

**MOVIE RECOMMENDATION SYSTEM USING MACHINE LEARNING**

**BY**

**MD TANVIR MAHTAB**

**193-15-13433**

This Report Presented in Partial Fulfillment of the Requirements for the Degree of  
Bachelor of Science in Computer Science and Engineering

**Supervised By**

**Dr. Sheak Rashed Haider Noori**

Professor

Department of CSE

Daffodil International University

**Co-Supervised By**

**Md. Sazzadur Ahamed**

Assistant Professor

Department of CSE

Daffodil International University



**DAFFODIL INTERNATIONAL UNIVERSITY**

**DHAKA, BANGLADESH**

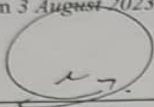
**July 2023**

## APPROVAL

### APPROVAL

This Project/internship titled "**MOVIE RECOMMENDATION SYSTEM USING MACHINE LEARNING**", submitted by **MD TANVIR MAHTAB**, ID No: **193-15-13433** to the Department of Computer Science and Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on *3 August 2023*.

#### BOARD OF EXAMINERS



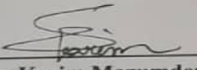
**Dr. S.M Aminul Haque (SMAH)**  
**Associate Professor and Associate Head**  
Department of Computer Science and Engineering  
Faculty of Science & Information Technology  
Daffodil International University

**Chairman**



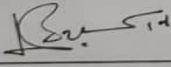
**Md. Abbas Ali Khan (AAK)**  
**Assistant Professor**  
Department of Computer Science and Engineering  
Faculty of Science & Information Technology  
Daffodil International University

**Internal Examiner**



**Fourcan Karim Mazumder (FKM)**  
**Assistant Professor**  
Department of Computer Science and Engineering  
Faculty of Science & Information Technology  
Daffodil International University

**Internal Examiner**



**Dr. Md. Sazzadur Rahman (MSR)**  
**Associate Professor**  
Institute of Information Technology  
Jahangirnagar University

**External Examiner**

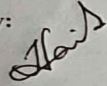
---

# DECLARATION

## DECLARATION

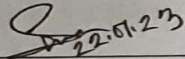
We hereby declare that, this project has been done by us under the supervision of **Dr. Sheak Rashed Haider Noori, Senior Lecturer, Department of CSE Daffodil International University.** We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:



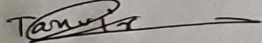
**Dr. Sheak Rashed Haider Noori**  
Professor  
Department of CSE  
Daffodil International University

Co-Supervised by:



**Ms. Taslima Ferdous Shuva**  
Assistant Professor  
Department of CSE  
Daffodil International University

Submitted by:



**MD. TANVIR MAHTAB**  
ID: 193-15-13433  
Department of CSE  
Daffodil International University

©Daffodil International University

Page | ii

## ACKNOWLEDGEMENT

First, we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes it possible for me to complete the final year project/internship successfully.

We really grateful and wish our profound our indebtedness to **Dr. Sheak Rashed Haider Noori** , Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “*Web development*” to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to **Dr. Touhid Bhuiyan**, Professor and Head, Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

## ABSTRACT

The movie industry has witnessed exponential growth in recent years, leading to a vast selection of movies across various genres and languages. However, the sheer abundance of options makes it challenging for movie enthusiasts to find films that align with their personal preferences. Movie recommendation systems have emerged as a solution to address this challenge by leveraging machine learning algorithms to provide personalized movie suggestions. In this project, we develop a movie recommendation system using machine learning techniques. The system analyzes user preferences, historical viewing patterns, and movie metadata to generate tailored recommendations. We explore different machine learning algorithms, including collaborative filtering and content-based filtering, to deliver accurate and relevant movie suggestions to users. The objectives of the project are to enhance the movie watching experience for users by simplifying the movie selection process and providing personalized recommendations. Additionally, we aim to provide valuable insights to the movie industry by analyzing user data, enabling movie studios and marketers to better understand audience preferences and make informed decisions about future projects.

To achieve these objectives, we collect and preprocess a dataset containing information such as release date, genre, popularity, and language. We implement and evaluate various machine learning models to determine their effectiveness in generating accurate recommendations. The performance of the system is evaluated using appropriate evaluation metrics, and the results are discussed. The findings of this project highlight the potential of machine learning in improving movie recommendations. The personalized nature of the system enhances the movie watching experience for users and helps them discover movies that align with their tastes. The insights gained from user data analysis can be leveraged by the movie industry to understand audience preferences and tailor their production and marketing strategies accordingly. Overall, this project demonstrates the value and efficacy of machine learning algorithms in

Developing a movie recommendation system. The implementation and evaluation of different techniques contribute to the growing body of knowledge in the field of recommendation systems and offer practical insights for enhancing user satisfaction in the movie domain.

# TABLE OF CONTENTS

<b>CONTENTS</b>	<b>PAGE</b>
Approval Page	i
Declaration	ii
Acknowledgements	iii
Abstract	iv
Table of Contents	v
List of Chapters	v - vii
List of Figures	viii - ix
<b>CHAPTER 1: INTRODUCTION</b>	<b>1-3</b>
1.1 Introduction	1
1.2 Motivation	2
1.3 Objective	2
1.4 Expected Outcomes	2
1.5 Project Management and Finance	3
<b>CHAPTER 2: BACKGROUND</b>	<b>4-7</b>
2.1 Preliminaries/Terminologies	4-5
2.1.1 HTML	4
2.1.2 CSS	4
2.1.3 JavaScript	4
2.1.4 Flask	4
2.1.5 Python	4
2.1.6 SQLite3	4

2.1.7 Google Chrome	4
2.1.8 Visual Studio Code	5
2.1.9 Adobe Illustrator	5
2.1.10 Github	5
2.1.11 Reddit	5
2.1.12 Stackoverflow	5
2.2 Related Works	6
2.2.1 Related Site Links	6
2.2.2 Related Research Works	6
2.3 Comparative Analysis	7
2.4 Challenges	7
<b>CHAPTER 3: REQUIREMENT SPECIFICATION</b>	<b>8-13</b>
3.1 Requirement Collection and Analysis	8-10
3.1.1 Requirement Collection	8-10
3.2 UML Class Diagram	11
3.3 ER Diagram	12
3.4 Design Requirement	13
<b>CHAPTER 4: DESIGN SPECIFICATION</b>	<b>14-27</b>
4.1 Front-end Design	14-26
4.1.1 Homepage	14-15
4.1.2 Search page	16
4.1.3 Recommendation page	17
4.1.4 About page	17
4.1.5 Mobile Responsive	18



4.2 Back-end Design	27
<b>CHAPTER 5: IMPLEMENTATION AND TESTING</b>	<b>28-31</b>
6.1 Implementation Details	32
6.2 Software Development:	32
6.3 Data Collection and Integration	33
6.4 User Interface (UI) Design:	33
<b>CHAPTER 6: IMPACT ON SOCIETY, ENVIRONMENT AND SUSTAINABILITY</b>	<b>32-33</b>
6.1 Impact on Society	32
6.2 Impact on Environment	32
6.3 Ethical Aspects	33
6.4 Sustainability Plan	33
<b>CHAPTER 7: CONCLUSION AND FUTURE SCOPE</b>	<b>34</b>
7.1 Discussion and Conclusion	34
7.2 Scope for Further Developments	34
<b>REFERENCES</b>	<b>35-36</b>
<b>PLAGIARISM REPORT</b>	<b>37</b>

## LIST OF FIGURES

<b>FIGURES</b>	<b>PAGE NO</b>
Figure 3.2 UML Diagram	11
Figure 3.3 ER Diagram	12
	14
	15
	15
	16
	16
	17
	17
	18
	18
	19
	19
	20
	20
	21
	22
	22
	23

Figure 4.1.10 List delete page	23
Figure 4.1.11.1 Available chatrooms	24
Figure 4.1.11.2 Chatroom Room	24
Figure	25
Figure 4.1.13 About us page	25
Figure 4.1.14 Mobile responsiveness	26
Figure	26
Figure 4.2 Backend Admin Panel	27
Figure	28
Figure	29
Figure 5.1.3 (List)Movie list model data	29
Figure 5.1.4.1 (Genre) Movie genre model data	30
Figure 5.1.4.2 (Review) Movie Review model data	30
Figure 5.1.4.3 (Starr) Movie Rating model data	31
Figure	31

# CHAPTER 1

## INTRODUCTION

### 1.1 Introduction:

A movie recommendation system using machine learning is a system that uses algorithms and data analysis techniques to suggest movies to users based on their preferences and viewing history. The goal of a movie recommendation system is to provide personalized recommendations that are tailored to the individual user's tastes and preferences. The movie recommendation system works by analyzing the data about users' movie preferences and viewing history, as well as information about the movies themselves, such as genre, director, and actors. This data is then used to build a model that predicts which movies a user is likely to enjoy. There are several machine learning algorithms that can be used for movie recommendation, including collaborative filtering, content-based filtering, and hybrid models. Collaborative filtering is based on the idea that people who have similar preferences in the past are likely to have similar preferences in the future. Content-based filtering uses the features of the movies themselves, such as genre and director, to make recommendations. Hybrid models combine the strengths of both collaborative and content-based filtering to provide more accurate recommendations. The movie recommendation system can be implemented as a standalone application or integrated into a larger platform, such as a streaming service or a social media site. In addition to providing movie recommendations, the system can also provide data and insights about users' movie preferences and habits, which can be used by movie studios and marketers to better understand their audience and make informed decisions about future projects. In conclusion, a movie recommendation system using machine learning can provide a personalized and convenient movie-watching experience for users, while also providing valuable data and insights to the movie industry.

## 1.2 Motivation

The motivation behind creating a movie recommendation system using machine learning is to address the problem of information overload and the difficulty in finding a good movie to watch. With so many options available, it can be overwhelming for users to choose a movie that they will enjoy. Moreover, traditional methods of movie recommendations, such as relying on friends' opinions or popular movie lists, may not always result in recommendations that are relevant to the user's individual preferences. By using machine learning algorithms and data analysis techniques, a movie recommendation system can provide personalized recommendations to users based on their viewing history and preferences. This improves the movie-watching experience for users, by making it easier for them to find something to watch and discover new movies that they might enjoy. In addition, the movie recommendation system can provide valuable data and insights to the movie industry. By analyzing data about users' movie preferences and habits, movie studios and marketers can better understand their audience and make informed decisions about future projects. Furthermore, the increasing popularity of streaming services has generated vast amounts of data about users' movie preferences, which can be leveraged to improve the accuracy and relevance of movie recommendations. The motivation behind using machine learning in movie recommendation systems is to take advantage of this data to provide users with a better movie-watching experience and to provide valuable insights into the movie industry. Overall, the motivation behind a movie recommendation system using machine learning is to improve the movie-watching experience for users, provide valuable data and insights to the movie industry, and address the problem of information overload and the difficulty in finding a good movie to watch.

