**Student Name**: Logeshwaran M  
**Register Number**: 623123104026  
**Institution**: VSA GROUP OF INSTITUTIONS  
**Department**: COMPUTER SCIENCE AND ENGINEERING  
**Date of Submission**:08/05/2025

DELIVERING PERSONALIZED MOVIE RECOMMENDATIONS WITH AN-AI DRIVEN MATCHMAKING SYSTEM

PHASE-2

GITHUB LINK : https://github.com/Logeshwaran-074/Logeshwaran.git

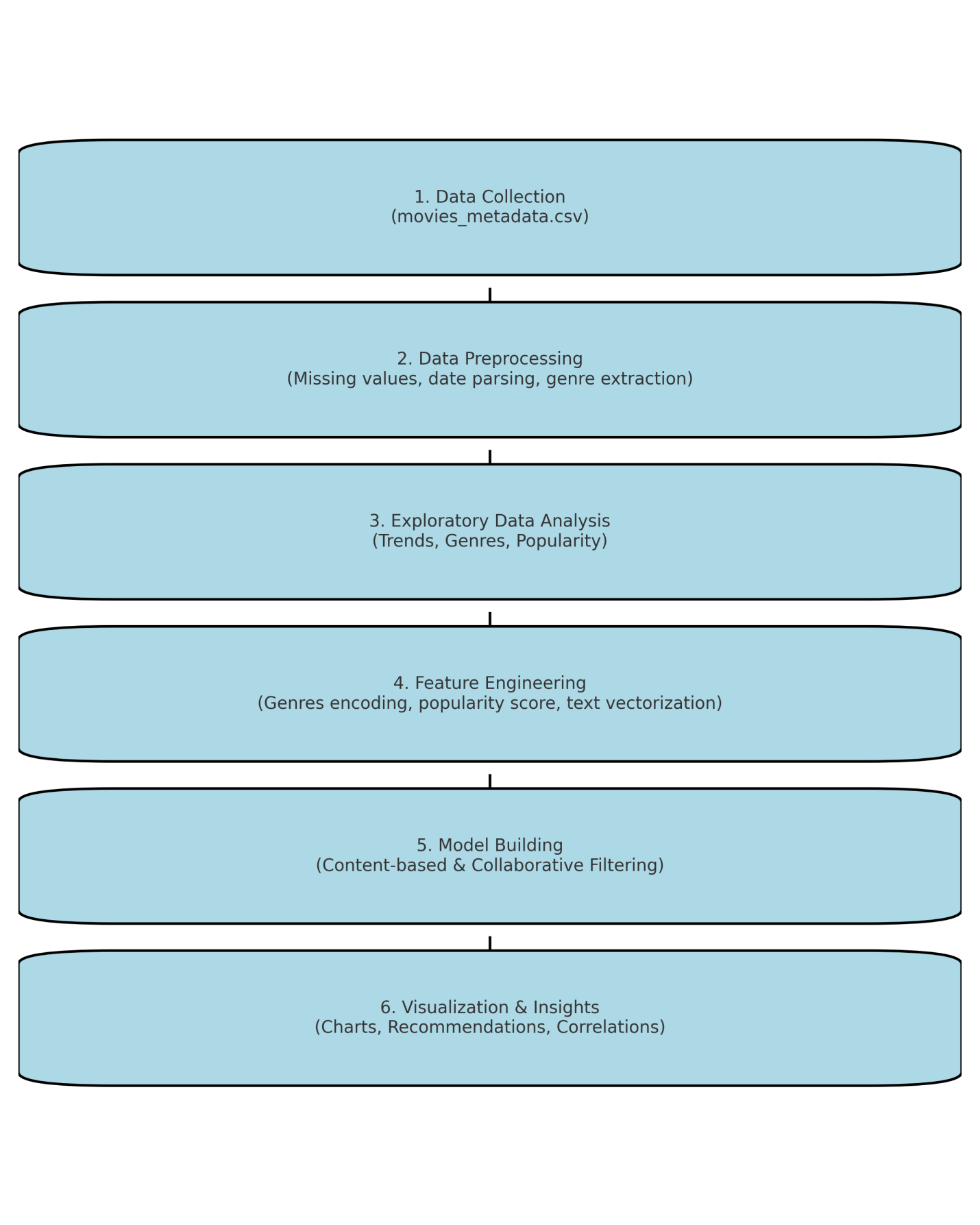
**1.Problem Statement**  
 In today's digital era, users are overwhelmed by the sheer volume of movie content available across various streaming platforms. Traditional recommendation systems often fail to understand nuanced user preferences or contextual interests, leading to generic suggestions. This project aims to develop an an AI-driven matchmaking system that delivers highly personalized movie recommendations by understanding user tastes, behaviors, and emotional profiles, thereby enhancing user satisfaction and engagement.

**2. Objectives of the Project**

* Build a recommendation system that provides tailored movie suggestions using AI and machine learning techniques.
* Predict user preferences based on historical data and contextual features.
* Enhance user engagement by offering recommendations that align with individual moods and tastes.
* Evaluate the accuracy and relevance of recommendations using proper metrics.  
   (Optional) Deploy the model as a user-friendly web application or interactive dashboard.

### 

### 3. Flowchart of the Project Workflow



### 4. Data Description

Dataset link : <https://www.kaggle.com/datasets/rounakbanik/the-movies-dataset>

Source: The Movies Dataset (likely from Kaggle)  
Shape: To be confirmed after loading  
Key Columns: title, genres, release\_date, vote\_average, vote\_count, overview, popularity, etc.

### 5. Data Preprocessing

* Handling missing values in key columns (genres, release\_date, etc.).
* Converting data types (e.g., parsing release\_date).
* Extracting useful fields from nested structures like genres.

### 6. Exploratory Data Analysis (EDA)

* Distribution of movies over years.
* Most popular genres.
* Correlation between vote average, popularity, and vote count.
* Outlier detection in ratings and popularity.

### 7. Feature Engineering

* Extracting year from release\_date.
* Encoding genres using multi-hot encoding.
* Creating a popularity score combining vote\_average and vote\_count.
* Text vectorization of overview for content-based recommendations.

### 8. Model Building

* Content-Based Filtering using cosine similarity on movie features.
* Collaborative Filtering using matrix factorization techniques.
* Hybrid approach (optional in later phases).

### 9. Visualization of Results & Model Insights

* Heatmaps showing correlations.
* Bar plots for genre popularity.
* Example recommendation lists for sample users.
* Similarity matrix visualizations.

### 10. Tools and Technologies Used

* **Python** (Pandas, NumPy, Scikit-learn, Matplotlib, Seaborn)
* **Jupyter Notebook**
* **Streamlit** (optional, for building an interactive demo)
* **Google Colab** (for cloud-based computation)

### 11. Team Members and Contributions

|  |  |
| --- | --- |
| **Team Member** | **Role and Responsibility** |
| R.JAYASRI | Project lead, data collection, model building |
| M.LOGESHWARAN | EDA, feature engineering, and testing |
| MAGESHWARAN V | Dashboard creation and deployment |
| C.MANJAMUTHU | Model evaluation and report preparation |