirá el modelo de aprendizaje para el sistema de recomendación, presentaremos los experimentos, mostraremos las métricas de calidad, en segundo lugar, mostraremos las pruebas funcionales del sistema web y las pruebas funcionales del sistema móvil.

3.1 Calidad del sistema de recomendación

El modelo implementado en el GRS ha sido tomado de (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020), como parte de los trabajos publicados dentro del grupo de Inteligencia Artificial y Tecnologías de Asistencia (GIIATA) del cual somos parte. En esta subsección se presenta los principales resultados del modelo que demuestran la alta calidad de las predicciones y recomendaciones del sistema de recomendación para grupos de usuarios descubiertos con clustering.

En primer lugar, en la Tabla 5 se detalla las características del dataset MovieLens utilizado en el proyecto.

Tabla 5 Características principales del Dataset usado en el proyecto

Dataset	Ratings	Items	Usuarios	Escala
MovieLens	1000209	3706	6040	Escala de 5 estrellas

En la tabla 6 (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020) se especifica los principales parámetros del diseño de experimentos, K indica la cantidad de grupos con los que se ha hecho los experimentos del paso de clustering, N es la cantidad de recomendaciones para medir la calidad de recomendación.

Tabla 6 Cross-validation values used in the experiments

Parámetros	Valores
Testing-Rating	20%
Training-Rating	80%
# clústers (k)	MovieLens:{20 to 200 step 5}
#recommendations (N)	15

Métricas de calidad

Se han utilizado las siguientes métricas de calidad, RMSE para medir la calidad de las predicciones y F1 para medir la calidad de las recomendaciones.

- **Root Mean Square Error (RMSE):** es una medida de la calidad de predicción.

$$RMSE = \frac{\Sigma_{i \epsilon I}(r_{u,i} - P_{G,i})^2}{n}$$

Donde ru,i es la calificación del usuario u para el elemento i. PG,i es la predicción para el elemento i para el grupo G en el que está el usuario u. I es el conjunto de todos los elementos y n es el número de clasificaciones disponibles en el equipo de prueba. Los valores bajos de RMSE son mejores, ya que significa que los errores de la predicción son menores (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020).

F1: para evaluar la calidad de las recomendaciones a los grupos de usuarios, definimos F1 como la media armónica que combina los valores de precision y recall. Definimos precision y recall para grupo G, las siguientes formulas fueron tomadas de (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020).

$$precision_G = \frac{\Sigma_{u \epsilon G} precision_u}{\#u}$$

$$recall_G = \frac{\Sigma_{u \epsilon G} recall_u}{\#u}$$

$$precision_u = \frac{\#TP}{\#(TP \cup FP)}$$

$$recall_u = \frac{\#TP}{\#T}$$

$$precision = \frac{\sum_G precision_G}{\#G}$$

$$recall = \frac{\sum_G recall_G}{\#G}$$

$$F1 = 2 * \frac{precision * recall}{precision + recall}$$

Donde G es el grupo en el que se encuentra el usuario u , $precision_u$ y $recall_u$ es la precisión y la recuperación para el usuario u (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020).

Resultados de reducción de dimensionalidad

En (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020) se demuestra que el modelo BNMF es mejor que los modelos PCA y SVD para el paso de reducción de dimensionalidad, por ello es el modelo de reducción implementado en el proyecto al obtener una mayor varianza utilizando el mismo número de dimensiones, en otras palabras, nos da mayor información del conjunto de datos originales. En la **figura 23** se aprecia que en BNMF con 15 factores/componentes se alcanza aproximadamente el 100% de la información (varianza equivalente a 1).

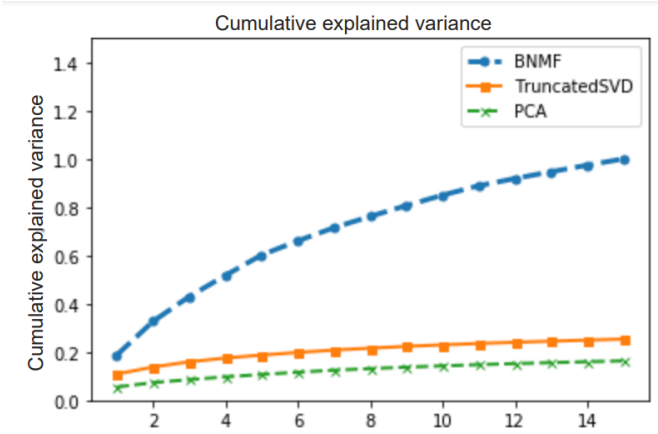


Figura 22 Comparación de técnicas de reducción de dimensionalidad. Figura tomada de (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020)

Resultados de clustering

En la figura 23 se demuestra que con BNMF se presenta distribución de usuarios más equitativa en comparación a PCA y SVD. Se puede apreciar que con PCA y SVD el grupo 1 supera los 1700 usuarios, lo que no es conveniente para nuestro escenario de segmentación de mercados. En general, en el contexto de CF podemos concluir que BNMF presenta mejores características en relación con los demás métodos y es capaz de comprimir de manera más adecuada la información original (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020)

.

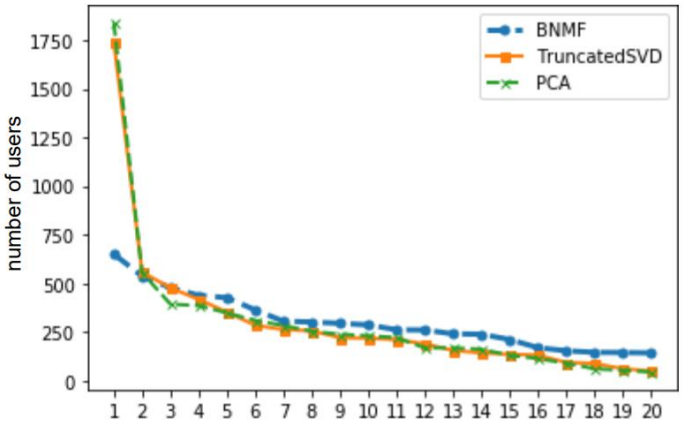


Figura 23 Distribución de usuarios en clusters. Figura tomada de (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020)

Resultados de calidad de predicción

La **figura 24** indica los resultados de calidad de predicción con el dataset MovieLens. La métrica RMSE obtenida cuando el método se ejecuta utilizando un número diverso de grupos (clusters) (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020). Se puede apreciar que el método RAF que es el método incorporado en este trabajo, presenta el error más bajo superando a PCA, RAP y VUR, que son otros métodos del estado del arte relacionado a recomendación a grupos de usuarios detectados con clustering. En (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020) se explica cada uno de estos métodos.

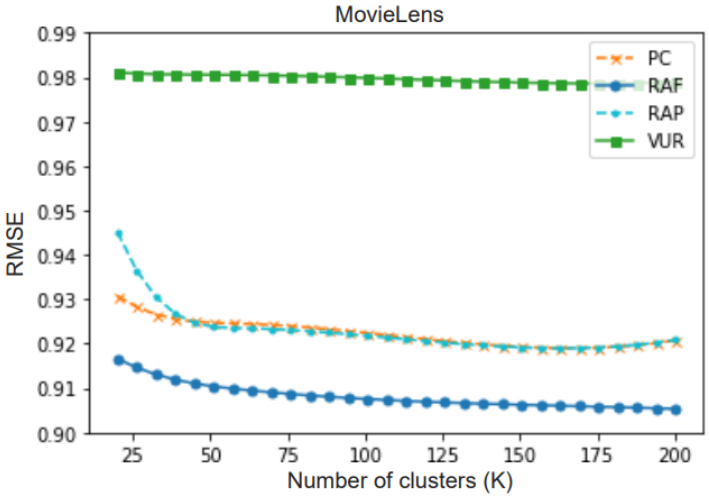


Figura 24 Calidad de predicción con RMSE. Figura tomada de (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020)

Resultados de calidad de recomendación

La **figura 25** indica los resultados de calidad de recomendación con el dataset MovieLens. La métrica F1 obtenida cuando el método propuesto se ejecuta utilizando un número diverso de grupos (Hurtado, Gutiérrez,

Bobadilla, & Alonso, 2020). Mientras F1 es mayor, la calidad es mejor. Se puede apreciar que el método incorporado RAF supera levemente en términos generales a los demás métodos, aunque compite con RAP y PC cuando se tiene una cantidad menor a 50 grupos. En (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020) se indica que el método RAF en cuanto a precision presenta los mejores resultados y esto es lo más relevante en el dominio del filtrado colaborativo.

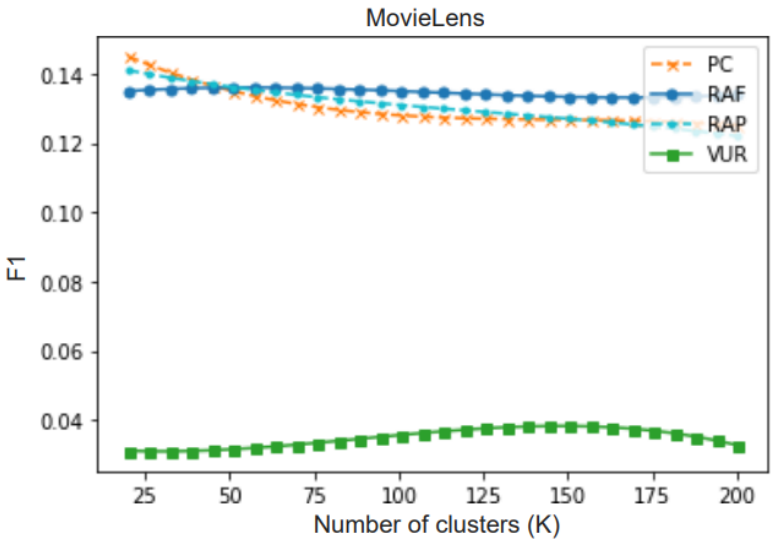


Figura 25 Calidad de recomendación con F1. Figura tomada de (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020)

Por lo tanto, debido a los resultados presentados, en el sistema informático desarrollado en este trabajo, se ha incorporado el modelo de recomendación RAF que combina BNMF, clustering y un método de predicción basado en la agregación de factores probabilísticos.

