Project Requirement Document (PRD) Project Title:

Movie Recommendation System

Prepared By:

24EU08150-D. Datta

24EU08164-K. Kundan

1. Overview

1.1 Project Summary

The Movie Recommendation System is a machine learning–based application that suggests movies to users based on their ratings and preferences. The system uses Collaborative Filtering techniques to identify similar users or items and generate
personalized recommendations.

1.2 Background and Context

With the rapid growth of digital content, users often struggle to discover movies aligned with their
interests. This project enhances user experience by automatically recommending relevant movies using historical user interaction data.

1.3 Stakeholders

- End Users (Movie viewers)
- Product Owner
- ML Engineer
- Data Analys
-

1.4 Scope

In Scope:

- User ratings and preference analysis
- Personalized movie recommendations
- Recommendation result visualization

Of Scope:

- Movie streaming or playback
- Content licensing and distribution
- Real-time recommendation updates

---

2. Objectives and Goals

2.1 Key Objectives

- Recommend relevant movies to users
- Implement collaborative filtering algorithms
- Improve user engagement and satisfaction

2.2 Success Metrics

- Recommendation accuracy (Precision/Recall)
- User click-through or engagement rate
- Reduced recommendation error

---

3. Requirements

3.1 Functional Requirements

- User can provide movie ratings
- System stores and analyzes user preferences
- System recommends movies based on similar users/items
- Display Top-N movie recommendations

3.2 Non-Functional Requirements

- Scalable recommendation engine

- Secure handling of user data
- Fast recommendation response time

3.3 User Requirements

- Easy-to-use rating interface
- Clear list of recommended movies
- Minimal user input required

---

4. Use Case

4.1 Use Case Title

Recommend Movies

4.2 Actors

- **End User**

4.3 Description

The user rates movies or selects preferences. The system applies collaborative filtering to suggest movies the user is likely to enjoy.

4.4 Steps to Execute

1. **User opens the application**

2. **Rates movies or selects preferences**
3. **System processes user similarity**
4. **Recommended movies are displayed**

4.5 Time Required

- **3–8 seconds**

---

5. Technical Requirements

5.1 Technology Stack

- Frontend: Streamlit
- Backend: Python
- ML Technique: Collaborative Filtering (User- based / Item-based)
- Libraries: Scikit-learn, Pandas, NumPy
- Dataset: MovieLens

5.2 Integration Needs

- Movie metadata dataset
- Optional external movie API

---

6. Approvals and Sign-Offs

Role:

Name Signature:

Date Project Manager:

Client/Stakeholder:

Attachments / Appendices

- Algorithm explanation
- Evaluation metrics report

Googledrive Link:

https://drive.google.com/drive/folders/1QDczHDC5KkxYLMKVNUiVr_DxRd6qy5jK?usp=drive_link

Repository link:

https://github.com/repos

Google Colab Link:

https://colab.research.google.com/drive/1eArPw4yNHWpBEAs73vsVSD_-hIf_GxmS?usp=sharing