Software Requirements Specification

for

Recommended System for Movie Web Application

Version 1.0 approved

Prepared by Muhammad Ali Sajjad

Collabera Trainee

29 January 2023

Table of Contents

Revision History

Name	Date	Reason For Changes	Version

1. Introduction

1.1 Purpose

A movie recommendation system for a web application suggests relevant and personalized movie choices to users based on their past viewing history, preferences and other factors. The goal is to improve the user experience by providing users with recommendations that are tailored to their individual tastes, and user movie ratings, thus making it easier for them to discover new and exciting movies they might enjoy. The users are able to filter the movie display by clicking on selected genre. Each movie will display details, short description about the movie, ratings, movie trailer which on click and the recommended movies which are similar to the selected genre .

1.2 Intended Audience and Reading Suggestions

The intended audience for a movie recommendation system is users of a movie streaming web application. They are likely to be users and developers who regularly watch movies online and are looking for a convenient and effective way to discover new films that match their interests and preferences.

1.3 Project Scope

The scope of the movie recommendation system project includes:

- Gathering and processing large amounts of data about users and their movie preferences, including explicit feedback (e.g. ratings) and implicit feedback (e.g. watch history).

- Implementing various algorithms and models to analyze the data and generate personalized recommendations.

- Integrating the recommendation system into the movie streaming web application, so that users can receive suggestions seamlessly as part of their normal movie browsing experience.

- Regularly testing and evaluating the performance of the recommendation system, and updating it as necessary to ensure that it remains effective and accurate over time.

- Providing an intuitive and user-friendly interface that allows users to view and interact with their recommendations in a simple and effective way.

It is important to note that the scope of the project will depend on the specific requirements and constraints of the movie streaming web application and its users.

1.4   References

References used for this system:

- Collaborative filtering: https://en.wikipedia.org/wiki/Collaborative_filtering .
- Pearson correlation coefficient: https://en.wikipedia.org/wiki/Pearson_correlation_coefficient .
- Jaccard index: https://en.wikipedia.org/wiki/Jaccard_index
- Cosine similarity: https://analyticsindiamag.com/cosine-similarity-in-machine-learning/

2.   Overall Description

2.1   Product Perspective

The product vision for the movie recommendation system is to provide users of a movie streaming web application with a personalized and engaging movie-discovery experience. The product should aim to

- Offer users relevant and highly personalized movie recommendations that match their unique tastes and preferences.

- Enhance the overall user experience of the movie streaming web application, making it easier and more enjoyable for users to find and watch new movies.

- Provide seamless integration with the existing web application, so that users can receive and interact with their recommendations as part of their normal movie browsing experience.

- Regularly improve the accuracy and effectiveness of the recommendations over time, by incorporating feedback from users and updating the algorithms as necessary.

The product goals should align with the overall goals of the movie-streaming web application and should aim to improve the value that the web application provides to its users. Additionally, the product should be user-centred and designed to meet the needs and expectations of the target audience.

2.2   Product Features

The following are some of the key features that the movie recommendation system for a movie streaming web application could include:

1. Personalized Recommendations: The system should be able to analyze a user's viewing history, preferences, and other factors to generate highly personalized movie recommendations that match their individual tastes.

2. Relevance Feedback: Users should be able to provide explicit feedback (e.g. "like" or "dislike") on the recommendations they receive, allowing the system to continuously improve the relevance and accuracy of the recommendations over time.

3. Integration with Web Application: The recommendation system should be seamlessly integrated into the movie streaming web application so that users can receive and interact with their recommendations as part of their normal movie browsing experience.

4. User-Friendly Interface: The recommendation system should have a clean and intuitive user interface that makes it easy for users to view and interact with their recommendations.

5. Advanced Filtering Options: The system should allow users to filter their recommendations based on additional factors such as genre, year of release, cast, and more.

6. Social Sharing: Users should be able to share their recommendations with friends and family via social media and other platforms.

7. Real-Time Updates: The system should be able to update the recommendations in real time as the user interacts with the web application, ensuring that the recommendations are always up-to-date.

8. Performance Analytics: The system should provide analytics and metrics on the performance of the recommendations, allowing the movie streaming web application to continuously improve the accuracy and effectiveness of the system over time.

It is important to note that the specific features of the movie recommendation system will depend on the requirements and constraints of the movie streaming web application and its users.

2.3 User Classes and Characteristics

For this system has two user roles:

- Normal Users: These are the users who use the movie recommendation system to discover new movies. They can include movie lovers and casual watchers, who value highly personalized and relevant recommendations and appreciate a fast, efficient, and user-friendly system.