3.2 Verificación del sistema

El objetivo de la administración de la calidad es asegurar que el proyecto satisfaga las necesidades para las cuales inició. Una herramienta eficaz para la planificación y verificación de la calidad es una lista de verificación de requerimientos o procesos. A continuación, se presenta la lista de verificación para el sistema web.

Las pruebas funcionales fueron guiadas por el tutor del proyecto de titulación el Ingeniero Remigio Hurtado en el grupo de investigación Giiata, hemos escogido algunos usuarios, pero por la extensión del documento se presenta las figuras más importantes, del grupo de usuarios que colaboró con las pruebas hemos escogido al usuario Jessica Pinduisaca.

Tabla 7 Lista de Verificación Sistema Web

Fecha de Revisión: 29-12-2020 **Firma: Jessica Pinduisaca**					
Numeración	Función	**Calificación**			Observación
		Cumple	Cumple Parcial	No Cumple	
1	Log in	X			
2	Crear Nuevo libro	X			
2.1	Editar Libro	X			
3	Crear Nuevo Usuario	X			
3.1	Editar Usuario	X			
4	Revisar Reportes	X			
4.1	Imprimir reportes	X			
5	Revisar información de perfil	x			

Tabla 8 Lista de Verificación Sistema Móvil

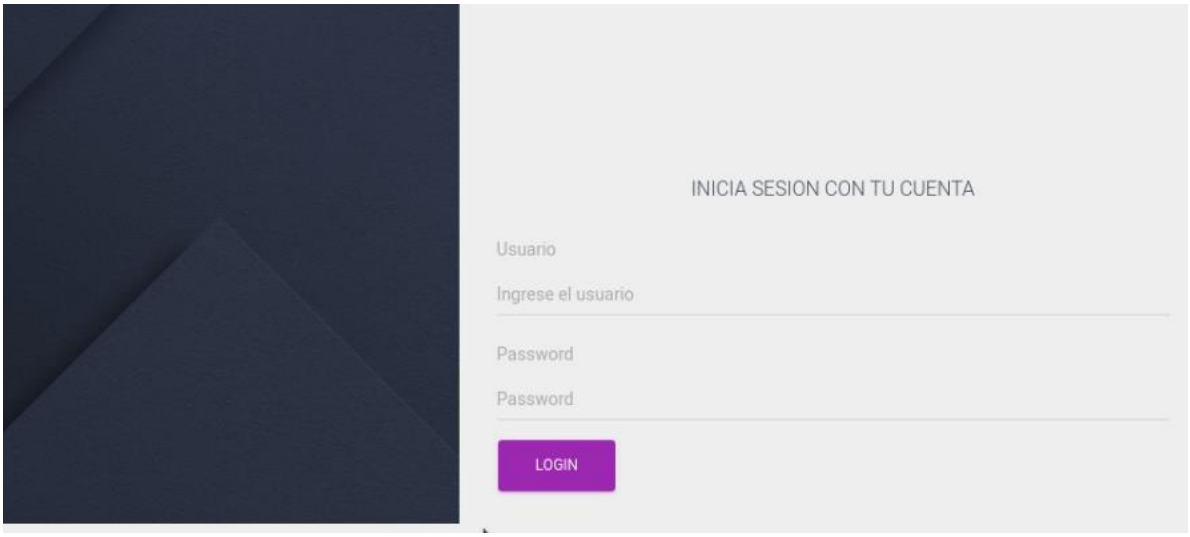
Fecha de Revisión: 29-12-2020 **Firma: Jessica Pinduisaca**					
Id Función	Función	**Calificación**			Observación
		Cumple	Cumple Parcial	No Cumple	
1	Registro	X			
1.1	Login	X			
1.2	Mensaje de Inicio	X			
1.3	Seleccionar Lista de categorías	X			
3	Crear Nuevo Grupo	X			
3.1	Agregar usuarios al grupo	X			
4	Revisar Perfil	X			
4.1	Editar Perfil	X			
5	Visualizar lista de libros recomendados al usuario	X			
5.1	Ingresar a libros recomendado al usuario	X			
6	Revisar lista de comentarios	X			
6.1	Agregar nuevo comentario	X			
7	Agregar nuevo rating al libro.	X			
8	Ingresar a grupos recomendados	X			

9	Cerrar Sesión	x			

3.3 Pruebas funcionales del sistema web

Tabla 9 Prueba Funcional Sistema Web - Login

Id Caso de Prueba:	LON-1 (Log-In)
Objetivo:	Probar el funcionamiento del formulario login
Descripción:	El administrador web ingresará el “usuario y contraseña” dada, para verificar el funcionamiento.
Actores:	Administrador Web
Precondiciones:	Que el administrador web a conectarse este registrado en el sistema.
Poscondiciones:	La aplicación web mostrara la página de inicio “Dashboard”
Escenarios: **1.** Por medio del web services ‘usuario/loginAdmin’ se valida el ingreso a la aplicación web.	
Condiciones (Datos de Entrada)	Usuario:admin Password:admin
Resultado Esperado	Ingreso correctamente, podrá visualizar y navegar por todas las ventanas disponibles en el menú.
Ejecución	
Fecha de ejecución	**29/12/2020**
Responsable de ejecución	**Jessica Pinduisaca**
Resultado Real	



¿Prueba aprobada? **SI/NO**	**SI**
Observaciones	Se realizó el proceso correctamente sin novedad alguna.

Tabla 10 Prueba Funcional Sistema Web - Crear y Editar Libro

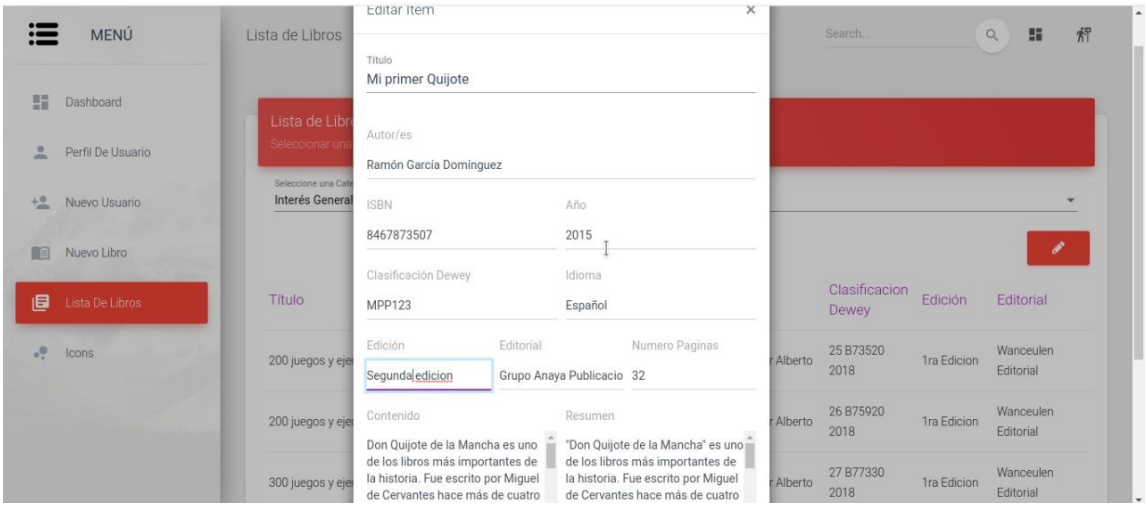
Id Caso de Prueba:	ICE-1 (ítems crear y editar)
Objetivo:	Crear y editar un ítem (libro).
Descripción:	El administrador web creará un nuevo libro ingresando desde el menú lateral izquierdo en la opción **Nuevo Libro** e ingresar la información solicitada y hacer clic en el botón guardar. Para editar un libro el administrador web ingresará desde el menú lateral en la opción **Lista de Libros**, debe seleccionar una categoría para listar los libros, una vez listados los libros seleccione el libro y haga clic en el botón superior derecho, se visualizará un popup con la información del libro.
Actores:	Administrador Web
Precondiciones:	Tener toda la información del nuevo libro a ingresar.
Poscondiciones:	La aplicación web arrojara un nuevo libro creado o un libro editado.