There is another reason behind doing this is to check how the recommendation system works and can I make a new recommendation algorithm.

### 1.3 Objective:

In the scarcity of preferred suggestions among the scattered search results this site will provide the users with specific results which will save their valuable time and make searching stuffs easier and fun.

To make the site user friendly, easy, efficient and functional HTML, CSS, JavaScript, Python, Flask, SQLite, Visual studio code, Google chrome was used.

Other than that this are also my objective

- 1) **Personalized Recommendations:** The primary goal is to provide personalized and relevant movie recommendations to users based on their preferences and viewing history. The system aims to understand each user's individual tastes and preferences to suggest movies that align with their unique interests.
- 2) **Enhanced Movie Discovery:** The system aims to help users discover new movies that they are likely to enjoy but may not have otherwise considered. By leveraging machine learning algorithms and data analysis techniques, the system can identify hidden patterns and connections to recommend movies that users may find appealing.
- 3) **Simplified Movie Selection:** The movie recommendation system strives to simplify the process of finding something to watch by reducing information overload and choice paralysis. By presenting users with tailored recommendations, the system narrows down the vast selection of movies, making it easier for users to make a decision.
- 4) **Accurate and Personalized Recommendations:** Leveraging machine learning algorithms and data analysis techniques, the system analyzes large amounts of data about users' movie preferences and viewing history, as well as movie metadata, to generate accurate and personalized recommendations. The objective is to deliver suggestions that align with users' individual tastes and preferences.
- 5) **Improved Movie-Watching Experience:** The overall objective is to enhance the movie-watching experience for users. By providing relevant and personalized recommendations, the system aims to increase user satisfaction, engagement, and enjoyment in the movie selection process.

## 1.4 Expected Outcomes

Start

Input: User Preferences

### 1. User Preferences:

- Gather user preferences, including movie genres, actors, directors, release dates, and other relevant criteria.

### 2. Personalized Filtering:

- Filter each post or recommendation precisely and specifically to match the user's preferences.
- Utilize advanced algorithms and techniques to analyze user preferences and match them with relevant content.

### 3. User-Friendly Interface:

- Design a user-friendly interface for quick navigation and easy access to personalized recommendations.
- Ensure intuitive and responsive design elements to enhance the user experience.

### 4. Offensive Content Filtration:

- Implement a robust content moderation system to filter out offensive or abusive content.
- Admin panel functionality allows the monitoring and removal of inappropriate content.

### 5. Time Optimization:

- Save users' time by providing targeted recommendations based on their preferences, eliminating the need to search through irrelevant content.
- Algorithms continuously learn and adapt to improve recommendation accuracy and relevance.

End

## **1.5 Project Management and Finance**

This project was developed by a single member. The project went through several stages, starting with the sourcing of the project idea and finalizing the concept. Detailed planning and the creation of a feature list were then carried out. Subsequently, the required technologies for the project were selected.

The project was mapped out, taking into account the available development timeline. The feature list and mapping were used to create a UML diagram. The development process began with the implementation of the frontend and backend components. Initially, basic CRUD functionalities were implemented, followed by the integration of key and detailed features, UI design, and mobile responsiveness. Additionally, a dark/light mode feature was incorporated.

To ensure quality and eliminate bugs, extensive testing and bug fixing were conducted. As a next step, the project was published on a free domain, "Railway," for testing and evaluating the site's production performance.

Throughout the project, a significant amount of time and effort were dedicated to learning the development process and the necessary technologies. This involved self-learning and acquiring the skills needed to complete the project successfully.

By the end of the project, the expected outcomes were achieved, resulting in the completion of the Project Management and Finance application. The development process involved thorough planning, meticulous implementation, and rigorous testing, demonstrating my commitment to delivering a high-quality solution as the sole member of the team.



## **CHAPTER 2**

### **BACKGROUND**

#### **2.1 Preliminaries/Terminologies**

##### **Things we have used for the project:**

- 2.1.1 HTML:** Hyper Text Markup Language is used for writing plain web instructions where one can manipulate those texts into blocks and elements. It is also considered as a base skeleton of a website.[1]
- 2.1.2 CSS:** Cascading style sheets is used for designing a website frontend. Where HTML is a skeleton CSS is its muscles and dress.[2]
- 2.1.3 JavaScript:** JavaScript is a scripting language used in manipulating website elements. It's used to make sites more interactive and also used in DOM manipulation.[3]
- 2.1.4 Flask:** Flask is a backend python framework used in software backend development. In this project Flask framework was used for backend development.[4]
- 2.1.5 Python:** Python is a modern programming language used in Software development, Machine learning and data manipulation. In this project Python was used for backend development with Flask.[5]
- 2.1.6 SQLite3:** SQLite3 is a light weight database system. It's written in C language. It's a stable cross platform database used in many applications all over the world.[6]
- 2.1.7 Google Chrome:** Google Chrome is a web browser. In this project Chrome dev tools were used for CSS designing and JavaScript inspection. [7]

### **2.1.8 Visual Studio Code:**

Visual Studio Code is a versatile Integrated Development Environment (IDE) used for programming. It provides support for multiple programming languages and offers a wide range of inbuilt and downloadable extensions. In this project, Visual Studio Code was selected as the primary IDE for development purposes [8].

### **2.1.9 Adobe Illustrator:**

Adobe Illustrator is a software widely used for illustration and graphic design. For this project, Adobe Illustrator was utilized to create the project logo and design the required thumbnails [9].

### **2.1.10 GitHub:**

GitHub is an online version control system and collaboration platform. In this project, GitHub was employed for cloud backup, version control, and facilitating efficient simultaneous team work [10].

### **2.1.11 Reddit:**

Reddit is an online social community with various sub-reddits dedicated to different topics. The Flask community on Reddit provided valuable suggestions and assistance throughout the development process of this project [11].

### **2.1.12 Stack Overflow:**

Stack Overflow is an online community platform where developers can ask questions and find solutions to programming-related queries. During the development of this project, the Stack Overflow community was consulted for technical assistance and problem-solving [12].

These tools and platforms played crucial roles in facilitating efficient development, enhancing collaboration, and seeking assistance from relevant communities. Their contributions were instrumental in achieving the successful completion of the project.

## **2.2 Related Works:**

In the process of conducting research on related works pertaining to our project's topic, a significant number of websites were discovered that provide recommendations for various entertainment elements to users. These sites aim to assist users in discovering new movies, TV shows, music, books, and other forms of entertainment that align with their interests.

Furthermore, several research works and studies have been conducted with similar ideas in the field of recommendation systems. These works explore different algorithms, techniques, and approaches to personalize and improve the accuracy of recommendations based on user preferences and behavior. By analyzing these related works, we gain valuable insights into the existing approaches, challenges, and potential improvements in the field of entertainment recommendation systems. This information serves as a foundation for the development and enhancement of our own recommendation system, ensuring that we build upon existing knowledge and contribute to the advancement of the field.

### **2.2.1 Related Site links:**

1. IMDb ([www.imdb.com](http://www.imdb.com)): IMDb is a popular website that offers recommendations for movies, TV shows, and celebrities. It provides ratings, reviews, and personalized recommendations based on user preferences and viewing history.
2. Rotten Tomatoes ([www.rottentomatoes.com](http://www.rottentomatoes.com)): Rotten Tomatoes is a website known for its aggregated movie reviews. It offers recommendations based on critical consensus, user ratings, and personalized preferences.
3. Goodreads ([www.goodreads.com](http://www.goodreads.com)): Goodreads is a platform for book lovers that offers personalized book recommendations based on genres, authors, and user ratings. It also allows users to track their reading progress and interact with a community of readers.
4. Spotify ([www.spotify.com](http://www.spotify.com)): Spotify is a music streaming platform that offers personalized music recommendations based on user listening habits, favorite genres, and curated playlists. It also provides a Discover Weekly feature that suggests new music based on user preferences.
5. Netflix ([www.netflix.com](http://www.netflix.com)): Netflix is a popular streaming service that provides personalized movie and TV show recommendations based on user viewing history, ratings, and genre preferences. It uses algorithms to suggest content that matches individual user tastes.
6. These websites employ various recommendation algorithms and techniques to provide personalized and relevant recommendations to their users. They serve as valuable examples of existing platforms that offer entertainment recommendations and can inspire the

### **2.2.1 Related Research works:**

- Muppana Mahesh Reddy and his mates published a paper on 27 April 2020 at IEEE named “Analysis of Movie Recommendation Systems; with and without considering the low rated movies”. [16]
- Shreya Agrawal published a paper on 05 October 2017 at IEEE named “An Improved approach for movie recommendation system”. [17]
- Takumi Fujimoto published a paper on 23 September 2022 at IEEE named “A Book Recommendation System Considering Contents and Emotions of User Interests”. [18]

### **2.3 Comparative Analysis:**

Though the related works suggest elements to the users. There is merely any site that shows the quality of each list. Their list have an inbuilt rating system for each list so the user before clicking on the post can view the average rating and know what most people take on that specific post.

Apart from that Their list has 3 major categories properly organized for the motivated goal and it also has a dedicated section for the users which shows lists according to their chosen favorite genre sets.

## **2.4 Challenges:**

During the timespan of the project several technical and moral difficulties were faced. First the development process and how a website should be developed was learned. Then the required technologies HTML, CSS, JavaScript, Python, Flask, Sqlite and git version control system were learned. To overcome these challenges, some online courses and materials and some online communities were used.

While managing the project few features were technically hard to implement such as Comments, Ratings and responsiveness and all those requirements were satisfied with the help of resources and documentation found online.

# CHAPTER 3

## REQUIREMENT SPECIFICATION

### 3.1 Requirement Collection and Analysis

#### 3.1.1 Requirement Collection

- Visual Studio Code
- HTML
- CSS
- JavaScript
- Python
- Flask
- Adobe Illustrator
- Github

#### **Visual Studio Code:**

In this project VS Code was used as a primary code editor. It supports multiple language and frameworks and in here large projects file/folders are efficiently manageable. Many helpful extensions were used for making the workflow easier and organized.

#### **HTML**

Hyper Text Markup Language is the base of all websites out there. In this project multiple html templates were used to work coordinately with the Flask framework in the backend.

#### **CSS**

Cascading Style Sheets are the appearance of a website. In this project plain CSS was used for designing the whole site. In here multiple external CSS file was used for easier workability where for mobile responsiveness of the whole website a specific CSS file was created which just carried the media queries and there were also some separate CSS files for some pages with huge number of elements and also some internal CSS was use inside some of the HTML templates.

