

✓ Title: **Recomendation System - Movies, Music & Books**

Milestone 1 Report: Session-Based Recommendation System

The objective of this project is to build a **session-based recommendation system** that suggests movies, music, and books to users based on their interactions with the interface and preferences. This project will use 3 datasets from Hugging Face. There will be an interactive user-interface of website will be developed for this application.

Technology Stack

- **Frontend:** React, CSS
- **Backend:** Flask
- **Machine Learning Frameworks:** PyTorch, transformers
- **Databases:** PostgreSQL / MongoDB (TBD based on requirements)
- **APIs & Libraries:** Hugging Face Transformers, Pandas, NumPy, Matplotlib, Seaborn

Datasets(Hugging Face)

- **Movies:** [IMDb Dataset](#)
- **Music:** [Spotify Tracklist Dataset](#)
- **Books:** [GoodReads Dataset](#)

Project Timeline

Task	Start Date	End Date
Data Collection	Feb 10, 2025	Feb 15, 2025
Data Preprocessing	Feb 16, 2025	Feb 18, 2025
Exploratory Data Analysis (EDA)	Feb 18, 2025	Feb 20, 2025
Model Selection & Training	Feb 23, 2025	Feb 28, 2025
Model Evaluation & Tuning	Mar 01, 2025	Mar 10, 2025
Integration with Backend	Mar 12, 2025	Mar 20, 2025
Frontend Development	Mar 23, 2025	Mar 30, 2025
Deployment	Apr 02, 2025	Apr 12, 2025

✓ Data Loading

```
#!pip install datasets
```

```
from datasets import load_dataset
```

```
movies_data = load_dataset("ExecuteAutomation/ImdbMovieDataSet")
```

```
print(movies_data['train'])
```

```
Dataset({
  features: ['names', 'date_x', 'score', 'genre', 'overview', 'crew', 'orig_title', 'status', 'orig_lang', 'budget_',
  num_rows: 10178
})
```

```
music_data = load_dataset("maharshipandya/spotify-tracks-dataset")
```

```
Dataset({
  features: ['Unnamed: 0', 'track_id', 'artists', 'album_name', 'track_name', 'popularity', 'duration_ms', 'explici
  num_rows: 114000
})
```

README.md: 100% 4.68k/4.68k [00:00<00:00, 98.4kB/s]
dataset.csv: 100% 20.1M/20.1M [00:00<00:00, 37.3MB/s]
Generating train split: 100% 114000/114000 [00:00<00:00, 145310.78 examples/s]

```
print(music_data['train'])
```

```
Dataset({
  features: ['Unnamed: 0', 'track_id', 'artists', 'album_name', 'track_name', 'popularity', 'duration_ms', 'explici
  num_rows: 114000
})
```

```
books_data = load_dataset("Eitanli/goodreads")
```

```
Repo card metadata block was not found. Setting CardData to empty.
WARNING:huggingface_hub.repocard:Repo card metadata block was not found. Setting CardData to empty.
```

```
print(books_data["train"])
```


```
Dataset({
  features: ['Unnamed: 0', 'Book', 'Author', 'Description', 'Genres', 'Avg_Rating', 'Num_Ratings', 'URL'],
  num_rows: 10000
})
```

```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
```

```
# Data
```

```
data = {
  'Task': [
    'Data Collection', 'Data Preprocessing', 'Exploratory Data Analysis (EDA)',
    'Model Selection & Training', 'Model Evaluation & Tuning',
    'Integration with Backend', 'Frontend Development', 'Deployment'
  ],
  'Start Date': [
    'Feb 10, 2025', 'Feb 16, 2025', 'Feb 18, 2025',
    'Feb 23, 2025', 'Mar 01, 2025', 'Mar 12, 2025',
    'Mar 23, 2025', 'Apr 02, 2025'
  ],
  'End Date': [
```

```
'Feb 15, 2025', 'Feb 18, 2025', 'Feb 20, 2025',  
'Feb 28, 2025', 'Mar 10, 2025', 'Mar 20, 2025',  
'Mar 30, 2025', 'Apr 12, 2025'  
]  
}  
  
df = pd.DataFrame(data)  
  
# Plotting the table  
fig, ax = plt.subplots(figsize=(10, 6))  
ax.axis('off')  
table = ax.table(cellText=df.values, colLabels=df.columns, cellLoc='center', loc='center')  
  
# Save as image  
image_path = "/content/images/project_schedule.png"  
plt.savefig(image_path, bbox_inches="tight", dpi=300)  
plt.close()  
  
image_path
```

 `"/content/images/project_schedule.png"`