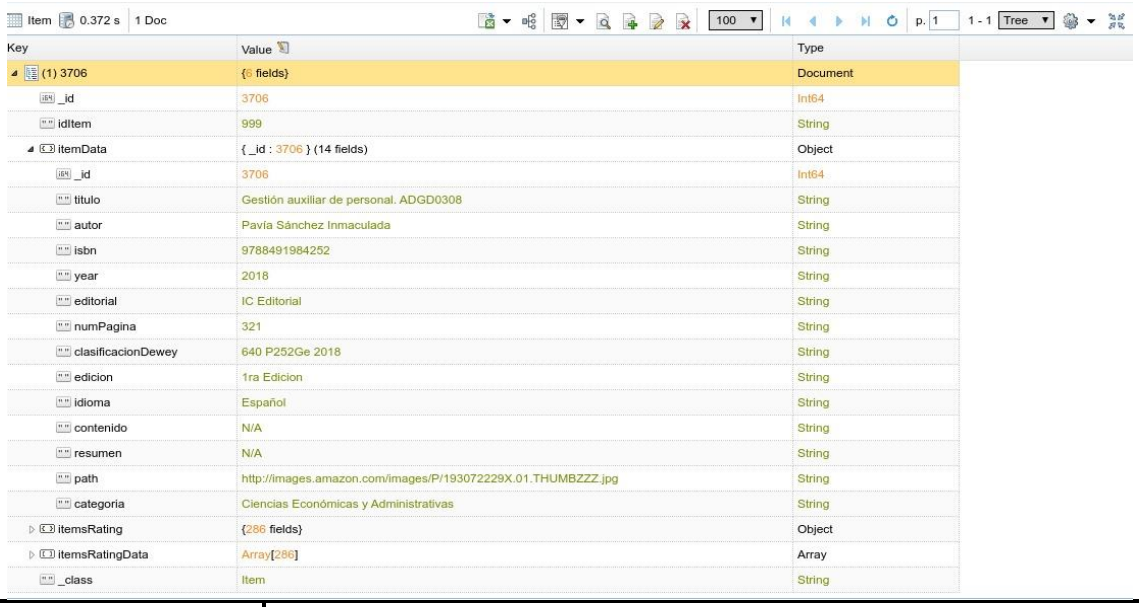
Escenarios:

1. El formulario Nuevo Libro, consume el web services 'item/nuevoItem'.
2. Tabla Lista de Libros, se selecciona la categoría para listar los libros a los que pertenece.
3. Al Editar un libro se consume el web services 'item/actualizarItem'

Condiciones (Datos de Entrada)	Título, Autor(es), ISBN, Year, Clasificación Dewey, Edición, Idioma, Contenido, Resumen, Path, Editorial, Categoria, Número de Páginas

Resultado Esperado	Acción Crear nuevo libro exitoso. Acción editar libro exitoso.
Ejecución	
Fecha de ejecución	**29/12/2020**
Responsable de ejecución	**Jessica Pinduisaca**
Resultado Real	





¿Prueba aprobada? SI/NO	**SI**
Observaciones	Se realizó el proceso correctamente sin novedad alguna.

Tabla 11 Prueba Funcional Sistema Web - Dashboard

Id Caso de Prueba:	DSH-1 (Dashboard)
Objetivo:	Analizar Dashboard
Descripción:	El administrador web podrá analizar las gráficas propuestas en la ventana Dashboard
Actores:	Administrador web
Precondiciones:	El administrador web debe iniciar sesión.
Poscondiciones:	La aplicación web arrojará las gráficas de libros y grupos.
Escenarios: **1.** El administrador inicia sesión. **2.** El administrador analiza las gráficas y toma de decisiones.	
Condiciones (Datos de Entrada)	Usuario, password
Resultado Esperado	Análisis de Gráficas
Ejecución	
Fecha de ejecución	**29/12/2020**

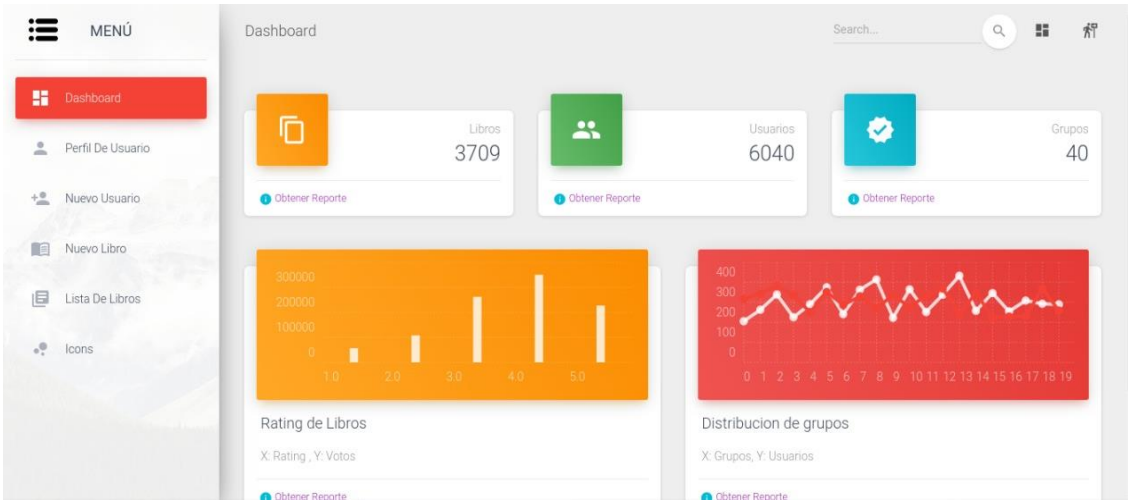
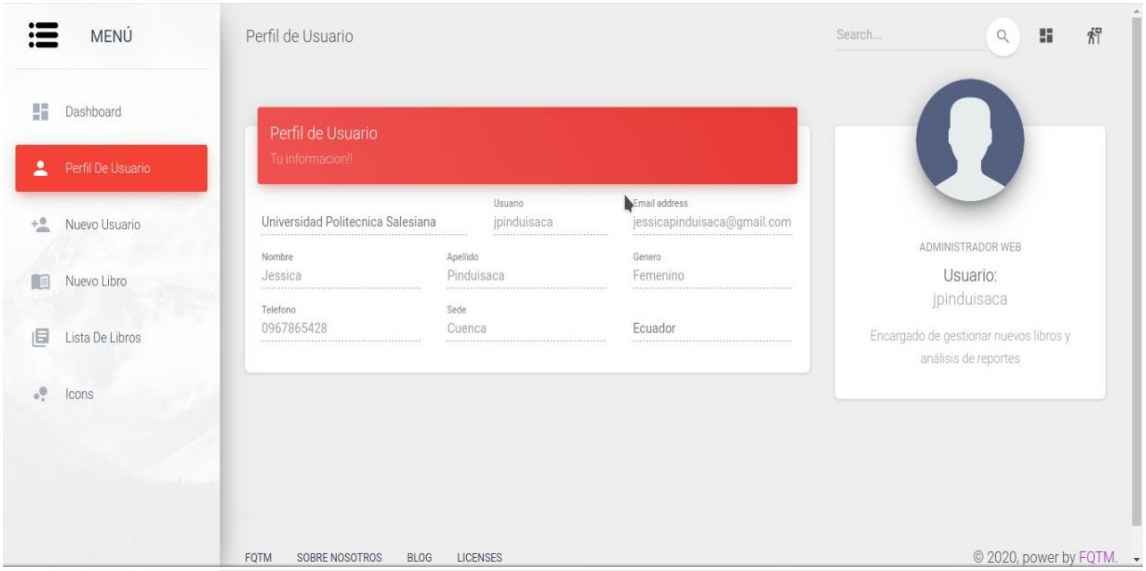
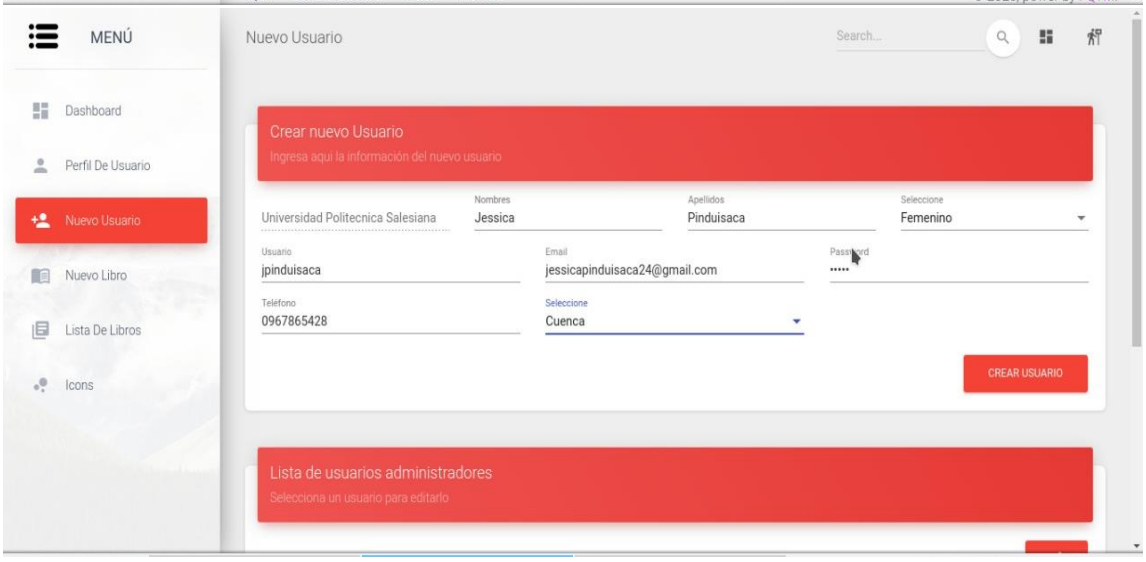
Responsable de ejecución	**Jessica Pinduisaca**
Resultado Real	
	
¿Prueba aprobada? SI/NO	**SI**
Observaciones	Se realizó el proceso correctamente sin novedad alguna.