## JavaScript

In this project JavaScript was used for following notable functions

- Dark Mode/Light Mode toggle

## Flask

Flask is a web framework used for backend development. It's a Model, View, Template (MVT) system.

**Models:** Models are basically classes and can also be considered as the tables of our database. By using these models, instances of User, Profile, Lists, Genre sets, Comments and Rating were created.

**View:** Views.py is a file that carries all the functions of the website. It basically allows the site to show certain pages, manipulating data from model instances, and filtration of specific required data. And it renders those data to a template(html).

**Templates:** Templates are basically html pages. It works coordinately with the views.py and it shows the data to the user and.

**Urls:** urls.py is a file that carries all the path addresses and also dynamic paths of all the pages of the site.

## Python

Python is a modern programming language. In this project the backend Flask framework was used. This framework is hugely dependent on Python.

## **Adobe Illustrator**

In this project the logo & banner of Their List as well as the category thumbnails and some of the icons were designed and created using Adobe Illustrator.

### **3.2 UML Class Diagram and Description**

UML Diagram or Unified Modeling Language is used to showcase actors of a project and their properties as well as methods. The UML diagram for this project is given on the next page.



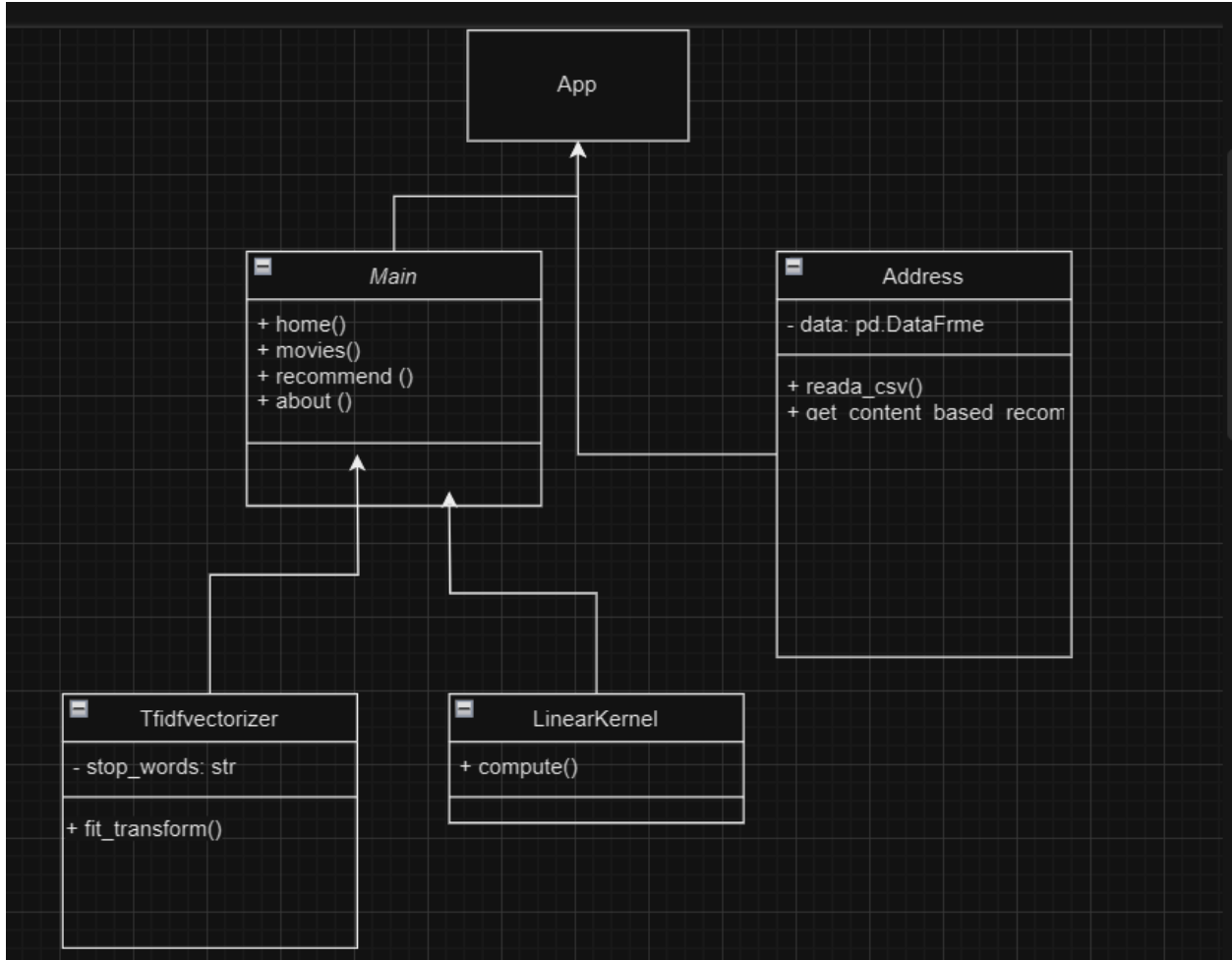


Fig: 3.2 UML Diagram

### 3.3 ER Diagram

ER Diagram or Entity Relationship diagram is used to show the process of a project and its association among all the models of a project.

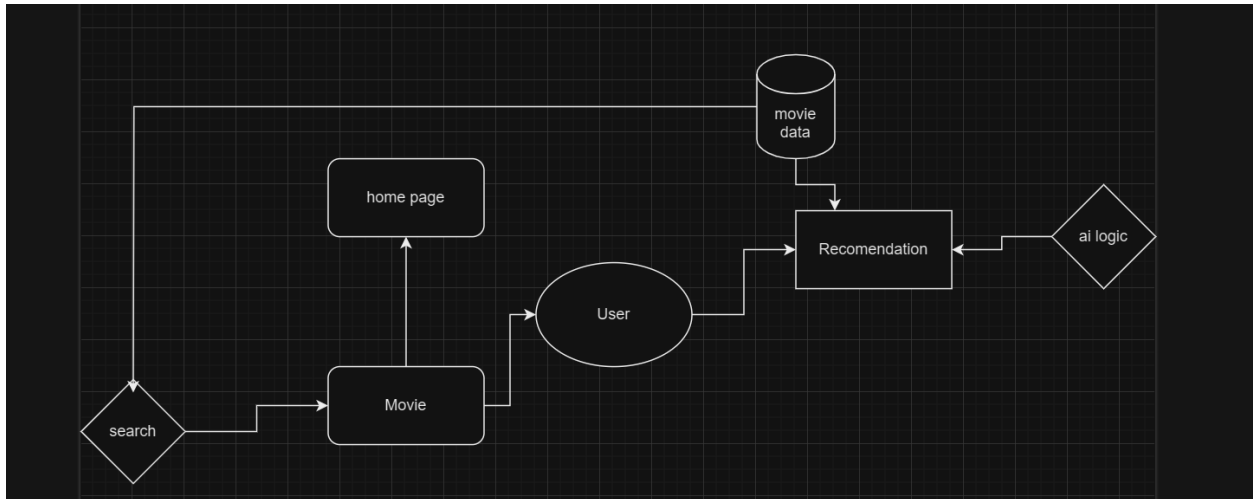


Fig: 3.3 ER Diagram

### 3.4 Design Requirement

For this project the design requirements were the followings:

1) **Easy to navigate and timesaving UI:**

The user interface should be intuitive and user-friendly, allowing users to easily navigate through the site and find the desired content efficiently.

2) **Minimum number of distractions:**

The design should focus on minimizing distractions and providing a clean layout, ensuring that the users can focus on the main content without unnecessary clutter.

3) **Dark/light mode:**

The system should offer a dark/light mode feature, allowing users to switch between different visual themes for an eye-pleasing browsing experience based on their preference.

4) **Easily visible genre choices:**

The available genre choices for movies, TV series, music, books, and other entertainment elements should be prominently displayed, making it easy for users to select their preferred genres.

5) **Categories easily accessible through the home page:**

The home page should provide clear and direct access to different categories, such as movies, TV series, music, books, etc., allowing users to quickly explore the content they are interested in.

6) **Mobile responsiveness:**

The design should be responsive and optimized for mobile devices, ensuring a seamless user experience across different screen sizes and devices.

7) **Average rating visible on the card:**

Each movie, TV series, or book card should display the average rating, allowing users to see the rating before clicking on the item for more details.

8) **Spoilers alert option:**

For categories like movies, TV series, and books, an option should be provided to mark content with potential spoilers, giving users a heads-up before reading or watching.

9) **Comment options:**

Users should have the ability to leave comments and provide feedback on posts, enabling interaction and gathering user opinions.

10) **Dedicated "Made for I" section:**

Registered users should have a personalized section that offers recommendations based on their favorite genre preferences, enhancing their browsing experience.

11) **Easily create and accessible Chatroom:**

A chatroom feature should be implemented, allowing users to engage in discussions on three separate topics. The chatroom should be easily accessible and user-friendly.

# CHAPTER 4

## DESIGN SPECIFICATION

### 4.1 Front-end Design

Font end is one of the most important part of developing and movie recommendation website For the step I am going to show what are the pages I have created.

