📊 Project Report: Movie Recommendation System

🔍 1. Project Title

Movie Recommendation System using Content-Based Filtering in Streamlit

---

🎯 2. Objective

The main objective of this project is to build an interactive web-based application that can recommend movies to users based on the genre similarity of selected movies using content-based filtering. The app enables users to upload their custom movie datasets and receive recommendations in real-time.

🧠 3. Project Description

In today's digital age, personalized recommendations are essential in enhancing user engagement on entertainment platforms. This project leverages **Natural Language Processing (NLP)** and **Machine Learning (ML)** techniques to provide relevant movie suggestions based on the genres of movies users like.

The recommendation logic is powered by **cosine similarity** on genre vectors, making it a content-based system that doesn't require user ratings.

🛠 4. Tools and Technologies Used

Tool/Technology	Purpose
Python	Core programming language
Streamlit	Web application interface
Pandas	Data manipulation and handling
Scikit-learn	Vectorization and similarity computation
CountVectorizer	Genre text vectorization
Cosine Similarity	Measure movie similarity

🗃 5. Dataset Description

- The dataset contains the following columns:

    - movieId: Unique identifier for each movie
    - title: Title of the movie (with release year)
    - genres: Genre tags separated by pipe (|)

A sample of 100 synthetic movie records was generated for testing and demonstration purposes.

🧮 6. Methodology

1. **Data Upload**: The user uploads a .csv or Excel file containing movie data.
2. **Preprocessing**: Genres are cleaned and tokenized using the CountVectorizer.
3. **Similarity Matrix**: A cosine similarity matrix is computed for genre vectors.
4. **Recommendation Logic**: The system retrieves top N similar movies based on genre similarity.
5. **Interface**: Users select a movie title and get real-time recommendations through the Streamlit interface.

🧑‍💻 7. Features

- Upload dataset dynamically (CSV/XLS/XLSX).
- Real-time recommendations based on genre similarity.
- Option to preview uploaded dataset.
- Adjustable number of recommendations.
- Interactive and user-friendly interface.

📈 8. Output Sample

Example recommendation output when the user selects "Toy Story (1995)":

markdown

CopyEdit

1. Jumanji (1995)

2. Balto (1995)

3. Tom and Huck (1995)

4. Cutthroat Island (1995)

5. GoldenEye (1995)

✅ 9. Conclusion

This project successfully demonstrates the use of **content-based filtering** for building a basic yet effective movie recommender system. By utilizing genre similarity, it helps users discover movies similar to their preferences. The integration with **Streamlit** enhances usability and provides a modern, interactive user experience.