Software Requirements Specification

Project Title: Intelligent Movie Recommendation System with Power BI Analytics
Domain: Data Analytics / Business Intelligence / Machine Learning
Prepared By: [VAIBHAV NEGI , VINEET SAINI , UPENDRA KUMAR YADAV]
Date: [29-05-2025]

---

1. Introduction

1.1 Purpose

This document outlines the requirements for the development of a Movie Recommendation System integrated with analytical dashboards using Power BI. The system provides intelligent recommendations using machine learning and visual insights into movie trends and performance using real-world data.

1.2 Scope

- Recommend top 5 similar movies based on content using ML (cosine similarity)
- Provide users with an easy-to-use UI via Streamlit
- Fetch dynamic movie posters via TMDB API
- Provide visual reports like genre trends, popularity, rating distribution, revenue analysis, and clustering using Power BI
- Target users include casual viewers, OTT analysts, and business decision-makers

1.3 Definitions, Acronyms, Abbreviations

- **ML:** Machine Learning
- **NLP:** Natural Language Processing
- **UI:** User Interface
- **TMDB:** The Movie Database (API provider)
- **Power BI:** Microsoft Business Intelligence tool
- **KPI:** Key Performance Indicator

---

2. Overall Description

2.1 Product Perspective

This is a standalone system built using Python and Streamlit for ML-based recommendation and Power BI for data visualisation. It uses structured datasets from Kaggle and APIs from TMDB.

2.2 Product Functions

- User selects a movie → Top 5 similar movies are recommended
- Posters are displayed using TMDB API
- If no movie selected, system shows random popular movies
- Power BI Dashboards provide insights on:
    - Genre popularity
    - Blockbuster vs Flop analysis
    - Clustered movie groups
    - Year-wise trends

2.3 User Classes and Characteristics

- Casual movie viewers
- Business analysts (Power BI dashboards)
- Data science students and evaluators
- Teachers/mentors for academic evaluation

2.4 Operating Environment

- Python 3.10+
- Streamlit
- Jupyter Notebook
- Power BI Desktop
- Windows OS (preferred)
- Requires internet access for TMDB poster fetch

2.5 Design & Implementation Constraints

- Limited dataset
- No user authentication

- Offline ML model (no cloud-based training)

---

3. Functional Requirements

ID	Description
FR1	User inputs/selects a movie title
FR2	System fetches 5 similar movies based on cosine similarity
FR3	System fetches posters using TMDB API
FR4	If no movie selected, show 5 random popular movies
FR5	Show Power BI dashboard with filters (genre, year, cluster)
FR6	Display KPI cards and charts (revenue, votes, rating, popularity)

---

4. Non-Functional Requirements

ID	Description
NFR1	System should respond within 1 second for recommendation
NFR2	Dashboard visuals should update within 2 seconds
NFR3	Poster fallback image should display if API fails
NFR4	UI should be simple and user-friendly
NFR5	All recommendations should be relevant and explainable

---

5. External Interface Requirements

5.1 User Interface

- Dropdown menu for movie selection
- Poster grid display
- “Show Recommendation” button
- Dashboard interface in Power BI

5.2 Hardware Interfaces

- Desktop/laptop system with internet access

5.3 Software Interfaces

- Python (Streamlit, pandas, numpy, sklearn)
- TMDB API for posters
- Power BI for dashboards
- CSV files (movies metadata, ratings, credits)

---

6. Data Requirements

1. Dataset from Kaggle:

   tmdb_5000_credits , tmdb_5000_movies

2. Preprocessed movie_list.pkl and similarity.pkl using Pickle

   Columns used:

   title, overview, genres, keywords, cast, crew, vote_average, popularity, revenue, budget

---

7. Performance Requirements

- Recommendation time ≤ 1 sec
- Dashboard filter refresh ≤ 2 sec
- Load time of similarity matrix from Pickle ≤ 1 sec

---

8. Assumptions and Dependencies

- TMDB API is up and running
- Power BI dashboard is refreshed with clean CSV data
- No real-time user feedback or login required