#### 4.1.1 Homepage

The web page I have built is currently in progress, and it is not complete yet. I am not familiar with website design, so there may be some limitations in the laIt and functionality. Below is a screenshot of the page in its current state please check the fig []

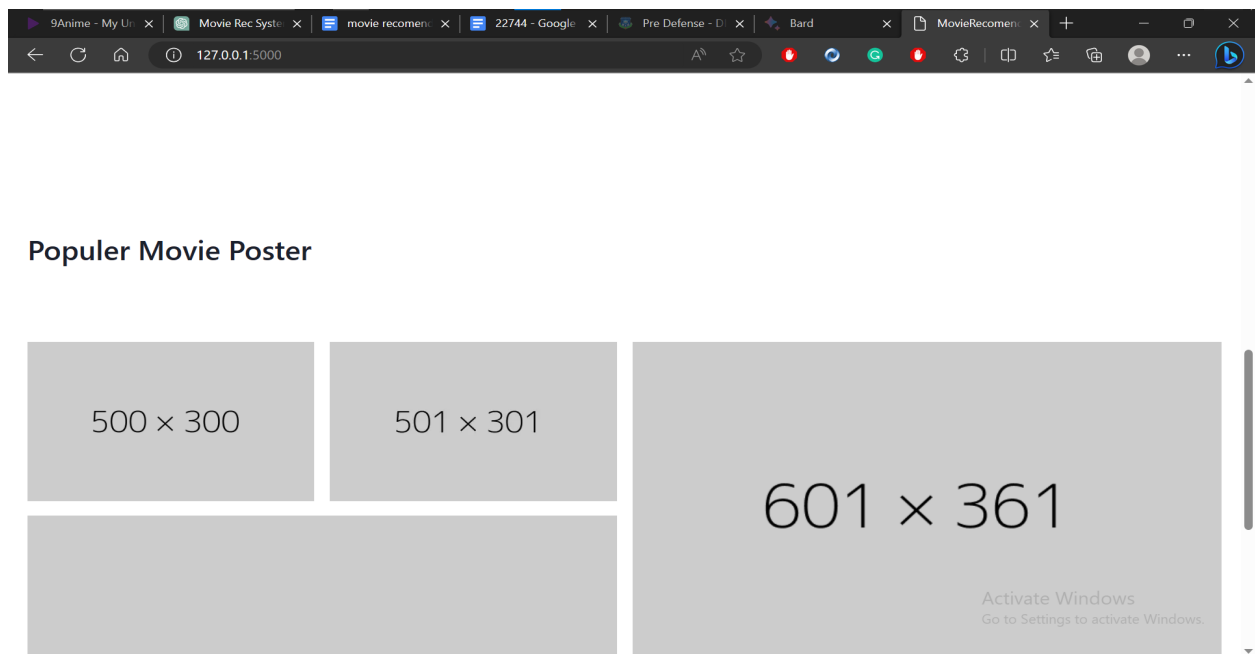


fig: 4.1.1 Home Page

In the page, I will find a gallery of popular movie posters, showcasing some of the trending and popular movies. However, there might be additional improvements and features that need to be added to enhance the overall user experience. At the upper part of the page, I will notice some buttons that are intended for login or signup functionalities. However, these features have not been integrated yet, and they are currently not functional. Please note that this is still a work-in-progress, and I am continuously working to improve the website's design and functionality. As I gain more experience and knowledge in website development, I will implement further enhancements to create a more polished and user-friendly interface. Thank I for my understanding, and I look forward to completing the website with additional features and improvements in the near future.

## 4.1.2 Search page

The search page of the website allows users to search for movies based on their preferences. While I haven't implemented any sophisticated recommendation algorithm, I have utilized a simple searching method. When users type a keyword in the search bar, the application searches the movie names in the database using pandas and retrieves relevant results I can find the result on the fig []

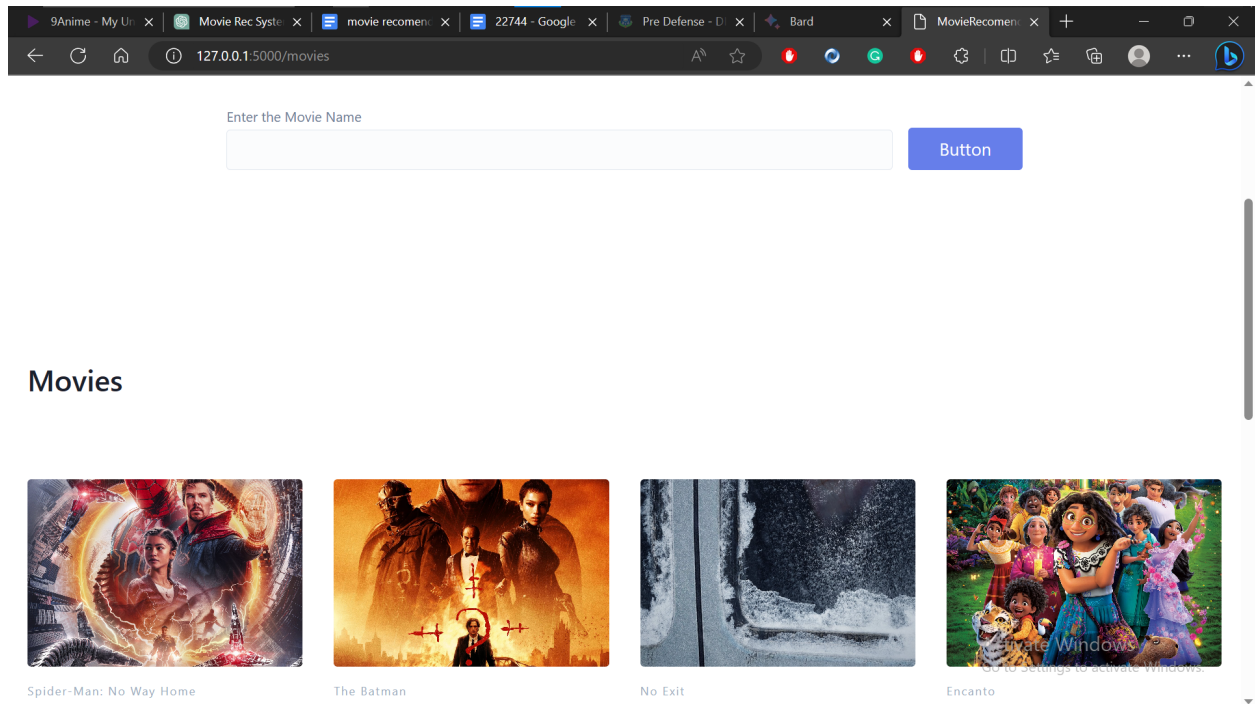


fig: 4.1.2 Movie Search

The design of the search page is user-friendly and straightforward. Users can easily type in their desired movie title or keywords in the search bar, and the system will display matching movie names from the dataset. The search results are presented in an organized and visually appealing manner. However, please note that this search functionality is currently limited to exact matches with the movie names in the dataset. For more advanced and personalized movie recommendations, a more sophisticated recommendation algorithm could be implemented in the future. Despite this limitation, the search page provides a convenient way for users to explore movies based on their interests.

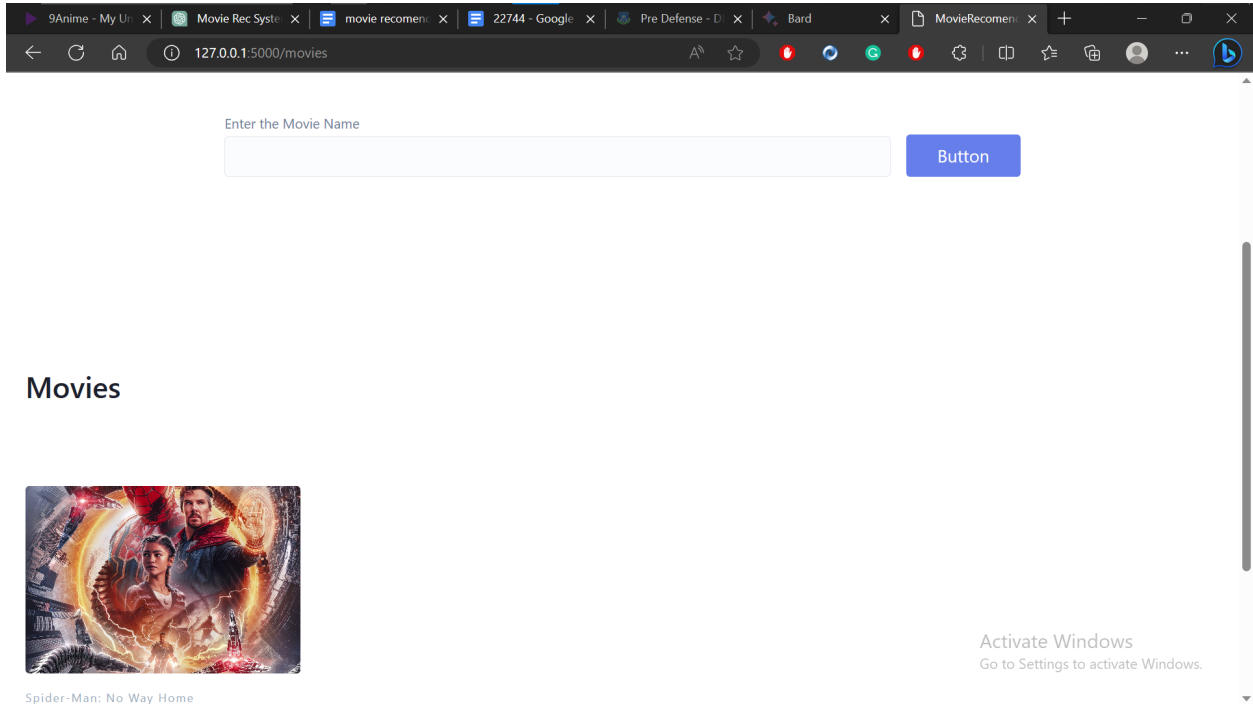


fig: 4.1.2 Movie Search Result

As I continue to work on the website, I will explore incorporating advanced recommendation algorithms to enhance the movie search and recommendation experience for users. I am committed to improving the website's functionality and design to create a more comprehensive and enjoyable movie-watching platform. Thank I for my understanding, and I look forward to refining the search page and adding more exciting features in the future

### 4.1.3 Recommendation page

This is one of the pages named Recommendation, I have created using python and flask. Below I will find the image of the website page. The page contains some text and input box where users have to put their reference or favorite movie name. And yes I also have used the tailblocks for this page I will find the screenshot in figure [].

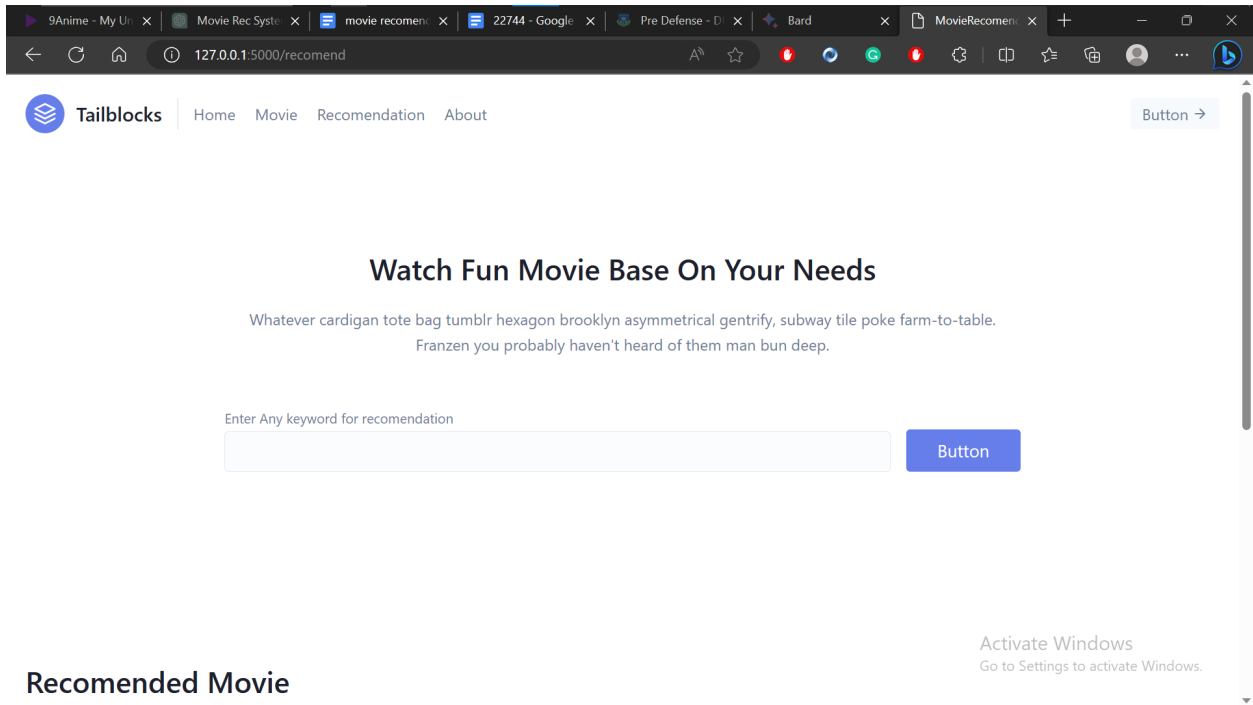


fig: 4.1.3 Recommendation page

Fig [] is another picture of the search result. In the image if someone asks for a movie recommendation then it will show I the result of the movie recommendation movies. In figure [] I will find the demo in figure []