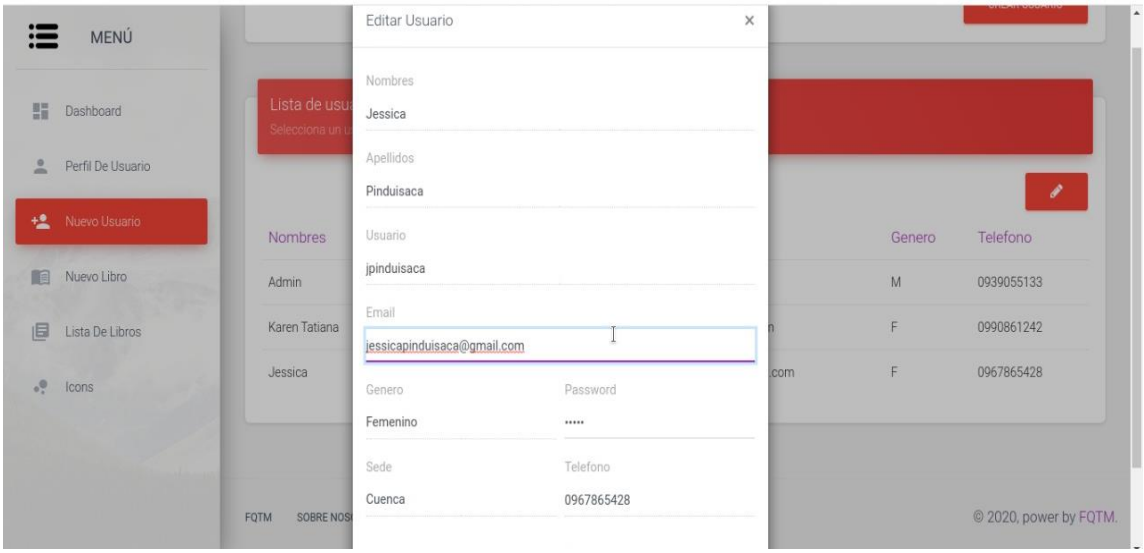
Tabla 12 Prueba Funcional Sistema Web - Administrar Usuario

Id Caso de Prueba:	USU-1 (Crear usuario, Editar Usuario, Perfil de Usuario)
Objetivo:	Administrar Usuario
Descripción:	El administrador web podrá crear un nuevo usuario o editar, de igual manera podrá visualizar la información de perfil.
Actores:	Administrador web
Precondiciones:	Tener la información del nuevo usuario.
Poscondiciones:	La aplicación devolverá una lista de usuarios administradores.
Escenarios: **1.** El administrador revisa su perfil de usuario. **2.** El administrador puede crear un nuevo usuario. 3. El administrador puede editar un usuario administrador.	
Condiciones (Datos de Entrada)	Crear Usuario: • Nombres, apellidos, usuario, email, password, sede, teléfono, genero. Editar Usuario: • Email, password
Resultado Esperado	Acción Crear un nuevo usuario exitoso. Acción Editar usuario exitoso Acción Ver perfil exitoso

Ejecución	
Fecha de ejecución	29/12/2020
Responsable de ejecución	Jessica Pinduisaca
Resultado Real	





	
¿Prueba aprobada? SI/NO	SI
Observaciones	Se realizó el proceso correctamente sin novedad alguna.

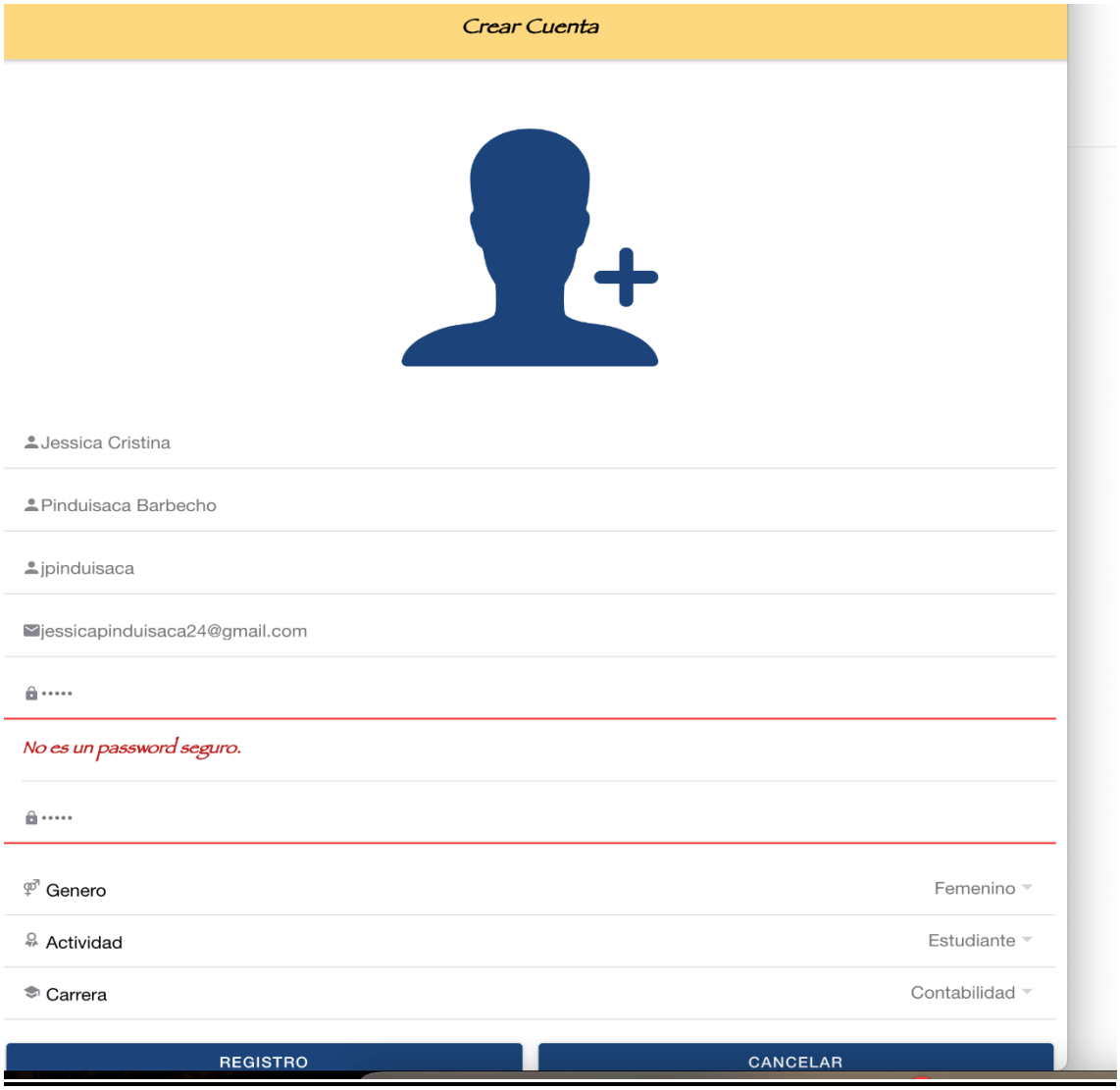
3.4 Pruebas funcionales del sistema móvil

A continuación, se presenta la lista de verificación para el sistema móvil.

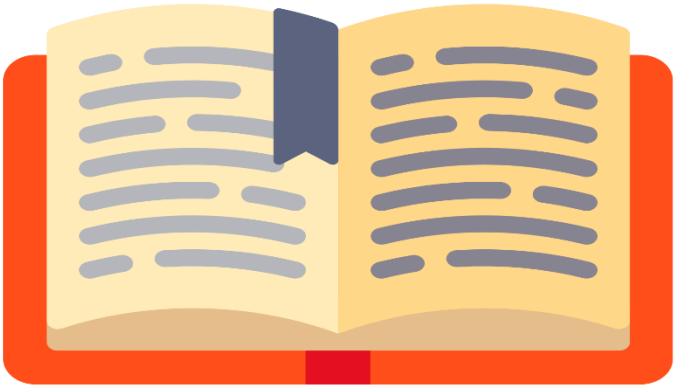
Tabla 13 Prueba Funcional Sistema Móvi - Login y Registrol

Id Caso de Prueba:	LOGREG-1 (Login – Registro)
Objetivo:	Validar formulario login y registro
Descripción:	Se utilizará la aplicación móvil para registrar un nuevo usuario que permita iniciar sesión en la aplicación.
Actores:	Estudiante o Docente
Precondiciones:	El estudiante/docente debe registrarse antes de iniciar sesión.
Poscondiciones:	La aplicación móvil devolverá los libros recomendados al usuario.
Escenarios: **1.** El estudiante/docente se registra en la aplicación móvil. **2.** El estudiante/docente inicia sesión con el usuario generado. **3.** Mensaje de inicio. **4.** Al ser un usuario nuevo, la aplicación solicita que escojan las categorías que más se asemeje a sus gustos.	
Condiciones (Datos de Entrada)	Formulario de Registro: • Nombres, apellidos, usuario, email, password, confirmar password, genero, actividad, carrera Formulario Login:

	• Usuario, password
Resultado Esperado	Acción Registro exitoso Acción Login exitoso
Ejecución	
Fecha de ejecución	29/12/2020
Responsable de ejecución	Jessica Pinduisaca
Resultado Real	



GRS-Biblioteca



Crea una cuenta gratis.

