MOVIE RECOMMENDATION SYSTEM

Requirements and Specification Document

VERSION 4.0

DATE 07/03/2021

BY
Dasari Jayanth
2019BCS-016
Vemula Satya Pavan Kalyan
2019BCS-069

TABLE OF CONTENTS

Document Revision History

DATE	VERSION	DESCRIPTION	AUTHOR
28/02/2021	<1.0>	SRS 1.0	GROUP-13
03/03/2021	<2.0>	SRS 2.0	GROUP-13
05/03/2021	<3.0>	SRS 3.0	GROUP-13
07/03/2021	<4.0>	SRS 4.0	GROUP-13

1. Introduction

1.1 Purpose

The purpose of the Software Requirements Specification (SRS) document is to provide a detailed overview of our software product, its parameters, and goals. This document aims to gather and analyze and give an in-depth insight into the **Hybrid Movie Recommender system** by defining the problem statement in detail. It concentrates on the capabilities required by stakeholders and their needs while defining high-level product features. The straightforward user interface, hardware, and software requirements of the **Hybrid Movie Recommender system** are provided in this document.

1.2 Scope of project

This software helps users of the customer platform explore content quickly with our recommendation system's help. The software we are developing is a Hybrid Recommendation System for Movies, which uses the combination of collaborative and content-based filtering in the context of web-based recommender systems. In particular, we will link the well-known TMDB data set. The content filtering part of the system is based on trained neural networks representing individual user preferences. Using various experiments, we will demonstrate the influence of supplementary user and item features on our proposed hybrid recommender's prediction accuracy. To decrease system runtime and reveal latent user and item relations, we will factorize our hybrid model via singular value decomposition (SVD). Due to the enormous amount of information available online, the need for highly developed personalization and filtering systems is growing permanently. Recommendation systems constitute a specific type of information filtering that attempts to present items according to the interests expressed by a user.

1.3 Glossary

Term	Definition
Stakeholder	Any person with interest in the project which is not a developer.
Collaborative filtering	It tries to find the correlation between users and recommend content from other categories that they might like.
Content-based filtering	Recommendation based on information available about the item but not about the user.
Singular Value Decomposition(SVD)	It is the factorization of a real or complex matrix that generalizes the eigendecomposition of a square normal matrix to any matrix via an extension of the polar decomposition.

1.4 References

[1] Abdollahpouri, Himan & Burke, Robin. (2019). Multi-stakeholder Recommendation and its Connection to Multi-sided Fairness.

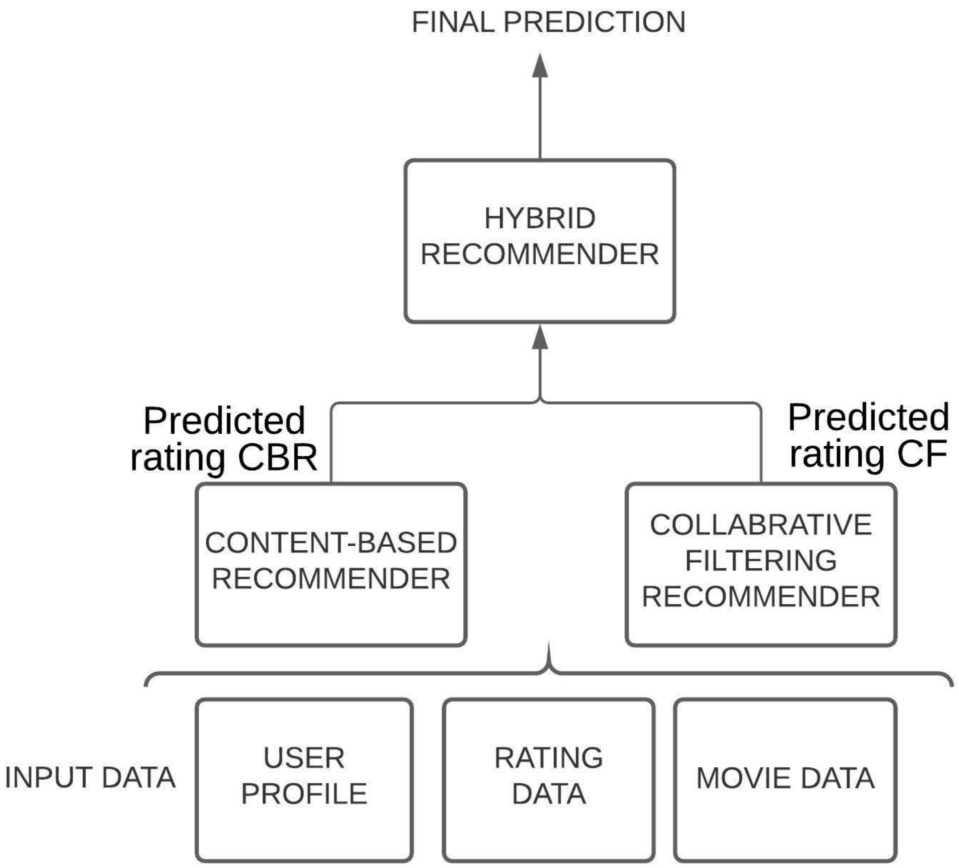
[2] Kunaver, Matevž & Pozrl, Tomaz & Pogacnik, Matevz & Tasic, Jurij. (2006). The evaluation of a hybrid recommender system for recommendation of movies.