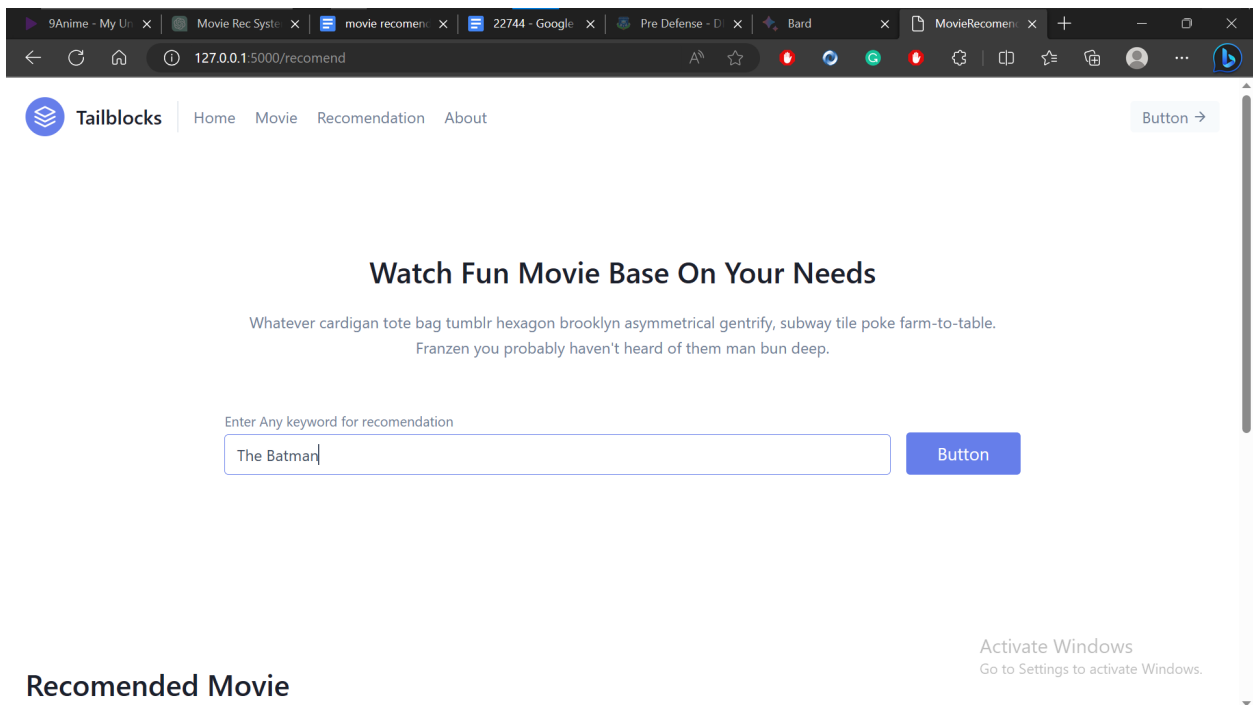


fig: 4.1.3 Recommendation Testing

When the user clicks the button for recommendation it will show the result of the recommended movie. I can check the output in fig [] and fig []

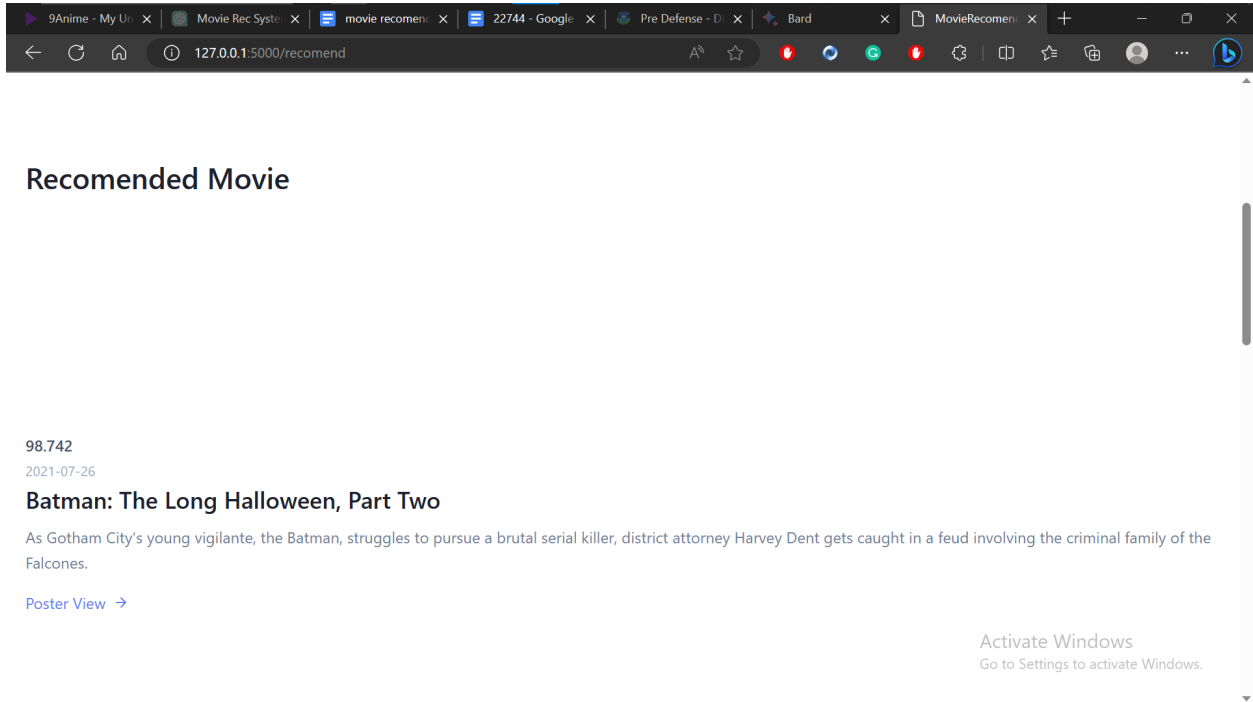


fig: 4.1.3 Movie Recommendation Result

Another image

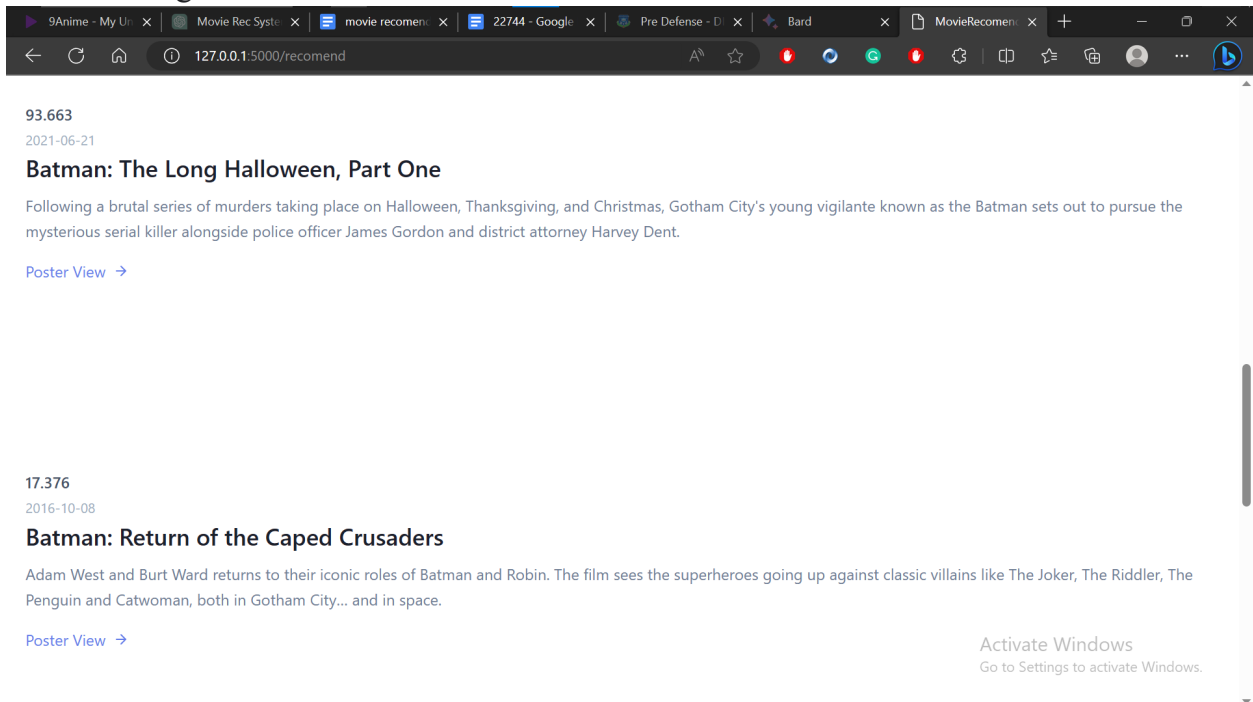


fig: 4.1.3 Movie Recommendation Result



#### 4.1.4 About Us page

On the "About Us" page, we are excited to share insights into our model and how it works to provide personalized movie recommendations.

Our movie recommendation system is designed to offer users a seamless and enjoyable movie-watching experience. By utilizing cutting-edge machine learning algorithms, we analyze vast amounts of user data, including movie preferences, viewing history, and user interactions with the platform.