LOGIN

REGISTRO

SALIR

Mensaje de Inicio

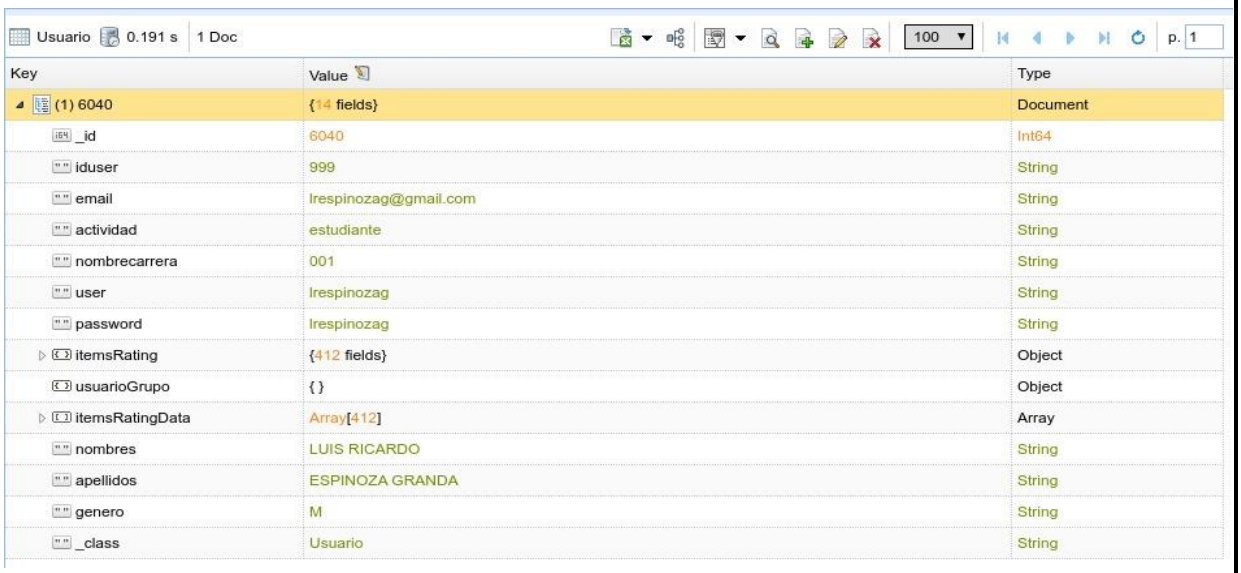
Te damos la bienvenida al GRS Biblioteca UPS

Para mejorar tu experiencia necesitamos que nos ayudes seleccionando las categorías que mas te gustan

OK

Categorias

- [x] Ciencias Exactas y Naturales
- [ ] Arquitectura, Urbanismo y Diseño
- [ ] Bellas Artes, Artes Visuales y Ciencias Semióticas
- [ ] Arquitectura, Urbanismo y Diseño
- [ ] Cengage Learning
- [ ] Ciencias Biológicas, Veterinarias y Silvoagropecuarias
- [x] Ciencias de la Información y de la Comunicación
- [ ] Ciencias de la Salud
- [x] Ciencias Económicas y Administrativas
- [x] Ciencias Sociales



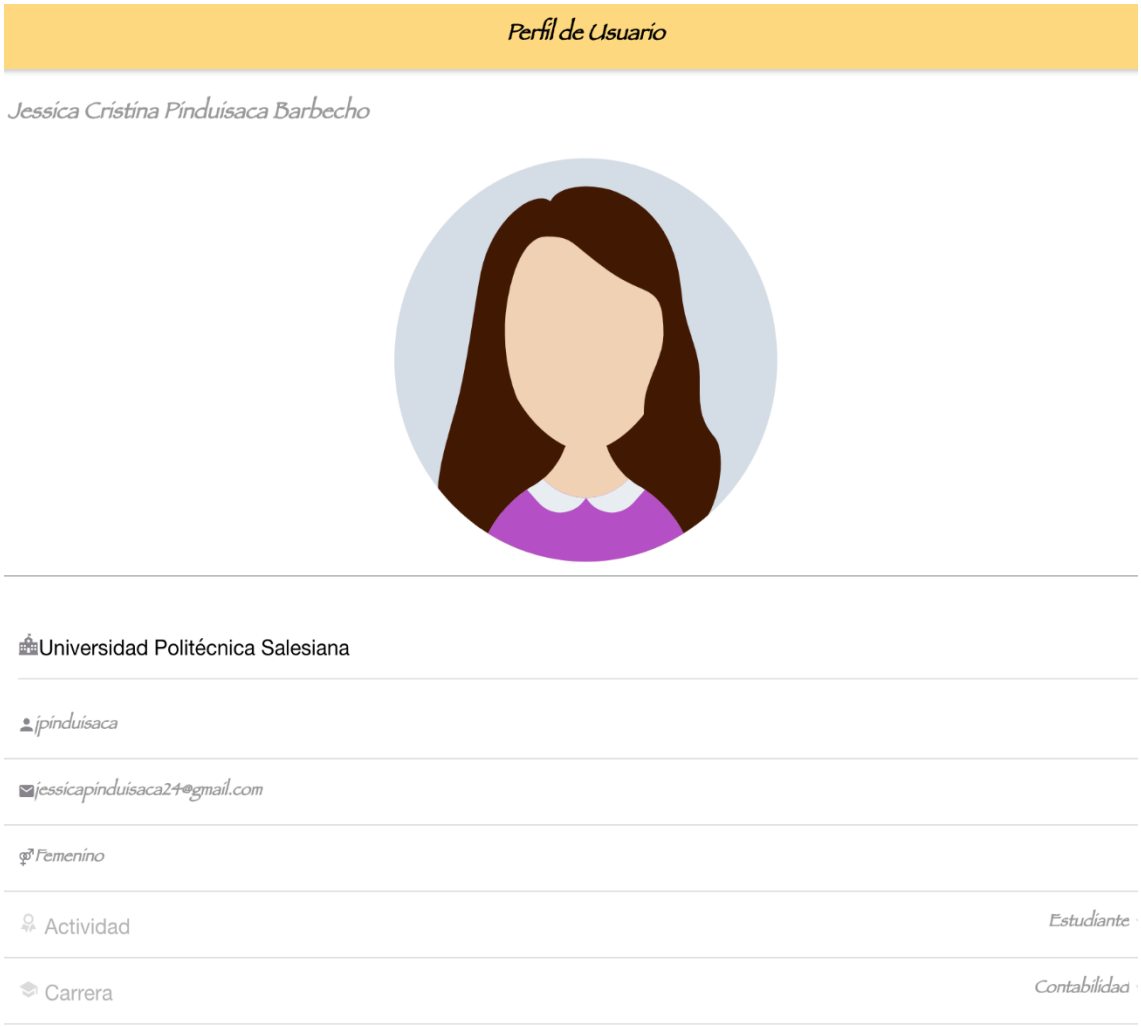
Usuario 0.191 s 1 Doc 100 p. 1

Key	Value	Type
(1) 6040	{14 fields}	Document
_id	6040	Int64
iduser	999	String
email	lrespinozag@gmail.com	String
actividad	estudiante	String
nombrecarrera	001	String
user	lrespinozag	String
password	lrespinozag	String
itemsRating	{412 fields}	Object
usuarioGrupo	{ }	Object
itemsRatingData	Array[412]	Array
nombres	LUIS RICARDO	String
apellidos	ESPINOZA GRANDA	String
genero	M	String
_class	Usuario	String

¿Prueba aprobada? SI/NO	SI
Observaciones	Se realizó el proceso correctamente sin novedad alguna.

Tabla 14 Prueba Funcional Sistema Móvil - Perfil

Id Caso de Prueba:	PER-1 (Perfil)
Objetivo:	Probar el funcionamiento del perfil de usuario.
Descripción:	El usuario móvil realizará las pruebas de funcionamiento en la ventana Perfil.
Actores:	Estudiante/Docente
Precondiciones:	Que el estudiante/docente a conectarse este registrado en el sistema.
Poscondiciones:	El web services arrojará la información del usuario que inicio sesión en la aplicación móvil.
Escenarios: **1.** El estudiante/docente hace clic en el botón perfil. **2.** El estudiante/docente revisa la información ingresada. **3.** Si desea editar la información, puede realizarla haciendo clic en el botón editar y guardar los cambios.	
Condiciones (Datos de Entrada)	Usuario Password
Resultado Esperado	Visualizar información del estudiante/docente ingresada cuando realizó un registro. Visualizar cambios realizados cuando se realizó la acción de editar.
Ejecución	
Fecha de ejecución	29/12/2020
Responsable de ejecución	Jessica Pinduisaca
Resultado Real	



¿Prueba aprobada? SI/NO	SI
Observaciones	Se realizó el proceso correctamente sin novedad alguna.

Tabla 15 Prueba Funcional Sistema Móvil - Crear Grupo

Id Caso de Prueba:	CRGRO-1
Objetivo:	Probar el funcionamiento de crear un grupo.
Descripción:	El estudiante/docente realiza las pruebas funcionales para crear un nuevo grupo de usuarios.
Actores:	Estudiante/Docente
Precondiciones:	Que el estudiante/docente hayan iniciado sesión en la aplicación móvil.

Poscondiciones:	La aplicación arrojara un nuevo grupo conformado por usuarios y libros.