[3]Rohan Nayak, Aniket Mirajkar, Jeetesh Rokade, Prof. Girish Wadhwa(2018). Hybrid Recommendation System For Movies

1.5 Overview

The remaining sections of this document provide a general description, including characteristics of this project’s users, the product's hardware, and the product’s functional and data requirements. A general description of the project is discussed in section 2 of this document. Section 3 gives the functional requirements, data requirements and constraints, and assumptions made while designing the E-Store. It also gives the user viewpoints of the product. Section 3 also gives the specific requirements of the product. Section 3 also discusses the external interface requirements and gives a detailed description of functional requirements. Section 4 is for supporting information.

2. Overall Description



Upon navigating to the main page, if an existing user, then we need prompt the user to log in. Upon logging in with the correct credentials, they will be redirected to searching movies data to the platform.Based on which new movies are recommended
In case the user is new, a registration prompt is provided first, then the user's profile is built and the user can search for movies he likes then system.
There are two types of users: normal users, who’s watch history and movie preferences are recorded, and admin users, which will be handled to the platform owners for the purpose of maintaining the platform.

The website should provide a search box, which enables the users to search movies. The admin user has the option to add and remove movies.

2.1 Customer

Our Customers are video-on-demand platform companies like Netflix, Amazon Prime, Hulu, and Hotstar. The business model that provides movies on demand to its customers enables searching for content on their website quickly and increases user satisfaction on the platform. Its success revolves around the potency of its recommendations. This System can expand to all video-on-demand platforms with the necessary data.

We conceptualize similarity in a broad sense, such as the similarity between movies, members, genres. It uses phrases such as 'Similar titles to watch instantly,' 'More like …'. Search is also one of the crucial aspects of any OTT Platform.

2.2 Competitive Landscape

There are many Recommender systems on the market. Still, all the systems recommend based on either a content-based recommendation or collaborative recommender system; we are trying to achieve better accuracy of recommendations for the platforms by implementing them and forming a new type of recommendation paradigm that is a hybrid of collaborative and content-based. This new approach differentiates our recommendation system from others. We cannot apply for patents, but the better our system works, and the more the users are using the system, the more collaborative filtering improves our system. Our user data is the key that makes our system perform better than any other recommender system.

Competitors such as Amazon, Hulu, Disney+, Sony, HBO, etc are also showing a major interest in such recommender systems. After all, they are the ones who produce movies. Why would they want intermediaries like AHA to take away the share? Many of them have started streaming their content by launching their own platforms.

2.3 Viewpoints of all the Stakeholders

The primary stakeholders of AHA are its subscribers and viewers. They are the ones who would be directly affected by the actions of this project. Our hybrid recommender system will help AHA company in boosting the subscriber numbers and the viewers.