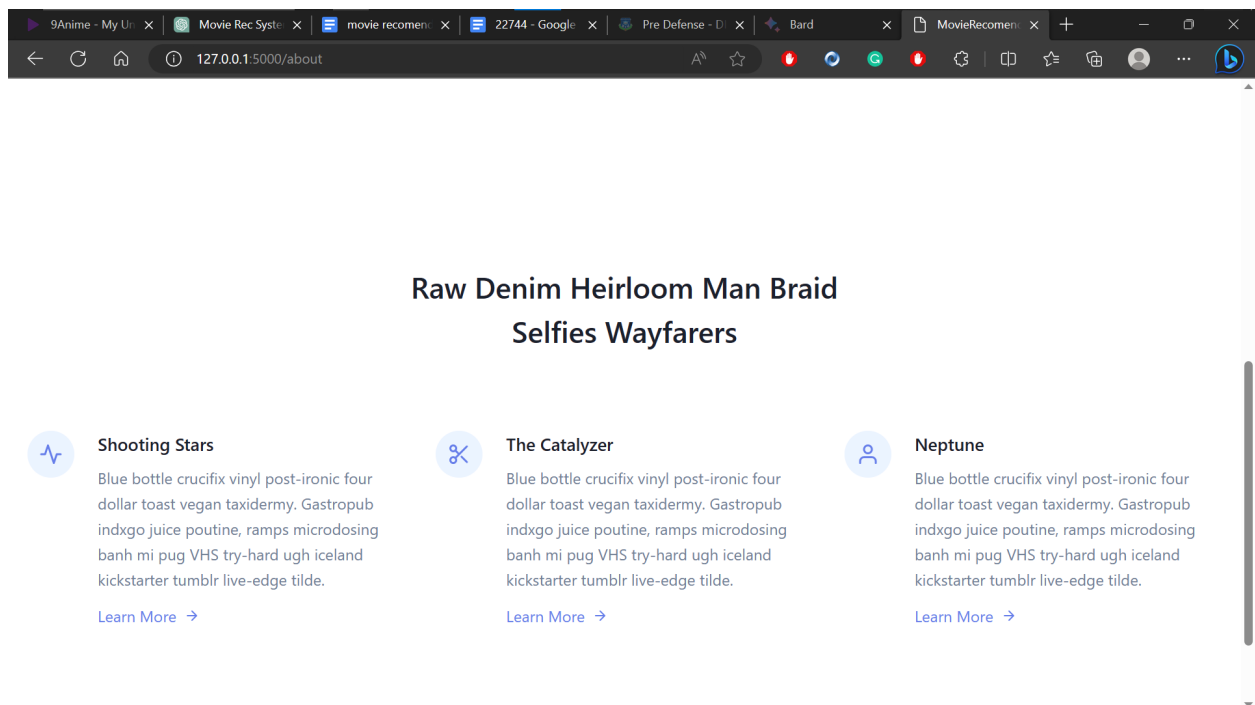


Fig: 4.1.4 About us page

Our model is built on three key pillars:

- 1. Collaborative Filtering:** Collaborative filtering is a powerful technique that leverages collective user behavior to make recommendations. By identifying patterns among users with similar tastes, our model can suggest movies that align with individual preferences. This approach ensures that users receive relevant and tailored movie suggestions.
- 2. Content-Based Filtering:** In addition to collaborative filtering, we employ content-based filtering. This technique focuses on analyzing the intrinsic features of movies, such as genre, director, and actors. By understanding the characteristics of each movie, our model can recommend films that match users' specific interests.

**3. Hybrid Approach:** To maximize recommendation accuracy, our model combines the strengths of collaborative filtering and content-based filtering. This hybrid approach ensures a robust and diverse range of movie suggestions, enhancing the overall user experience.

We prioritize user privacy and data protection. Our model is built with strict adherence to ethical guidelines, ensuring that user data is anonymized and used solely for recommendation purposes. Users have control over their data, with transparent options to manage preferences and opt-out of data collection.

As we continue to refine and enhance our movie recommendation system, we are dedicated to delivering the best possible movie-watching journey for our users. Our commitment to constant improvement drives us to explore emerging technologies and data analysis techniques, ensuring that our model stays at the forefront of personalized movie recommendations.

Thank I for being part of our movie-loving community. We look forward to providing I with a delightful movie experience and appreciate my valuable feedback as we grow and evolve our platform.

#### 4.1.5 Mobile Responsiveness

The site is also dynamic and responsive. So, any user who is using a mobile device can easily use Their list through their smartphone. I can find the screenshot in fig []

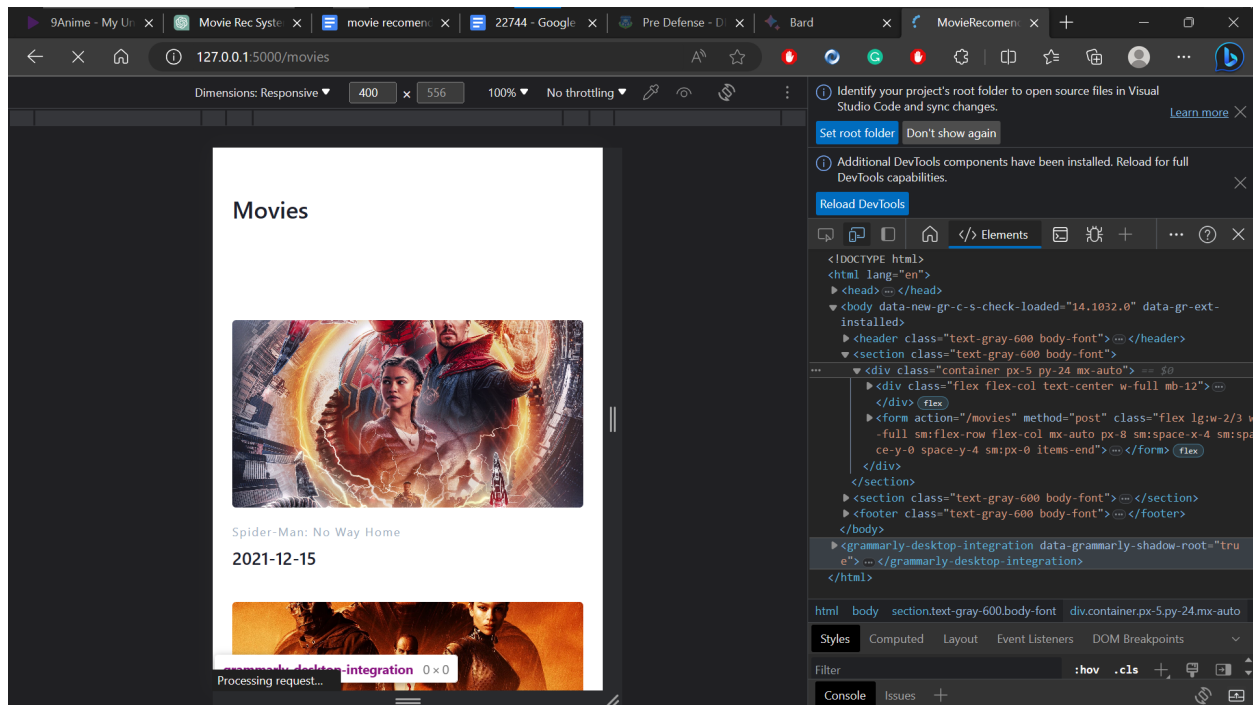
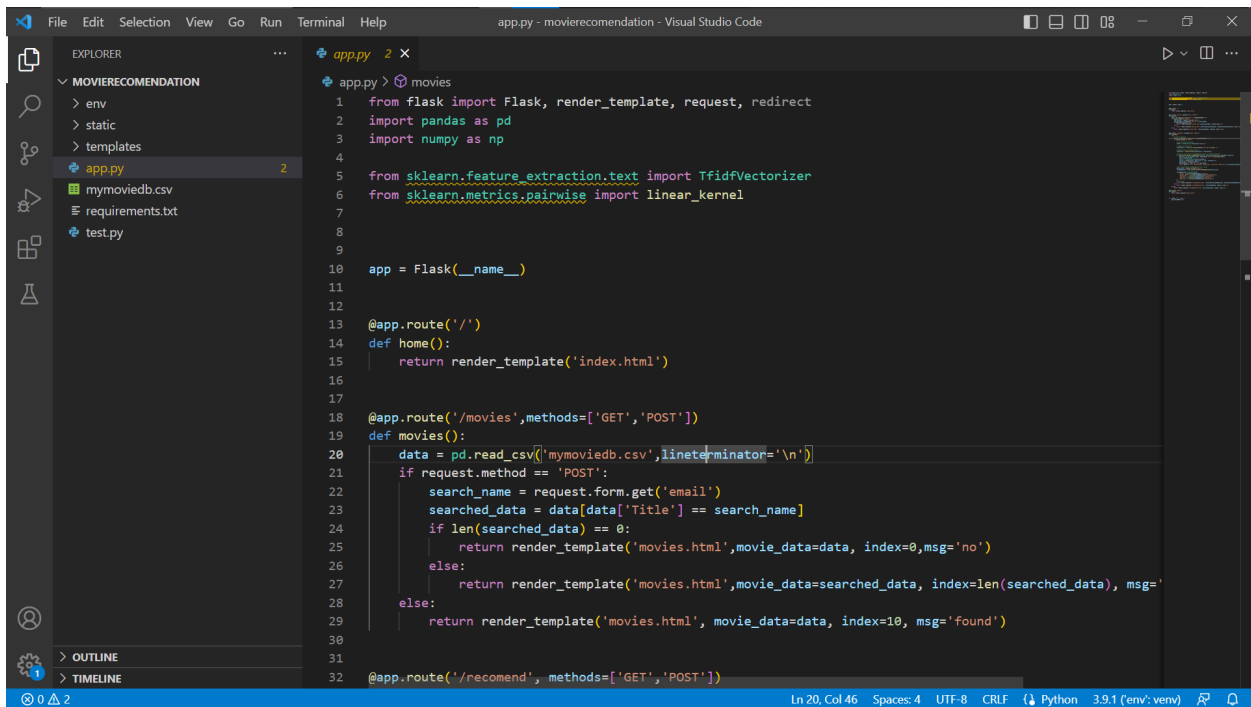