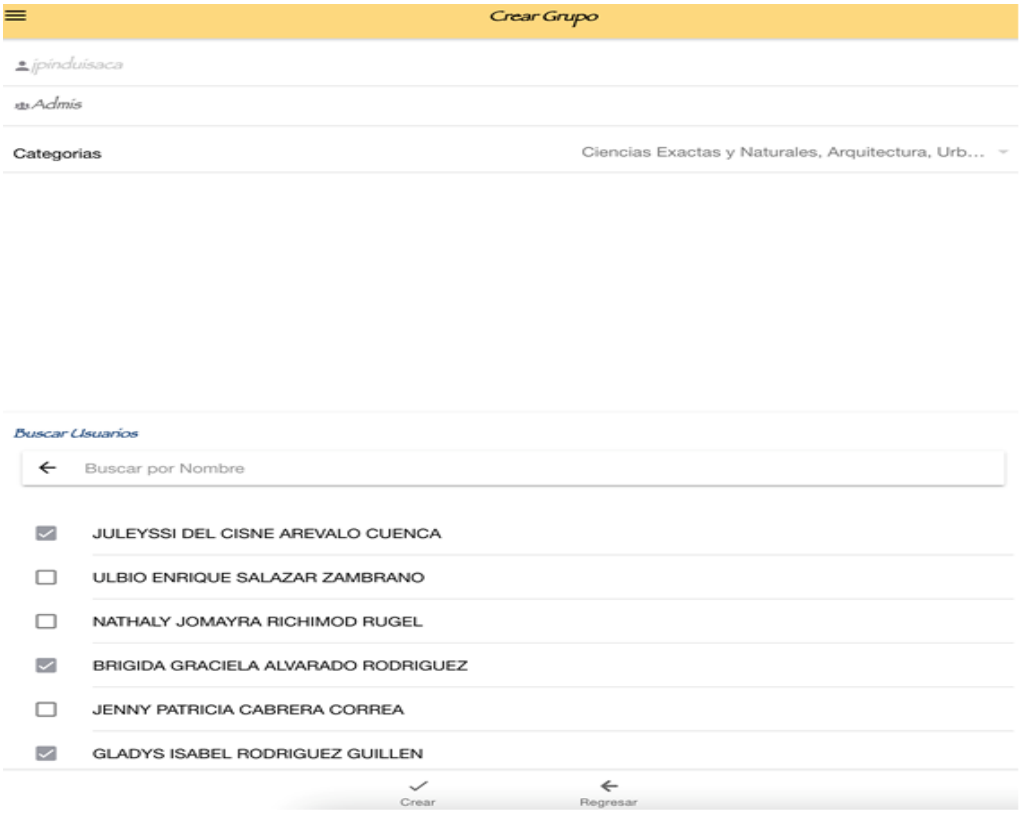
Escenarios:

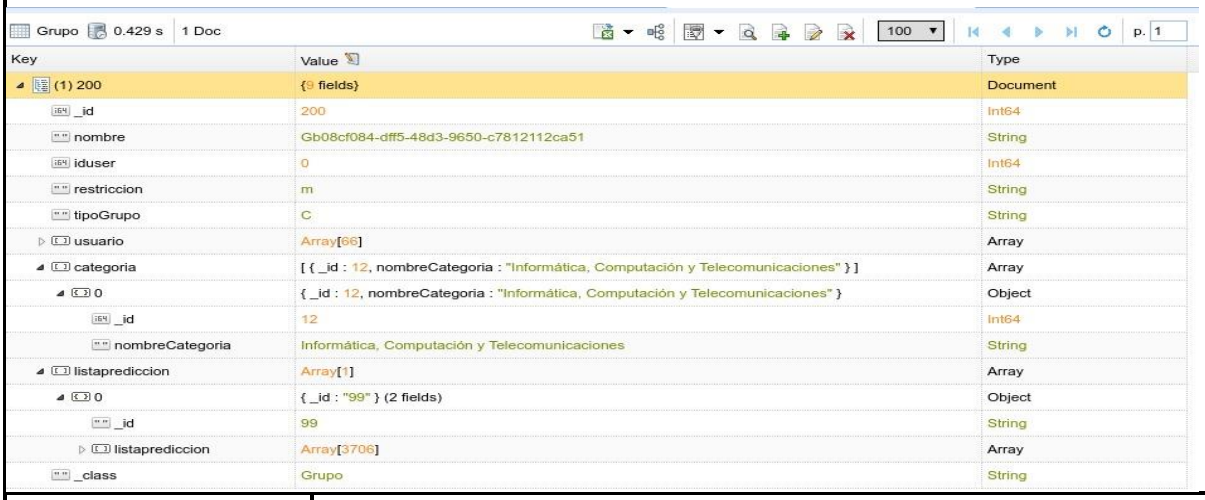
1. El estudiante/docente hará clic en el icono menú, opción crear grupo.
2. Agregar la información que solicita el formulario.
3. Clic en el botón crear.

Condiciones (Datos de Entrada)	• Nombre del grupo, seleccionar categoría, seleccionar los usuarios que pertenecen al grupo.
Resultado Esperado	Crear nuevo grupo exitoso.

Ejecución	
Fecha de ejecución	29/12/2020
Responsable de ejecución	Jessica Pinduisaca

Resultado Real





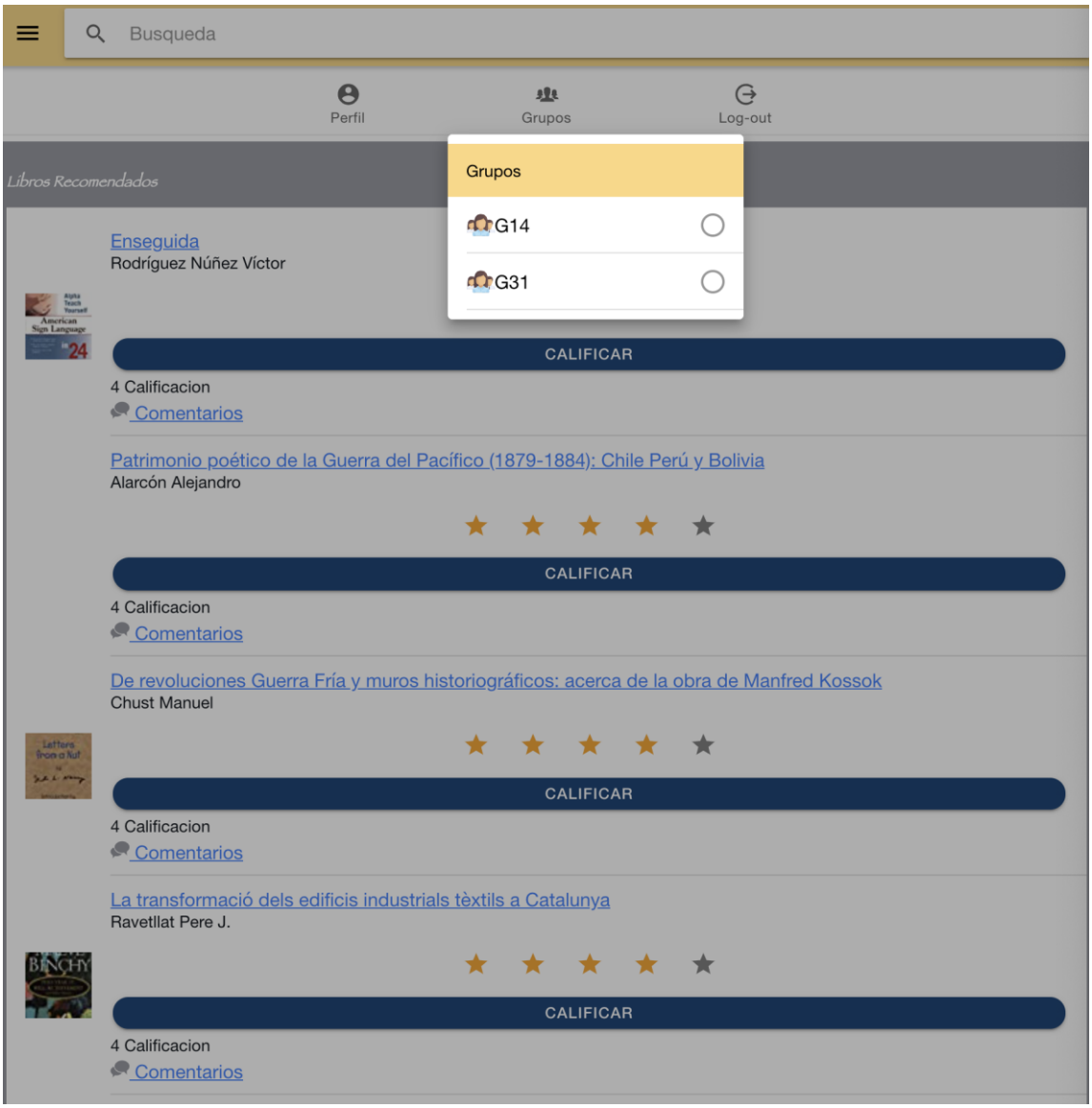
¿Prueba aprobada? SI/NO	SI
Observaciones	Se realizó el proceso correctamente sin novedad alguna.

Tabla 16 Prueba Funcional Sistema Móvil - Home

Id Caso de Prueba:	HOM-1
Objetivo:	Revisar información recomendada al usuario en la página de inicio.
Descripción:	El estudiante/docente al iniciar sesión tiene la página de inicio, que le brinda una lista con libros recomendados, según las categorías que seleccionó al iniciar sesión por primera vez.
Actores:	Estudiante/Docente
Precondiciones:	Que el estudiante/docente inicie sesión en la aplicación móvil.
Poscondiciones:	La aplicación móvil mostrará libros recomendados al usuario.
Escenarios: **1.** El usuario visualizará la lista de libros recomendados.	
Condiciones (Datos de Entrada)	Usuario, password
Resultado Esperado	Visualizar libros recomendados e ingresar a cada libro para revisar la información.
Ejecución	
Fecha de ejecución	29/12/2020

Responsable de ejecución	Jessica Pinduisaca

Resultado Real



¿Prueba aprobada? SI/NO	SI
Observaciones	Se realizó el proceso correctamente sin novedad alguna.

Tabla 17 Prueba Funcional Sistema Móvil - Grupos Recomendados

Id Caso de Prueba:	GRUREC-1
Objetivo:	Obtener grupos recomendados.

Descripción:	El estudiante/docente visualizará en la parte superior tres iconos, hacer clic en el icono grupos, se listará los grupos recomendados y los grupos creados, seleccionar uno para ingresar y visualizar toda la información del grupo como: nombre, categorías, usuarios, libros.
Actores:	Estudiante/Docente
Precondiciones:	Que el estudiante/docente haya iniciado sesión y tenga un grupo creado.
Poscondiciones:	La aplicación móvil mostrará los grupos creados y recomendados.