The secondary stakeholders are its employees, with respect to the task, the secondary stakeholders are the research team of AHA who are directly involved with the development and maintenance of the algorithm and the system.

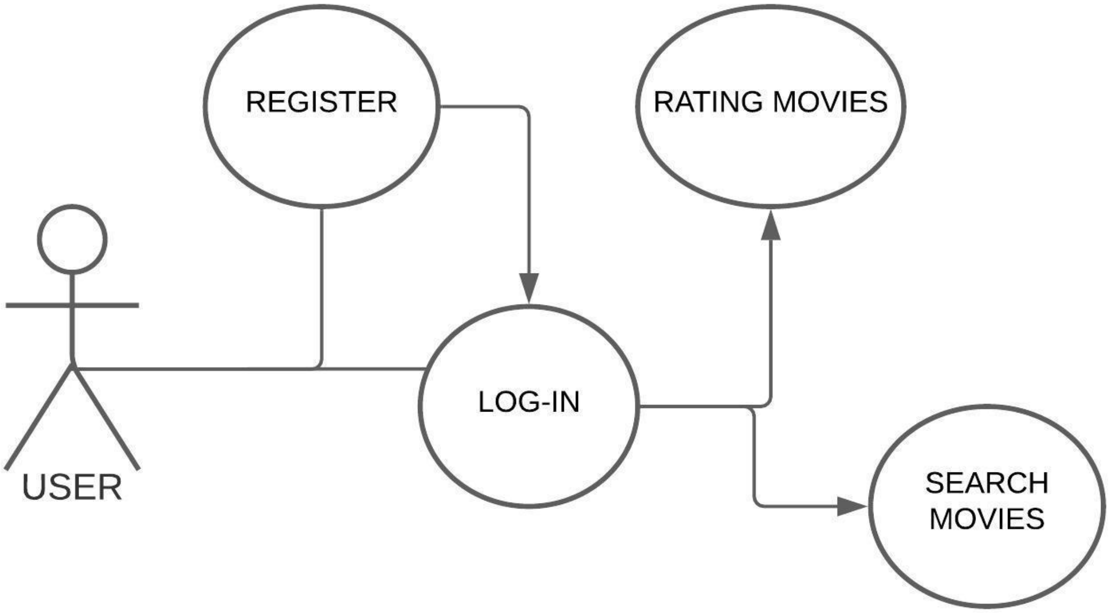
The third class of stakeholders is singular: the system owner. The owner profits directly when movies are watched. However, the owner is also concerned with the long-term health of the business. Studios can take their movies elsewhere and users can subscribe to other services that offer similar viewing experiences. These considerations should be part of the owner's utility function. A rational system owner seeks to maximize its own utility.

2.4 User Characteristics

The user is expected to be Internet literate and be able to use a search engine. The main screen of the Recommendation Website will have the search function and a link to add new content for the content providers.

The Content provider is expected to be Internet literate, have login credentials to navigate through the website, and be able to log in and have full details about the content to be added so that the recommendation system works optimally.

2.5 User Use Case Diagram



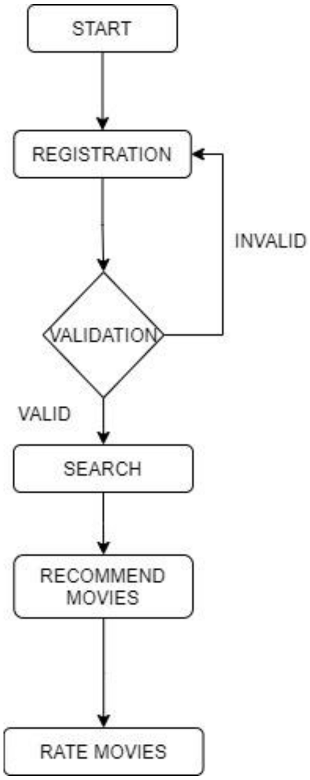
3. System Description

3.1 Functional Requirements Specification

This section outlines the functionality of each actor in this system.