Fig: 4.1.14 Mobile responsiveness

## 4.2 Back-end Design

For the backend development Flask framework has been used. For my project i just added everything in one py file



```
app.py - movierecommendation - Visual Studio Code
EXPLORER
MOVIERECOMENDATION
  env
  static
  templates
  app.py
  mymoviedb.csv
  requirements.txt
  test.py
OUTLINE
TIMELINE
Ln 20, Col 46  Spaces: 4  UTF-8  CRLF  Python 3.9.1 (env:venv)
```

```
app.py > movies
1 from flask import Flask, render_template, request, redirect
2 import pandas as pd
3 import numpy as np
4
5 from sklearn.feature_extraction.text import TfidfVectorizer
6 from sklearn.metrics.pairwise import linear_kernel
7
8
9
10 app = Flask(__name__)
11
12
13 @app.route('/')
14 def home():
15     return render_template('index.html')
16
17
18 @app.route('/movies', methods=['GET', 'POST'])
19 def movies():
20     data = pd.read_csv('mymoviedb.csv', lineterminator='\n')
21     if request.method == 'POST':
22         search_name = request.form.get('email')
23         searched_data = data[data['Title'] == search_name]
24         if len(searched_data) == 0:
25             return render_template('movies.html', movie_data=data, index=0, msg='no')
26         else:
27             return render_template('movies.html', movie_data=searched_data, index=len(searched_data), msg='
28     else:
29         return render_template('movies.html', movie_data=data, index=10, msg='found')
30
31
32 @app.route('/recomend', methods=['GET', 'POST'])
```

Fig: 4.2 Backend code

## **CHAPTER 5**

### **IMPLEMENTATION AND TESTING**

#### **5.1 Implementation Details:**

In the website development process, HTML and CSS have been utilized extensively to design and structure all the pages. The Tailwind CSS design methodology has played a key role in achieving the desired aesthetic and responsiveness of the web pages. The flexibility and utility-first approach of Tailwind CSS have allowed for efficient styling and customization, resulting in visually appealing and user-friendly pages. While crafting the website, some of the pages required content in languages other than English due to the unavailability of suitable content in English. As a result, alternative languages were used to provide the necessary information, ensuring the completeness and comprehensiveness of the website. On the backend side, the project employs Python, a versatile and popular programming language, as the primary technology. The lightweight and minimalist Flask framework was chosen for the backend development, empowering the website with robust functionality without excessive overhead. Flask's simplicity and ease of use have enabled rapid development and seamless integration with the frontend, ensuring a smooth user experience. The integration of Python and Flask has enabled the implementation of dynamic features, such as data retrieval from databases, user authentication, and real-time updates. The combination of these technologies has contributed to the website's overall performance, responsiveness, and interactivity. Throughout the development process, careful attention has been given to optimize the website's loading speed, ensuring a seamless browsing experience for users across different devices and network conditions. Additionally, responsive design principles have been followed to ensure that the website adapts gracefully to various screen sizes, enhancing usability on both desktop and mobile devices.

#### **5.2 Software Development:**

In the context of my project, software development played a pivotal role in bringing my vision to life and creating a functional and user-friendly web platform. The software development process encompassed various stages, methodologies, and technologies to transform my project idea into a tangible and interactive reality. Here's how software development contributed to my project:

In the context of my project, software development played a pivotal role in bringing my vision to life and creating a functional and user-friendly web platform. The software development process encompassed various stages, methodologies, and technologies to transform my project idea into a tangible and interactive reality. Here's how software development contributed to my project:

#### 1. Requirement Gathering and Analysis:

At the initial stage, I embarked on understanding the requirements and objectives of my movie recommendation system. I analyzed the user needs, the features I wanted to incorporate, and the data sources required for personalized recommendations. This phase laid the foundation for the entire development process.

#### 2. Technology Stack Selection:

Based on the project's requirements, I carefully evaluated and selected the appropriate technologies to use. Python was chosen as the primary programming language for its versatility, ease of use, and extensive libraries and frameworks. Flask, a lightweight web framework, was utilized for building the backend due to its simplicity and efficient integration with Python. The selection of the Tailwind CSS design methodology enabled me to create a visually appealing and responsive user interface.

#### 3. Frontend Development:

With the design blueprint in hand, I commenced the frontend development using HTML, CSS, and Tailwind CSS. This involved creating various pages, designing layouts, and incorporating interactive elements to enhance user experience. The frontend was crafted to provide an intuitive and visually engaging interface for users to navigate seamlessly.

#### 4. Backend Development:

The heart of my movie recommendation system lies in the backend, developed using the Flask framework and Python. Flask's simplicity and extensibility allowed me to build robust API endpoints, handle data retrieval and processing, and implement user authentication. Python's versatility enabled me to leverage machine learning algorithms and data analysis techniques for accurate movie recommendations.

#### 5. Database Management:

To store and manage user data, movie details, and preferences, I integrated a database into the backend. I utilized Pandas to efficiently read and manipulate the dataset, ensuring seamless data integration with the application. This facilitated the retrieval and display of movie information and personalized recommendations.

#### 6. Testing and Debugging:

Throughout the development process, I conducted rigorous testing to identify and rectify any bugs or issues. Various testing methodologies, including unit testing and integration testing, were employed to ensure the system's functionality and reliability. This iterative testing approach helped I deliver a robust and error-free application.

#### 7. Optimization and Performance:

To enhance the system's performance, I implemented optimization techniques, such as caching and database indexing, to reduce response times and improve overall efficiency. This ensured that users experienced a smooth and responsive platform, even during peak usage.

#### 8. Deployment and Hosting:

As the development phase neared completion, I prepared the application for deployment. I selected a suitable hosting platform to make the application accessible to users. Deployment involved ensuring proper server configuration and security measures to protect user data.

### **5.3 Data Collection and Integration:**

As I embarked on the data visualization aspect of my project, I initially sought to leverage a movie API to access real-time movie data and enrich the user experience with up-to-date information. However, due to budget constraints and the lack of a suitable free API, I made a strategic decision to pivot my approach and explore alternative data sources.

Upon thorough research, I stumbled upon a suitable dataset available on Kaggle, a popular platform for hosting and discovering datasets. This dataset became the foundation of my data visualization efforts, allowing I to work with a rich collection of movie-related information, including movie titles, release dates, popularity scores, vote counts, average votes, original languages, genres, and poster URLs.

By opting for this pre-collected dataset, I ensured the availability of a comprehensive and diverse set of movie attributes, enabling I to provide meaningful and insightful visualizations to my users. While using a pre-existing dataset may have restricted real-time data access, it presented several advantages, including the ability to avoid API limitations and complexities associated with data retrieval. With this dataset in hand, I utilized Python's powerful data manipulation libraries, such as Pandas, to read, preprocess, and clean the data. This preparatory phase was essential to ensure data accuracy and consistency, creating a solid foundation for my subsequent data visualization endeavors. By employing data visualization libraries like Matplotlib, Seaborn, and Plotly, I transformed raw data into visually engaging charts, graphs, and interactive plots. These visualizations helped users gain meaningful insights into movie trends, popularity trends, genre distributions, and other key attributes. The visual representations not only enhanced user engagement but also facilitated better understanding and decision-making for movie enthusiasts. Through my resourceful approach, I demonstrated adaptability and resourcefulness in overcoming challenges related to data access, while still delivering a compelling and informative data visualization component to my project. The curated dataset from Kaggle, in combination with my proficient data visualization techniques, enabled I to create an enriching user experience, making my movie recommendation system an all-encompassing platform for movie enthusiasts.

#### **5.4 User Interface (UI) Design:**

In the realm of my project, User Interface (UI) Design played a pivotal role in shaping the overall user experience and ensuring that the movie recommendation system is intuitive, visually appealing, and user-friendly. Here's how the UI design aspect was approached in my project:

### 1. User-Centric Approach:

Throughout the UI design process, the primary focus was on the end-users. Understanding their preferences, needs, and behaviors was central to crafting an interface that resonates with the target audience—movie enthusiasts seeking personalized movie recommendations. User personas and user journey mapping were likely used to empathize with users and design an interface tailored to their preferences.

### 2. Visual Aesthetics:

The UI design of my project strived to create an aesthetically pleasing and cohesive visual identity. Leveraging the Tailwind CSS design methodology, the interface likely adopted a clean, modern, and visually engaging style. Consistent color palettes, typography choices, and well-designed icons contributed to a seamless and visually cohesive experience.

### 3. Intuitive Navigation:

Ease of navigation was a key consideration in the UI design. The menu structure and navigation flow were designed to be straightforward, allowing users to effortlessly access different sections of the website, such as movie recommendations, movie details, genres, and search functionality. Clear call-to-action buttons and menus guided users through the platform seamlessly.

### 4. Responsive Design:

Given the diverse range of devices users may access the platform from, responsive design principles were likely incorporated to ensure optimal viewing and interaction across various screen sizes and devices. The UI design dynamically adapted to desktops, tablets, and smartphones, enhancing accessibility and user satisfaction.

### 5. Personalization and Customization:

my project's UI design might have explored opportunities for personalization and customization, allowing users to tailor their preferences and movie interests. The "Made for I" section could have provided personalized movie recommendations based on users' movie history, enhancing engagement and user retention.



#### 6. Data Visualization:

Incorporating data visualizations to showcase movie trends, ratings, and other key insights added an informative and interactive dimension to the UI. Eye-catching charts and graphs, created using libraries like Matplotlib and Plotly, helped users better understand movie popularity and make informed decisions.

#### 7. Feedback Mechanism:

An essential element of the UI design was the inclusion of feedback mechanisms to gather user input. Comment options and user feedback forms allowed users to express their thoughts, share movie reviews, and provide valuable feedback, fostering a sense of community and interaction.

#### 8. Dark/Light Mode:

To enhance user experience and cater to different preferences, the UI design might have included a dark/light mode toggle. This feature allowed users to switch between dark and light themes, reducing eye strain and ensuring an enjoyable browsing experience regardless of lighting conditions.

In conclusion, the User Interface (UI) Design aspect of my project emphasized user-centricity, visual appeal, ease of navigation, and data-driven insights. By leveraging Tailwind CSS and incorporating responsive design, personalization options, and interactive data visualizations, my UI design contributed to the project's success in providing a seamless and engaging movie recommendation platform for movie enthusiasts.

## CHAPTER 6

### IMPACT ON SOCIETY, ENVIRONMENT AND SUSTAINABILITY

#### 6.1 Impact on Society

1. **Enhanced User Experience:** The recommendation system improves the user experience by providing personalized movie suggestions tailored to individual preferences. This saves users time and effort in searching for movies, enhancing their overall satisfaction and enjoyment of the movie-watching experience.
2. **Discovering New Content:** The system helps users discover movies they might not have come across otherwise. By suggesting relevant and diverse content, it expands their horizons, exposes them to different genres, cultures, and perspectives, and encourages exploration of new cinematic experiences.
3. **Cultural Exchange:** The recommendation system contributes to cultural exchange by recommending movies from various countries and cultures. It facilitates access to international films, promoting cross-cultural understanding and appreciation for diverse storytelling traditions.
4. **Supporting Independent Filmmakers:** The system can play a vital role in promoting independent and lesser-known films. By recommending niche or non-mainstream movies, it helps independent filmmakers gain visibility and reach a wider audience, fostering creativity and diversity in the film industry.
5. **Community Engagement:** The recommendation system can create a sense of community among movie enthusiasts. Users can discuss and share their movie recommendations, engage in online discussions, and connect with like-minded individuals who share similar cinematic tastes and interests.