- Administrators: These are the users who manage and maintain the movie recommendation system. They are likely to have the technical expertise and be responsible for updating the algorithms, monitoring performance metrics, and ensuring the system is functioning correctly. They will need a user-friendly interface for managing and monitoring the system, as well as access to performance metrics and analytics to assess its effectiveness.

It is important to note that the specific user characteristics and requirements for administrators will depend on the technical sophistication and resources of the movie streaming web application, as well as the overall goals and objectives of the recommendation system. However, both normal users and administrators should value a recommendation system that is accurate, relevant, and easy to use.

2.4 Operating Environment

This system will be operating using the following software:

Software	Purpose
Visual Studio Code	To run React Js and the front-end of the system

Visual Studio 2022	To run ASP.NET Web Api and back-end of the system
Mssql Server Management	Create and store database for the system
TMDB web Api	To get the Movie details

The following are the key operating environment considerations for the movie recommendation system:

1. Technical Infrastructure: The system will require a robust and scalable technical infrastructure, capable of processing large amounts of data and delivering high-quality recommendations in real-time. This may include a combination of hardware and software components, such as servers, storage systems, and data processing platforms.

2. Network Connectivity: The system should be designed to work over a range of network connectivity conditions, from high-speed broadband to mobile networks. This will help ensure that users can receive recommendations regardless of their location or device.

3. User Devices: The recommendation system should be compatible with a range of user devices, including desktop computers, laptops, smartphones, and tablets. This will help ensure that users can receive and interact with recommendations regardless of their device or platform.

4. Security: The system should be designed with security in mind, to ensure that users' personal data and viewing history are protected. This may include measures such as encryption, authentication, and access control.

5. Data Privacy: The recommendation system should be designed to respect users' privacy, and should comply with relevant data protection regulations and standards. This may include measures such as obtaining user consent for collecting and processing personal data and providing users with control over their data.

6. User Interfaces: The system should include user-friendly interfaces for both normal users and administrators, allowing them to easily view and interact with recommendations and manage the system, respectively.

2.5   Design and Implementation Constraints

The following are the key design and implementation constraints for the movie recommendation system:

1. Scalability: The system should be scalable, to accommodate growth in user numbers and data volumes. This may include measures such as load balancing, data partitioning, and failover capabilities.

2. Performance: The system should deliver recommendations in real time, with a fast and responsive user experience. This will require careful design and optimization of algorithms, data structures, and system architecture.

3. Data Quality: The quality of recommendations will depend on the accuracy and completeness of the underlying data. The system should be designed to handle data inaccuracies, missing values, and outliers, and to automatically detect and correct any data quality issues.

4. Privacy: The system should respect users' privacy, and comply with relevant data protection regulations and standards. This may include measures such as encrypting personal data, obtaining user consent, and providing users with control over their data.

5. Flexibility: The system should be flexible, allowing administrators to update and modify algorithms and parameters, and to experiment with different recommendation techniques.

6. User Experience: The system should provide a seamless and intuitive user experience, with recommendations that are relevant, accurate, and tailored to individual users' preferences.

7. Integration: The system should integrate seamlessly with other systems and platforms, such as the movie streaming web application, user authentication and access control systems, and data analytics platforms.

2.6 User Documentation

The following is an overview of the user documentation for the movie recommendation system:

- Introduction: An introduction to the movie recommendation system, explaining its purpose and how it works.

- Getting Started: A step-by-step guide for users to start using the system, including how to log in, customize their recommendations, and view recommended movies.

- User Interface: A description of the user interface, including the different features and tools available, such as the recommendation list, movie search, and user preferences.

- Managing Recommendations: A guide to managing recommendations, including how to rate movies, provide feedback, and update user preferences.

- Privacy and Security: Information on the privacy and security measures in place, including how personal data is used and protected, and what rights users have over their data.

- Troubleshooting: A guide to troubleshooting common issues, such as issues with recommendations, login problems, and system performance.

- Glossary: A glossary of technical terms and concepts used in the movie recommendation system.

2.7   Assumptions and Dependencies

The following are some of the key assumptions and dependencies for the movie recommendation system:

1. User Data Availability: The system assumes that sufficient user data is available, including movie ratings, viewing history, and preferences, to train the recommendation algorithms and generate accurate recommendations.

2. Movie Data Availability: The system assumes that sufficient movie data is available, including movie metadata, genre information, and user ratings, to support the recommendation algorithms and provide relevant recommendations.

3. Integration with Movie Streaming Web Application: The system assumes that it is integrated with the movie streaming web application, and that it can access and process user and movie data from the application.

4. User Adoption: The system assumes that users will adopt and use the recommendation system, providing sufficient data to train the algorithms and generate accurate recommendations.