3.1.1 Registration For New User

Activity Diagram For New User



REQUIREMENT-1:

If the User is new to the site, he first registers himself then searches for the movies he wants and then gives likes or dislikes, which helps create a profile for the user based on his likes/dislikes and the content of movies he likes; the system will recommend movies to the user.

REQUIREMENT-1.1:

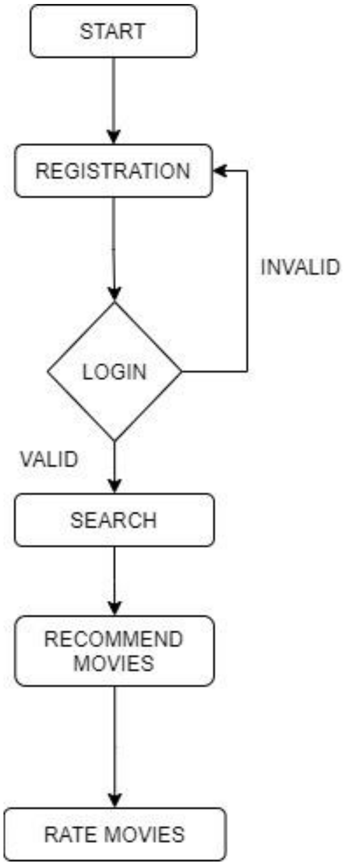
INPUT: “REGISTER” option selected by the user
OUTPUT: user prompted to enter personal data for registration

REQUIREMENT-1.2:

INPUT: the user enters registration data
OUTPUT: user redirected to the home page with the search engine
PROCESSING: check if the input values are in the correct format as needed

3.1.2 Recommendations For Existing User

Activity Diagram For Existing User



REQUIREMENT-2:

If the user is an existing one, he logs into the website.Based on what he searches in the search box and based on his likes/dislikes and the content of movies he likes we will recommend movies to the user.

REQUIREMENT-2.1:

INPUT: “LOG-IN” option selected by the user
OUTPUT: user prompted to enter USER NAME and PASSWORD

REQUIREMENT-2.2:

INPUT: enter USER NAME and PASSWORD
OUTPUT:
If the credentials are correct, then redirect to the home page with the search engine.
If the credentials are wrong, it shows a prompt of wrong credentials and asks to re-enter the credentials.

REQUIREMENT-3.1:

INPUT: “search” option,
OUTPUT: user prompted to enter “MOVIE” name

REQUIREMENT-3.2:

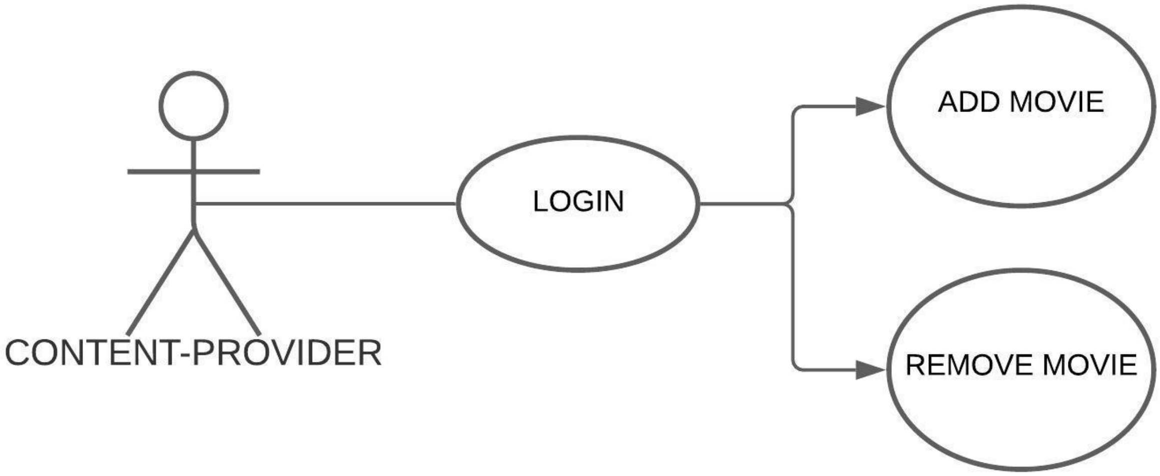
INPUT: “MOVIE NAME”
OUTPUT: Details about the movie if present in dataset and related recommendations of other movies.
PROCESSING: if the user is logged in, then based on collaborative and content filtering, new movies are recommended if not based only on content filtering, movies are recommended