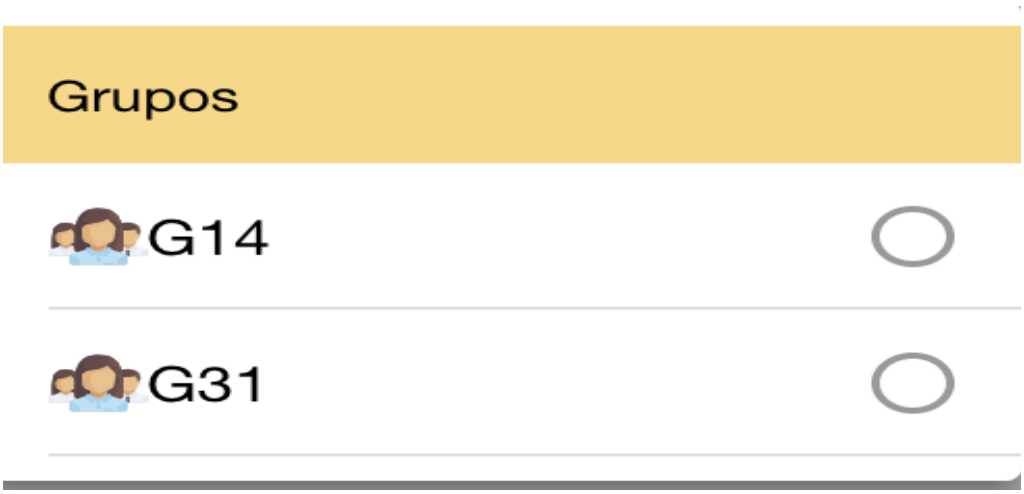
Escenarios:

1. El usuario hace clic en el icono grupos.
2. Se visualizará una lista de grupos recomendados, seleccionar el grupo que desee para revisar la información.
3. Ingresa al grupo

Condiciones (Datos de Entrada)	Usuario, password
Resultado Esperado	Visualizar e ingresar a los grupo recomendados o creados por el estudiante/docente.

Ejecución	
Fecha de ejecución	29/12/2020
Responsable de ejecución	Jessica Pinduisaca

Resultado Real



Informacion del Grupo

Nombre Grupo:

Restriccion:

Libros Recomendados para el grupo

La transformació dels edificis industrials tèxtils a Catalunya
Ravetllat Pere J.

CALIFICAR

4 votos

promedio 4 de 5

Comentarios

De revoluciones Guerra Fría y muros historiográficos: acerca de la obra de Manfred Kossok
Chust Manuel

CALIFICAR

4 votos

promedio 4 de 5

Comentarios

El hogar infernal: 50 películas esenciales de mansiones y casas encantadas
Cuadrado Alvarado Alfonso

CALIFICAR

4 votos

promedio 4 de 5

Comentarios

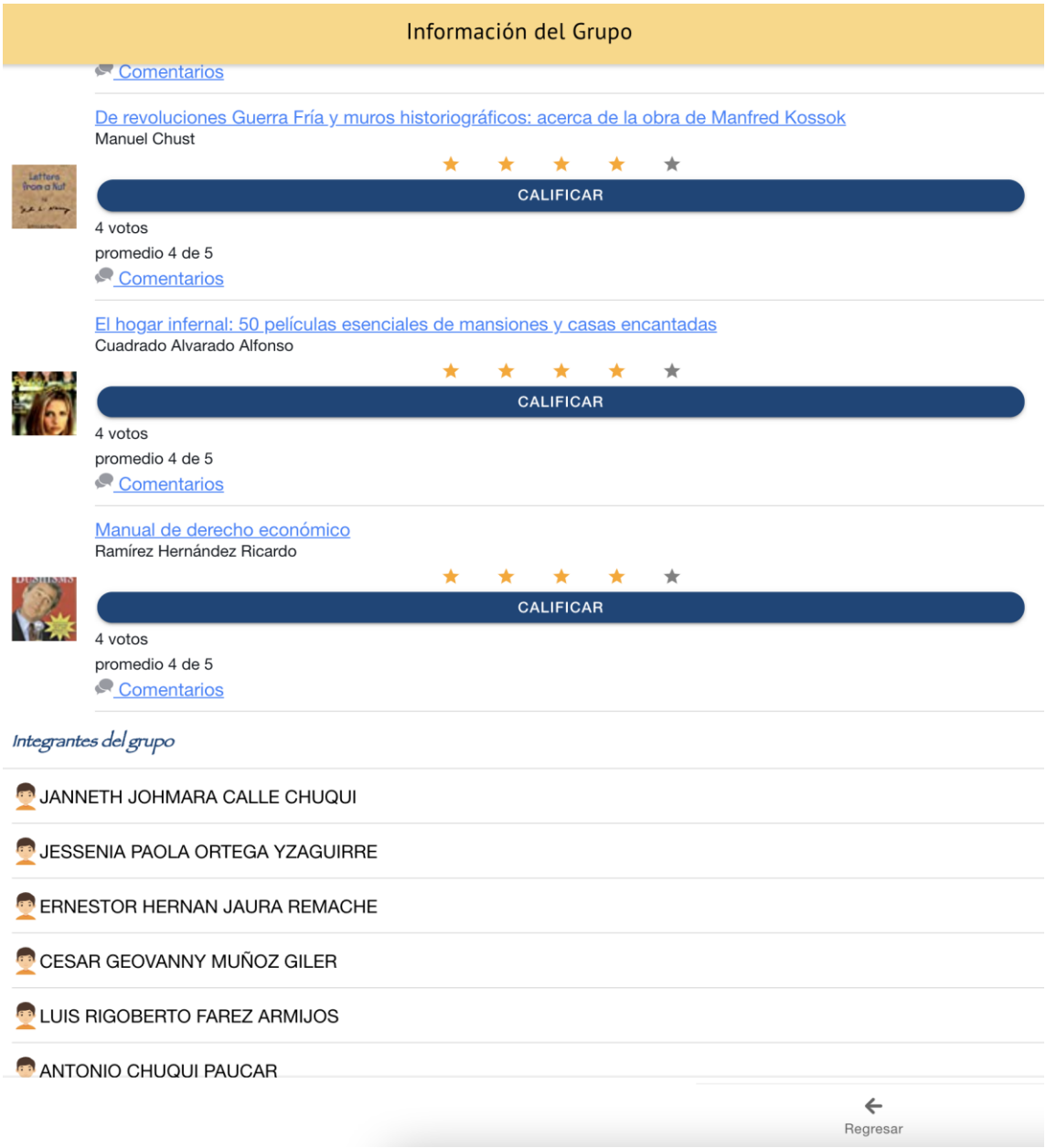
Manual de derecho económico
Ramírez Hernández Ricardo

CALIFICAR

4 votos

promedio 4 de 5

Comentarios



¿Prueba aprobada? SI/NO	SI
Observaciones	Se realizó el proceso correctamente sin novedad alguna.

Tabla 18 Prueba Funcional Sistema Móvil - Comentarios

Id Caso de Prueba:	COM-1
Objetivo:	Crear y visualizar comentarios
Descripción:	Cada libro tiene la opción de comentarios, puede hacer clic en el y se visualizará una nueva ventana con los comentarios, usuarios y fecha que se han realizado. En la misma venta se puede crear un nuevo

	comentario.
Actores:	Estudiante/Docente
Precondiciones:	El usuario debe ingresar a los comentarios del libro.
Poscondiciones:	La aplicación móvil devolverá una lista de comentarios por cada libro

Escenarios:

1. El estudiante/docente ingresa al comentario del libro.
2. Revisa los comentarios que se han realizado al libro.
3. Agrega un nuevo comentario al libro.

Condiciones (Datos de Entrada)	Nuevo comentario
Resultado Esperado	Visualizar y crear nuevos comentarios a los libros.

Ejecución

Fecha de ejecución	29/12/2020
Responsable de ejecución	Jessica Pinduisaca

Resultado Real

¿Prueba aprobada? SI/NO	SI
Observaciones	Se realizó el proceso correctamente sin novedad alguna.

Tabla 19 Prueba Funcional Sistema Móvil - Rating

Id Caso de Prueba:	RAT-1
Objetivo:	Dar votación a libros
Descripción:	El estudiante/docente puede dar su votación a cada libro, para ello haga clic en las estrellas y clic en el botón calificar.
Actores:	Estudiante/Docente
Precondiciones:	El estudiante revisar la información del libro para dar el voto.
Poscondiciones:	Al dar un voto a un libro, se almacenará la nueva información y el GRS te recomendará un nuevo libro.
Escenarios: **1.** El estudiante/docente ingresa a revisar la información de los libros. **2.** Da su votación al libro.	
Condiciones (Datos de Entrada)	Rating a libros
Resultado Esperado	Nuevos libros recomendados
Ejecución	

Fecha de ejecución	29/12/2020
Responsable de ejecución	Jessica Pinduisaca
Resultado Real	
¿Prueba aprobada? SI/NO	SI
Observaciones	Se realizó el proceso correctamente sin novedad alguna.

Se ha realizado la verificación y validación del modelo de aprendizaje como las funciones del sistema. Para ello se ha validado el modelo de recomendación mediante métricas estándares y las funciones del sistema se han verificado y validado mediante la lista de verificación y pruebas funcionales.

CONCLUSIONES

En el presente proyecto de titulación se ha integrado el modelo de recomendación BNMF orientado a grupos de usuarios (Hurtado, Gutiérrez, Bobadilla, & Alonso, 2020), estudiado y publicado por el grupo de investigación GIIATA al que pertenecemos. Se destacan dos aplicaciones de esta tesis: 1) una aplicación web que realiza la gestión administrativa; y 2) una aplicación móvil que será para el docente o estudiante, permitiéndole obtener libros recomendados tanto para el usuario como para grupos de usuarios al que pertenece.