5. Technical Infrastructure: The system assumes that a robust technical infrastructure is in place, including a high-performance computing environment, storage and network infrastructure, and scalable data processing and analytics platforms.

6. Algorithm Performance: The system assumes that the recommendation algorithms perform well, generating accurate and relevant recommendations in real-time, and that they can be updated and modified as required.

7. User Privacy and Data Protection: The system assumes that user privacy and data protection regulations and standards are respected and complied with, and that users have control over their data.

3.   **System Features**

The movie recommendation system for a movie streaming web application may include the following features:

- Personalized Recommendations: The system provides personalized movie recommendations based on user preferences, viewing history, and ratings.

- Movie Search and Filtering: Users can search for movies based on various criteria, such as genre, release year, cast, and director, and filter the results based on their preferences.

- User Preferences: Users can set and update their preferences, such as preferred genres, actors, and directors, to influence the recommendations generated by the system.

- Movie Ratings: Users can rate movies, providing feedback to the recommendation algorithms and helping to improve the accuracy of recommendations.

- User Profiles: Users can create and manage personal profiles, which store their preferences, viewing history, and ratings, and provide a summary of their activity on the movie streaming web application.

- Recommendation History: Users can view their recommendation history, including the movies recommended to them and the reasons behind the recommendations.

- Real-Time Recommendations: The system generates recommendations in real-time, providing users with up-to-date and relevant suggestions for movies to watch.

- Scalable and Robust Infrastructure: The system is designed to scale and support high volumes of data and traffic, providing a fast and reliable recommendation experience for users.

- Data Privacy and Security: The system is designed to protect user data and privacy, ensuring that personal information is securely stored and processed in accordance with data protection regulations and standards.

4. External Interface Requirements

4.1 User Interfaces

The external interface requirements for the user interface of the movie recommendation system:

1. User-Friendly Design: The user interface should have an intuitive and user-friendly design, allowing users to easily find and use the recommendation system.

2. Responsive Design: The user interface should be responsive, adapting to different screen sizes and devices, and providing a consistent and seamless user experience.

3. Search and Filtering: The user interface should support search and filtering of movies, allowing users to quickly find and select movies based on their preferences.

4. User Profiles: The user interface should allow users to create and manage personal profiles, and to view and update their preferences, viewing history, and ratings.

5. Movie Ratings: The user interface should allow users to rate movies, providing feedback to the recommendation algorithms and helping to improve the accuracy of recommendations.

6. Recommendation History: The user interface should allow users to view their recommendation history, including the movies recommended to them and the reasons behind the recommendations.

7. Real-Time Recommendations: The user interface should display real-time recommendations, providing users with up-to-date and relevant suggestions for movies to watch.

8. Accessibility: The user interface should be accessible, providing support for users with disabilities, and meeting accessibility standards and guidelines.

9. Internationalization: The user interface should support internationalization, allowing the recommendation system to be used in different countries and regions, with localized language, currency, and time zones.

4.2 Hardware Interfaces

The Hardware interface requirements for the user interface of the movie recommendation system:

- Display: The user interface should support various display sizes and resolutions, including desktop computers, laptops, tablets, and smartphones.

- Input Devices: The user interface should support different types of input devices, such as mouse, touch screen, and keyboard, allowing users to interact with the recommendation system in a variety of ways.

- Network Connectivity: The user interface should require a stable and reliable internet connection, allowing users to access the recommendation system and receive real-time recommendations.

- Browser Compatibility: The user interface should be compatible with different web browsers, including Google Chrome, Mozilla Firefox, Apple Safari, and Microsoft Edge, to ensure that users can access the recommendation system using their preferred browser.

- Performance: The user interface should be fast and responsive, providing an optimal user experience and minimizing load times and delays.

- Security: The user interface should be secure, protecting user data and privacy, and ensuring that sensitive information is securely transmitted and stored.