REQUIREMENT-4:

INPUT: LIKE / DISLIKE MOVIE
OUTPUT: The movie rating is added to the user profile

3.1.3 Content Provider

Content Provider Diagram:

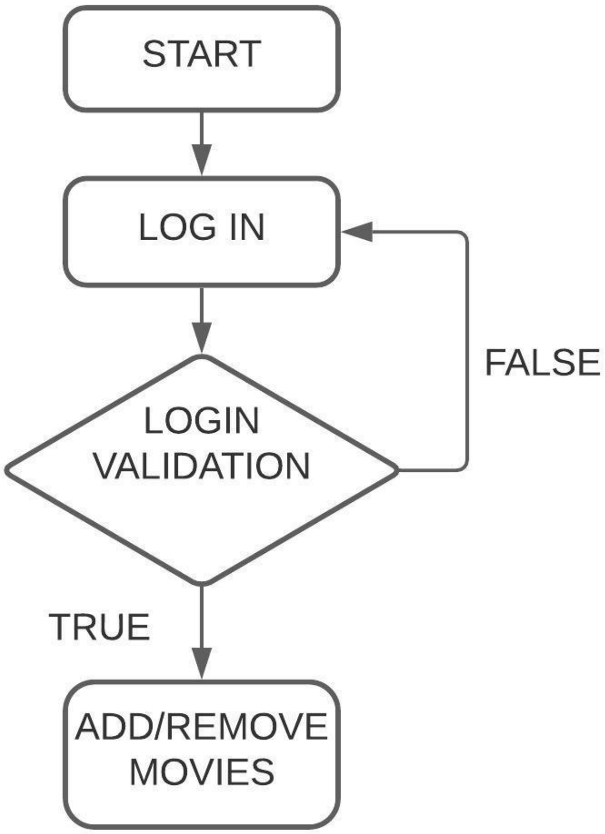


BRIEF-DESCRIPTION:
The content provider can add movies and their related information to the platform.

Initial Step-By-Step Description:
The content provider navigates to the main recommendation website and then

- navigates to the add content subpage
- upon logging in with the correct credentials
- HTML form appears when the data regarding the new movie to be added is entered, which is then added to the database.

Activity Diagram For Content-Provider



Requirement-5:

The content provider has already had the required credential using which he logs in to the site and then adds new movies data.

Requirement-5.1:

INPUT: "ADD/REMOVE MOVIES" option selected by the user.
OUTPUT: user prompted to enter USER NAME and PASSWORD

Requirement-5.2:

INPUT: enter USER NAME and PASSWORD
OUTPUT: If the credentials are correct, then redirect to ADD/REMOVE page with search engine
-If not, it sends a warning of wrong credentials and asks for login again

Requirement-5.3:

INPUT: If Add movie is selected then Enter MOVIE DATA
OUTPUT: If the DATA is in the correct format, it is added to the database

Requirement-5.4:

INPUT: If Remove data is selected then search movie option selected.
OUTPUT: Search box is prompted
.

Requirement-5.4.1:

INPUT: Proper keyword is entered to select the movie.
OUTPUT: If the movie is present in the data set then delete the movie from the dataset.If movie is not present in the dataset show it to the content provider.
PROCESSING: Based on the keyword of movie entered the movie is removed from the database

3.2 Non-Functional Requirements

The online Recommendation system will be hosted on Heroku, which provides free servers for hoisting our sites.

The users and content-providers need to have access to the necessary bandwidth internet to access the website.

