

Phase-1 Submission Template

Student Name: Dhinoshni K

Register Number: 731123106014

Institution: Government College of Engineering, Erode.

Department: Electronics and Communication and Engineering.

Date of Submission: 02/05/2025

1. Problem Statement

In an era of digital content overload, users often face decision fatigue when choosing movies. Existing recommendation systems frequently provide generic or irrelevant suggestions, lacking personalization and contextual awareness. This project aims to develop an intelligent matchmaking system that delivers highly personalized movie recommendations by leveraging AI techniques such as collaborative filtering, content-based filtering, and mood analysis.

2. Objectives of the Project

- Develop a hybrid AI model combining collaborative and content-based filtering.
- Create dynamic user profiles based on viewing history, ratings, and preferences.
- Incorporate mood/contextual analysis using NLP.
- Build a scalable system capable of handling large datasets.
- Visualize recommendation accuracy and user satisfaction insights.

3. Scope of the Project

This project focuses on developing an AI-powered movie recommendation

system that provides personalized suggestions to users based on their preferences, viewing history, and emotional context. By combining collaborative filtering, content-based filtering, and sentiment analysis, the system delivers accurate and diverse recommendations. It features a user-friendly web interface and is designed for real-world deployment, offering scalability and explainability for improved user trust and engagement

4.Data Sources

- **User Data:** Ratings, watch history, user ID, timestamps.
- **Movie Data:** Title, genres, director, actors, release date, plot summary.
- **Auxiliary Data:** Mood tags (e.g., from social sentiment), user demographics (optional)

5.High-Level Methodology

- Handle missing data (e.g., incomplete ratings).
- Normalize rating values.
- Encode categorical variables (e.g., genres, actors).
- Tokenize and clean movie plots for NLP tasks.
- Merge user and movie datasets on common key

6.High-Level Methodology

- Rating distribution per user/movie.
- Popular genres and their average ratings.
- User engagement trends over time.
- Word clouds from movie plots by genre.
- Mood-based movie grouping using sentiment scores
- **Data Collection** – For this project, we will utilize the **MovieLens dataset**, which contains user ratings, movie metadata (titles, genres, release dates), and unique user identifiers. Additionally, we will **scrape or access APIs like IMDb and TMDb** to collect auxiliary data such as

movie plot summaries, user reviews, cast details, and director names. These will enrich our dataset for more personalized and content-aware recommendations.

Data Cleaning – The raw data often contains issues such as:

- **Missing values** in ratings or metadata fields
- **Duplicate records** for movies or users
- **Inconsistent formatting** (e.g., genre lists, release year formats)

Exploratory Data Analysis (EDA) – EDA will help us understand patterns and user behavior:

- **Distribution plots** for user ratings across genres
 - **Heatmaps** for user-item interaction
 - **Word clouds** and **bar charts** from plot summaries and genres
 - **Sentiment and mood analysis** to detect emotional tone in movie descriptions and reviews
- These insights will shape our feature selection and model development.

Feature Engineering –

To enhance our model:

- **TF-IDF vectorization** of plot summaries for content-based similarity
 - **One-hot encoding** of genres, actors, and directors
 - **User profiling** using average ratings per genre, watch time, and mood preferences
 - **Sentiment scores** derived from user reviews and plot descriptions
- These features help match users with movies not just based on ratings but emotional resonance and preferences.
- **Model Building** –
 - **Collaborative Filtering**: Matrix Factorization using SVD.
 - **Content-Based Filtering**: Cosine similarity on movie metadata and NLP features.
 - **Hybrid Model**: Weighted sum of collaborative and content scores.
 - **Context-Aware Extension**: Add mood/emotion vectors to recommendation inputs.

- **Visualization & Interpretation –**

Results and model performance will be visualized using:

- **Confusion matrices and ROC curves** (for rating predictions).
- **Bar and line charts** for genre preference and user trends.
- **Interactive dashboards** (using Streamlit or Plotly Dash) showing:
 - Personalized recommendations
 - Similar movies with explanations ("because you liked...")
 - Feature importance and model insights

Deployment – The system will be deployed as a **web-based application** using

Flask or Streamlit, where users can:

- Create a profile
- Rate movies
- View personalized recommendations
- Provide feedback to continuously improve the model

The backend will be hosted on **cloud platforms** like **Heroku** or **AWS**, and data will be stored in **SQLite or PostgreSQL** for efficient access.

7. Tools and Technologies

- **Languages:** Python
- **Libraries:** Pandas, NumPy, Scikit-learn, TensorFlow/PyTorch, Matplotlib, Seaborn, NLTK, SpaCy
- **Frameworks:** Flask/Streamlit for web app
- **Database:** SQLite or PostgreSQL
- **Deployment:** AWS/GCP for model hosting
- **Version Control:** Git & GitHub

8. Team Members and Roles

Name	Contribution
Dhinothni K	Data preprocessing, exploratory data analysis (EDA), and feature engineering
Dharshini A	Natural Language Processing (NLP) for mood analysis and content-based filtering
Krithik Roshan K M	Model building (collaborative & hybrid recommendation systems) and evaluation
Niteesh Raj B K	UI/UX design, dashboard visualization, and system integration & deployment