4.3   Software Interfaces

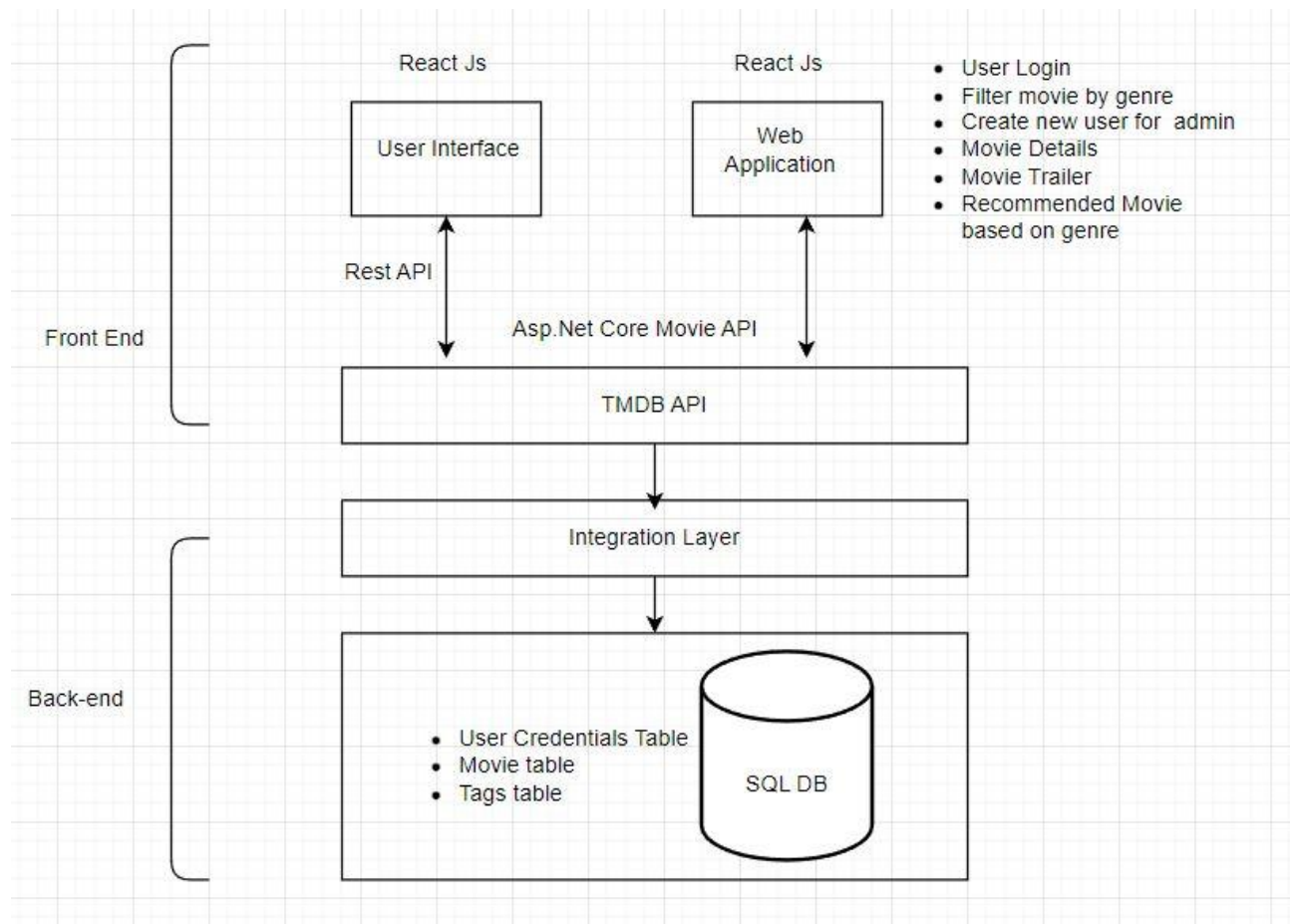The Sofwater iinterface requirements for the user interface of the movie recommendation system:

- User Management: The user interface should interface with the underlying user management system, allowing users to create and manage personal profiles, and to view and update their preferences, viewing history, and ratings.

- Recommendation Algorithms: The user interface should interface with the underlying recommendation algorithms, providing real-time recommendations based on user preferences and viewing history.

- Movie Database: The user interface should interface with the underlying movie database, allowing users to search and filter movies, view movie details, and rate movies.

- User Analytics: The user interface should interface with the underlying user analytics system, collecting and processing data about user behavior and preferences, and providing insights into the effectiveness of the recommendation system.

- Data Privacy: The user interface should comply with data privacy regulations and standards, protecting user data and privacy, and ensuring that sensitive information is securely transmitted and stored.

- Scalability: The user interface should be scalable, allowing the recommendation system to be deployed and used by a large number of users, without compromising performance or stability.

5.   Other Nonfunctional Requirements

5.1   Performance Requirements

- Response Time: The user interface should respond to user requests in a timely manner, providing real-time recommendations and updates, and minimizing waiting time for users.

- Load Performance: The user interface should be designed to handle high levels of traffic, allowing a large number of users to access the recommendation system simultaneously, without compromising performance or stability.

- Throughput: The user interface should support high levels of data throughput, allowing large amounts of data to be transmitted and processed in real-time.

- Scalability: The user interface should be scalable, allowing the recommendation system to be deployed and used by a large number of users, without compromising performance or stability.

- Resource Utilization: The user interface should utilize system resources efficiently, minimizing resource usage and maximizing performance.

5.2   High Level Diagram

5.3 Low Level Diagram