PROCESSES AND TECHNOLOGY :

The recommender system uses supervised approaches such as classification and regression and unsupervised approaches such as dimensionality reduction and clustering/compression using topic modeling. Matrix factorization, Singular Value Decomposition, factorization machines, connections to probabilistic graphical models, and methods that can be easily expanded to be tailored for different problems.

3.2.1 Performance

The product shall be based on the web and has to be run from a web server. The product shall take initial load time depending on internet connection strength which also depends on the media from which the product is run. The performance shall depend upon the hardware components of the client/customer.If the Internet connection is good then The application should update the interface on interaction within 2-4 seconds.The database should be normalized to prevent redundant data and improve performance.

3.2.2 Security

A hosting service provider secures the server on which the website is hosted.

Data Transfer: The system will use secure sockets in all transactions that include any confidential customer information. The system will automatically log out all customers after a period of inactivity. The system shall confirm all transactions with the customer’s web browser. The system will not leave any cookies on the customer’s computer containing the user’s password or any of the user’s confidential information

Data Storage: The customer’s web browser and the system’s back-end servers will never display a customer’s password. The system’s back-end databases will be encrypted.

3.2.3 Safety

Databases should be redundant to prevent loss of data.Backups of the databases should be done hourly.

3.2.4 Software Quality Attributes

Availability: All the movies may not be present,we should contain as many movies as possible based on priority and ratings etc.If a user searches for a movie it should be available for the user.

Correctness: If a user searches for a movie with a keyword like genres,movie name etc,the correct and related movies should be recommended in case the movie searched is not present.

Maintainability: The application should use continuous integration so that features and bug fixes can be deployed quickly without downtime.

Usability: The interface should be easy to learn without a tutorial and allow users to accomplish their goals without errors.

3.3 External Interface Requirements

The protocol used will be HTTP. There shall be a logical address of the system in IPv4 format.

3.3.1 User Interfaces

The user interface for the software shall be compatible with any browser such as Internet Explorer, Mozilla, and Google Chrome by which users can access the system.

3.3.2 Hardware Interfaces

Since the application must run over the internet, all the hardware required to be connected to the internet will be a hardware interface for the system. As for e.g. Modem, WAN – LAN, Ethernet Cross-Cable.

3.3.3 Software Interfaces

We have chosen Windows operating system for its best support and user-friendliness.To save the movie details, users preferences etc, we have chosen SQL database.To implement the project we will use Python libraries,Javascript,Html etc.

3.4 Risks and Mitigations

RISKS:

1. Too few users might use the recommendations engine, increasing data scarcity and reducing as well the return on investment.
2. The Recommender system might recommend bad items if the items pool is not controlled.
3. The Recommender system might raise data privacy concerns on the users' sides.

MITIGATIONS:

For Risk 1: The existence of the recommender engine must be known by all the users. Additionally, users must be prompted explicitly to provide their opinion on the recommendations.

For Risk 2: We suggest that item pools be pre-filtered in order to keep only desirable items. For instance, news articles must undergo editors' pre-filtering phase before they are included in the news articles pool. This would prevent the recommender engine from recommending them.

For Risk 3: As far as data privacy is concerned, measures must be taken to ensure that connections of users do not appear in the recommendations stage.

3.5 Evaluation metric

For evaluating recommendation engines, we can use the following metrics.

Precision:

$$\text{Precision} = \frac{tp}{tp + fp}$$

Where,
tp = movies that user like
tp+fp = Total movies recommended to the user
Evaluation metrics for recommendation engines

Recall:

What proportion of items are recommended that a user likes.
It is given by:
Here tp represents the number of items recommended to a user that he/she likes, and tp+fn represents the total items that a user likes.

$$Recall\ = \frac{True\ Positive(TP)}{True\ Positive(TP) + False\ Negative(FN)}$$

The larger the recall, the better are the recommendations

RMSE (Root Mean Squared Error):

It measures the error in the predicted ratings:
Here, Predicted is the rating predicted by the model, and Actual is the original rating.
Lesser the RMSE value, the better the recommendations