**Phase-2 Submission Template**

**Student Name:** LOKESHWARI G

**Register Number:** 421323104028

**Institution:** KRISHNASAMY COLLEGE OF ENGINEERING AND TECHNOLOGY

**Department:** COMPUTER SCIENCE AND ENGINEERING

**Date of Submission:** 28-04-2025

**Github Repository Link:**

### **1. Problem Statement**

“Design an AI-driven matchmaking system that provides personalized movie recommendations to users. The project addresses real-world challenges such as information overload in movie platforms, cold start for new users, and the need for engaging content discovery. This is a recommendation problem, typically involving collaborative filtering, content-based filtering, or hybrid modeling techniques. Solving this improves user satisfaction, engagement, and retention on digital streaming platforms”.

### **2. Project Objectives**

Develop a recommendation system using machine learning to predict and suggest movies tailored to individual user preferences.

Improve the precision of movie suggestions from a baseline (e.g., 60%) to over 90%.

Provide a personalized user interface, increasing platform engagement.

Evolve objectives based on data insights such as viewing patterns, genre affinities, and sparsity handling.

### **3. Flowchart of the Project Workflow**

### IMG_256

### **4. Data Description**

### Sources: IMDB, TMDb APIs, and streaming platform public datasets.

### Type: Structured (user ratings, metadata), text (reviews), and categorical (genres).

### Size: Estimated thousands to millions of rows (user ratings, movies).

### Nature: Dynamic, as user behavior and movie catalog change.

### Target: Rating or implicit watch behavior (for supervised recommendation tasks).

### **5. Data Preprocessing**

### Missing Values: Imputation for user attributes and movie metadata.

### Duplicates: Checked and removed based on unique user-movie interactions.

### Outliers: Treated in ratings and viewing frequency.

### Data Consistency: Unified formats for date, genres, etc.

### Encoding: One-hot encoding for genres, label encoding for categorical inputs.

### Scaling: Applied standardization on numeric features like user activity.

### **6. Exploratory Data Analysis (EDA)**

### Univariate: Distribution of ratings, genre frequency, user watch patterns.

### Bivariate: Heatmaps of genre preference by age/location.

### Insights: High activity among 18–35 age group.

### Action, sci-fi, and drama are common favorite genres

### Cold start users lean toward trending movies initially.

### **7. Feature Engineering**

### Created user profile vectors using past viewing genres.

### Extracted movie similarity scres from metadata.

### Generated user-movie interaction matrices.

### Combined collaborative and content similarity scores.

### Optional dimensionality reduction applied (e.g., PCA for embeddings).

### **8. Model Building**

### Implemented and compared:

### Collaborative Filtering using matrix factorization.

### Content-Based Filtering using metadata similarity.

### Hybrid Model (best-performing).

### Evaluation Metrics: Precision@k, Recall@k, F1-Score, NDCG.

### 

### **9. Visualization of Results & Model Insights**

### Plots:

### Confusion matrix for binary relevance.

### Precision-Recall curves.

### Top features: genre match, director similarity, user rating trends.

### Insights:

### Hybrid model showed 15–20% better engagement prediction.

### High-performing features: viewing history and genre preference.

### **10. Tools and Technologies Used**

### Language: Python

### IDE: Google Colab, VS Code

### Libraries: pandas, numpy, scikit-learn, TensorFlow Recommenders, Surprise

### Visualization: seaborn, matplotlib, Plotly

### APIs: IMDB, TMDb

### **11. Team Members and Contributions**

MAHAPRIYADHARSHINI C – Project Manager, Report and Coordination

KEERTHANA S – Model Design and Training

LOGESHWARI G – Data Collection (User Data)

MAHASRI P – Data Collection (Movie Metadata)

MANISHA M – Platform Development and Deployment