Al término de este proyecto, se tienen las siguientes conclusiones:

- Se cumplió con el objetivo principal, creando el sistema recomendador para grupos utilizando la reducción de dimensionalidad (BNMF) para aliviar el problema de dispersión de datos en filtrado colaborativo, clustering y método de recomendación a grupos mediante el enfoque de factores de usuarios de grupos.
- Se logró que los usuarios puedan instalar la aplicación móvil en dispositivos con sistemas android y ios.
- Se logró integrar la aplicación en la nube.
- El software libre permitió el desarrollo, para la creación e integración de los componentes o módulos del proyecto.
- El desarrollo se realizó en maquina locales, la integración en la nube se realizó en servidores virtuales como OVH y VULTR, utilizando servidores con sistema Linux.
- Ionic permitió el desarrollo de una aplicación móvil hibrida permitiendo desarrollar una sola interfaz para los dos sistemas móviles.
- Angular permitió el desarrollo de la aplicación web para la administración de usuarios administradores, que pueden llegar a tomar decisiones dentro de la organización, tomando en cuenta las gráficas disponibles en la página.
- Se implementó Apache como servidor web para alojar la página desarrollada en angular para la administración de la organización.
- Se trabajó con base de datos MongoDB, ya que facilita el manejó de objetos al no ser una base de datos relacional. Permite flexibilidad a cambios que se presentan en el desarrollo de la aplicación.
- Para el desarrollo de backend se trabajó con la herramienta Spring boot, simplificando la compleja configuración de librerías, dependencias, despliegue, integración con la base de datos. Preocupándonos netamente del desarrollo del core de la aplicación.
- Los grupos conformados están distribuidos equitativamente, por ejemplo: no existen grupos conformados con apenas 2 usuarios y otros con 1000 usuarios.
- El entrenamiento tiene un alto consumo de recursos del pc, por lo cual el entrenamiento es local y no online. Las recomendaciones son online.

En definitiva, se puede concluir: “la implementación de la solución que integra varios componentes o módulos, que en conjunto conforman todo un sistema de recomendador para grupos, ayuda de gran manera a la organización (en la toma de decisiones basada en gráficas de muestreo), el usuario tiene productos, grupos recomendados, lo cual mejora su estilo y calidad de vida”.

RECOMENDACIONES

Al término de este proyecto, tenemos las siguientes recomendaciones:

- Utilizar herramientas libres, estables y actualizadas para obtener un mejor resultado, una mayor posibilidad de desarrollo, escalabilidad a nuevas versiones, flexibilidad de integración.
- Manejar repositorios para tener un mejor control de versiones en el desarrollo de las aplicaciones, con esto podemos prevenir la perdida de código y el doble esfuerzo del equipo de trabajo.
- Utilizar servidores locales linux para el desarrollo, cuando se realice la puesta en producción utilizar servidores VPS con sistemas Linux que permitan escalabilidad, con esto nos evitamos el gasto en equipos físicos. Si el proyecto no es rentable simplemente terminamos el contrato con nuestro proveedor de VPS y no tenemos el problema de tener equipos físicos.
- Cuando se maneje una base de datos MongoDB manejar la autentificación, no dejar la configuración por defecto, con esto prevenimos de ataques que alteren la información ya migrada en la base de datos.

Continuar con la investigación para en un futuro poder realizar el entrenamiento en tiempo real (activo), como lo realizan plataformas de Amazon, Netflix, agregando más filtros de recomendación (demográficos, contenido, colaborativo de popularidad).

TRABAJO FUTURO

- Realizar un chat interno entre usuarios.
- Incluir filtrados: demográfico, basado en contenido, basado en popularidad dentro de las métricas para recomendar un grupo.
- Reservar ítems.
- Notificaciones de grupos recomendados para los usuarios, agregar si acepta o no el grupo recomendado.
- Migrar usuarios administradores desde archivos.
  Realizar recomendación de grupos en tiempo real dependiendo de la votación de los usuarios.
  Medir la calidad de las recomendaciones a un grupo de usuarios y mostrar en gráfica a los administradores.

REFERENCIAS

Abreu-Lee, Y., Infante-Abreu, M. B., Delgado Fernández, T., & Delgado Fernández, M. (2013). Modelo de vigilancia tecnológica apoyado por recomendaciones basadas en el filtrado colaborativo. *Ingenieria Industrial*(2).

Amazon. (n.d.). From https://aws.amazon.com/es/nosql/

Angular. (n.d.). *Angular*. From https://angular.io/guide/architecture

Bobadilla, J., Bojorque, R., Hernando, A. E., & Hurtado, R. (2017). Recommender Systems Clustering Using Bayesian Non Negative Matrix Factorization. *IEEE Access, 6*, 3549-3564. doi:10.1109/ACCESS.2017.2788138.

Bojorque, R., Hurtado, R., & Inga, A. (2019). A Comparative Analysis of Similarity Metrics on Sparse Data for Clustering in Recommender Systems. *Advances in Artificial Intelligence, Software and Systems Engineering, 787*, 291-299. doi:https://doi.org/10.1007/978-3-319-94229-2_28

COBA DE LA TORRE, C. A., & GÓMEZ GÓMEZ, P. M. (2014). *Análisis, diseño, construcción e implementación de un Geoportal con las herramientas MongoDB, JSON y GEOJSON para el proyecto IDE-UPS.* Quito.

COBA, J. R. (2016). *DISEÑO E IMPLEMENTACIÓN DE UNA APLICACIÓN MOVIL PARA EL PROCESO DE RESERVACIÓN DE HABITACIONES EN EL HOSTAL QUINTA SUR.* Guayaquil.

Galán Nieto, S. M. (2007). Filtrado Colaborativo y Sistemas de Recomendación. *Universidad Carlos III de Madrid*.

Galvis, J. P. (2015). *Propuesta de notación gráfica para el modelo orientado a documentos MongoDB.* Bogotá.

Ghobar, E. W. (2017). *Un sistema de recomendación basado en perfiles generados por agrupamiento y asociaciones.* Valencia.

GroupLens. (2000). *MoviLens Dataset.* From http://files.grouplens.org/datasets/movielens/ml-1m-README.txt

Hernando, A., Bobadilla, J., & Ortega, F. (2016). A non negative matrix factorization for collaborative filtering recommender systems based on a Bayesian probabilistic model. *Knowledge-Based Systems, 97*(C), 188-202.

Hurtado Ortiz, R. (2020). *Recomendación a grupos de usuarios usuando conceptos de singularidades.* Madrid.

Hurtado Ortiz, R., Bojorque Chasi, R., & Inga Chalco, C. (2019). Clustering-Based Recommender System: Bundle Recommendation Using Matrix Factorization to Single User and User Communities. *Advances in Artificial Intelligence, Software and Systems Engineering, 787*, 330-338. doi:https://doi.org/10.1007/978-3-319-94229-2_32

Hurtado, R., Bobadilla, J., Gutiérrez, A., & Alonso, S. (2020). A Collaborative Filtering Probabilistic Approach for Recommendation to Large Homogeneous and Automatically Detected Groups. *International Journal of Interactive Multimedia and Artificial Intelligence, 6*(2).

Hurtado, R., Bobadilla, J., Gutiérrez, A., & Alonso, S. (2020). A Collaborative Filtering Probabilistic Approach for Recommendation to Large Homogeneous and Automatically Detected Groups. *IJIMAI journal, 6*(2), 90-100.

Inga Chalco, C., Bojorque Chasi, R., & Hurtado Ortiz, R. (n.d.). Hierarchical clustering for collaborative filtering recommender systems. *Advances in Intelligent Systems and Computing, 787*, 346-356.

IONIC FRAMEWORK. (2020, 04 02). Retrieved 10 12, 2020 from https://ionicframework.com/docs#:~:text=One%20codebase%2C%20running%20everywhere,at%20home%20on%20every%20device.

Jesús, M. M. (2016). *Un enfoque para Sistemas de Recomendación de paquetes turísticos basado en restricciones de usuario.* Buenos Aires.

Kyoungsoo , B., Inbae , J., Jongtae , L., & Jaesoo , Y. (2019). Expert Finding Considering Dynamic Profiles and Trust in Social Networks. *Electronics, 8*.

Leiva Olivencia, J. L., & Vargas Pérez, P. (2016). Sistema De Recomendación Turístico Grupal Basado en Personalidad. *Revista de Análisis Turistico*(21).

Moquillaza Henríquez, S. D., Vega Huerta, H., & Guerra Grados, L. (2010). Programación en N Capas. *Revista de Investigacion de Sistemas e Informática, 7*(2), 57-67.

OPEN JS FUNDATION . (n.d.). Retrieved 10 12, 2020 from https://nodejs.org/es/docs/

Oracle. (n.d.). *Oracle*. Retrieved 10 12, 2020 from https://docs.oracle.com/javaee/6/firstcup/doc/gkhoy.html

Ortega, F., Bobadilla, J., Guitiérrez, A., Hurtado , R., & Li, X. (2018). Artificial Intelligence Scientific Documentation Dataset for Recommender Systems. *IEEE Access, 6*.

Ortega, F., Hurtado, R., Bobadilla, J., & Bojorque , R. (2018). Recommendation to Groups of Users Using the Singularities Concept. *IEEE Access, 6*.

TORRES RUDLOFF, N. (2017). *CLUSTERING Y DIVERSIDAD EN SISTEMAS DE RECOMENDACIÓN TOP-N.* Santiago.

Zhu, B., Hurtado, R., Bobadilla, J., & Ortega, F. (2018). An Efficient Recommender System Method Based on the Numerical Relevances and the Non-Numerical Structures of the Ratings. *IEEE Access, 6*.