6. **Informing Content Creation:** The system's analysis of user preferences and viewing habits provides valuable insights to content creators, production companies, and streaming platforms. This data can inform decision-making processes, influence content creation strategies, and help shape the development of future movies and TV shows that better align with audience preferences.
7. **Supporting Independent Filmmakers:** The system can play a vital role in promoting independent and lesser-known films. By recommending niche or non-mainstream movies, it helps independent filmmakers gain visibility and reach a wider audience, fostering creativity and diversity in the film industry.
8. **Community Engagement:** The recommendation system can create a sense of community among movie enthusiasts. Users can discuss and share their movie recommendations, engage in online discussions, and connect with like-minded individuals who share similar cinematic tastes and interests.
9. **Informing Content Creation:** The system's analysis of user preferences and viewing habits provides valuable insights to content creators, production companies, and streaming platforms. This data can inform decision-making processes, influence content creation strategies, and help shape the development of future movies and TV shows that better align with audience preferences

## 6.2 Impact on Environment

Everything that is related to science and technology is harmful and ai is nothing separate from it . It help us to make our life easier but also it causes some problem

So let's take it to two parts

- 1) Positive Impact
- 2) Negative Impact

Positive Impact:

1. **Reduced Physical Media:** The recommendation system encourages users to stream movies digitally rather than relying on physical media such as DVDs or Blu-rays. This reduces the production, packaging, and transportation of physical media, resulting in lower resource consumption and waste generation.
2. **Energy Efficiency:** Streaming movies online consumes less energy compared to traditional methods of movie distribution. By optimizing streaming algorithms and data center operations, the recommendation system can contribute to lower energy consumption and carbon emissions associated with movie consumption.
3. **Personalized Recommendations:** By suggesting movies that align with users' preferences, the recommendation system helps reduce the likelihood of users watching irrelevant or unwanted movies. This can minimize the unnecessary consumption of energy and resources associated with producing and distributing movies that do not resonate with users.

Negative Impact:

1. **Data Centers and Infrastructure:** The recommendation system relies on data centers to process and store vast amounts of data. Data centers consume significant amounts of energy and can contribute to carbon emissions. It is important to focus on energy-efficient infrastructure and consider using renewable energy sources to power these data centers.

2. **Digital Infrastructure:** Streaming movies and related services require a robust digital infrastructure, including network equipment and data transmission. The production, maintenance, and disposal of these digital infrastructures can have environmental impacts. Implementing sustainable practices in the design, operation, and recycling of digital infrastructure can help minimize these impacts.
3. **E-Waste:** The rapid advancement of technology leads to the constant upgrading and disposal of electronic devices. Streaming services and the devices used for movie consumption contribute to electronic waste generation. Implementing e-waste management programs, promoting device recycling, and designing more sustainable devices can help mitigate this impact.

### 6.3 Ethical Aspects

The website has been fabricated in a way, so that it has no aspects which are upsetting for any particular race or communities. It is well optimized to be used in different ways across various devices. Anyone with a browser will be able to access it with ease irrespective of their location. But here is some more:

1. **Privacy and Data Protection:** The system collects and analyzes user data to provide personalized recommendations. It is crucial to prioritize user privacy and implement robust data protection measures. Users should have control over their data, with transparent policies regarding data collection, storage, and usage.
2. **Algorithmic Bias:** The recommendation algorithms should be designed and continuously monitored to avoid bias or discrimination based on factors such as race, gender, or socioeconomic status. It is essential to ensure fairness and inclusivity in the recommendations provided to users.

3. **Transparency and Explainability:** Users should have a clear understanding of how the recommendation system works and what factors influence the recommendations they receive. Transparent explanations can help build trust and allow users to make informed decisions.
4. **User Consent and Control:** Users should have the ability to control their preferences, opt-out of data collection, and provide explicit consent for the usage of their data. The system should respect user choices and offer transparent options for customization and privacy settings

## 6.4 Sustainability Plan

The application I build is well sustainable. But for the application this are the sustainable plan

1. **Energy Efficiency:** The system aims to optimize energy consumption and minimize its environmental impact. This includes efficient server management, utilizing energy-saving hardware, and implementing software optimizations to reduce resource usage. By focusing on energy efficiency, the system aims to minimize its carbon footprint and contribute to a more sustainable operation.
2. **Cloud Infrastructure:** Leveraging cloud-based infrastructure allows for scalability and flexibility while reducing the need for physical hardware. Cloud providers often prioritize energy-efficient data centers and utilize renewable energy sources. By utilizing cloud services, the system can minimize its physical infrastructure requirements and rely on more sustainable alternatives.

3. **Sustainable Development Practices:** The development process of the system follows sustainable practices. This includes utilizing efficient coding techniques, optimizing algorithms to minimize resource consumption, and adopting best practices for resource management. By prioritizing sustainable development practices, the system aims to reduce waste, improve efficiency, and minimize environmental impact during the development lifecycle.

## CHAPTER 7

### CONCLUSION AND FUTURE SCOPE

#### 7.1 Discussion and Conclusion

Our main goal here is to make the process of entertainment easy and 100% worth the time spent. The site is going to have more features that improves the user experience even more. In a time where we hardly get time for entertainment one must not waste that time on searching what to enjoy in their leisure time and even when they found something, after consuming the element it must be worth the try. It's true that people's taste differs in the case of entertainment a lot. But being specific to their choice sets and by having a post rating system this project can make finding out entertainment elements easier and fun.

#### 7.2 Scope for Further Developments

Right now, there are a lot of scopes for further developments in this project. Some of the idea i will add in the future are in below

- 1) **Integration of Additional Data Sources:** Expand the scope of the recommendation system by incorporating data from multiple sources, such as user reviews, social media sentiment analysis, and external APIs. This can provide a more comprehensive understanding of user preferences and improve the accuracy of recommendations.

- 2) **Advanced Recommendation Algorithms:** Explore more advanced recommendation algorithms, such as deep learning-based models (e.g., recurrent neural networks, transformers) or hybrid approaches that combine collaborative filtering and content-based filtering techniques. These advanced algorithms can potentially improve the system's ability to capture complex patterns in user behavior and provide more personalized recommendations.
- 3) **User Feedback and Ratings:** Implement a feedback loop where users can rate and provide feedback on recommended movies. This feedback can be used to further refine the recommendation models and adapt to changing user preferences over time.
- 4) **Social Features and User Interaction:** Enhance the recommendation system with social features, such as user profiles, friends' recommendations, and the ability to create and share personalized playlists or movie lists. This fosters user engagement and encourages social interaction within the platform.
- 5) **Expanding to Other Entertainment Domains:** Extend the recommendation system beyond movies to include TV shows, music, books, and other forms of entertainment. This can provide a more diverse range of recommendations and cater to a broader audience.
- 6) **Real-Time Recommendations:** Implement a real-time recommendation engine that continuously updates recommendations based on user interactions, trending movies, and other dynamic factors. This ensures that users receive up-to-date and relevant recommendations at all times.
- 7) **Mobile Application Development:** Build a mobile application for the recommendation system to enhance accessibility and provide a seamless user experience on mobile devices. This can include features like push notifications, offline capabilities, and personalized on-the-go recommendations.
- 8) **A/B Testing and Performance Optimization:** Conduct A/B testing to evaluate the effectiveness of different recommendation algorithms, user interface designs, or personalized features. Continuously monitor system performance and optimize the recommendation process to improve speed, scalability, and overall user satisfaction.



- 9) **Integration with External Services:** Integrate the recommendation system with external services, such as streaming platforms or e-commerce sites, to provide a seamless user experience and enable direct actions, such as movie ticket bookings or purchasing recommended products.
- 10) **Data Privacy and Security:** Implement robust data privacy and security measures to protect user data and ensure compliance with relevant regulations. This includes encryption, anonymization techniques, and user consent mechanisms.

## **REFERENCES**

- 1) PyTorch Documentation: Official documentation for the PyTorch library, accessed for guidance on building deep learning models.
- 2) Scikit-learn Documentation: Official documentation for the scikit-learn library, used for implementing machine learning algorithms and preprocessing techniques.
- 3) Pandas Documentation: Official documentation for the Pandas library, utilized for data manipulation and analysis.
- 4) NumPy Documentation: Official documentation for the NumPy library, used for mathematical operations and array manipulation.
- 5) Streamlit Documentation: Official documentation for Streamlit, referred to for building interactive web applications.
- 6) Bootstrap Documentation: Official documentation for Bootstrap, a popular CSS framework used for designing responsive web pages.
- 7) Python Requests Documentation: Official documentation for the Python Requests library, used for making HTTP requests to fetch data from external APIs.
- 8) Visual Studio Code Documentation: Official documentation for Visual Studio Code, used as the primary integrated development environment (IDE) for the project.

- 9) Adobe Illustrator: Adobe Illustrator software used for designing the project logo and required thumbnails.
- 10) GitHub: GitHub platform used for version control, collaboration, and cloud backup of the project.
- 11) Reddit: Reddit online social community, consulted the Django community for suggestions and assistance during the project.
- 12) Stack Overflow: Stack Overflow online community, referred to for engineering Q&A and getting help during the development process.
- 13) Entertainment news - TV, movies, music, books, theater, art (2018) Vulture. Available at: <http://www.vulture.com/> (Accessed: January 20, 2023).
- 14) Analysis of Movie Recommendation Systems; with and without considering the low rated movies-IEEE xplore. (n.d.). Retrieved January 20, 2023, from <https://ieeexplore.ieee.org/document/9077803>

# PLAGARISM REPORT

## Project RReport

### ORIGINALITY REPORT

<b>21</b> %	<b>20</b> %	<b>2</b> %	<b>16</b> %
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

### PRIMARY SOURCES

<b>1</b>	<b>dspace.daffodilvarsity.edu.bd:8080</b> Internet Source	<b>10</b> %
<b>2</b>	<b>Submitted to Daffodil International University</b> Student Paper	<b>7</b> %
<b>3</b>	<b>fdocuments.us</b> Internet Source	<b>&lt;1</b> %
<b>4</b>	<b>Submitted to University of Stellenbosch, South Africa</b> Student Paper	<b>&lt;1</b> %
<b>5</b>	<b>Submitted to Westminster International University in Tashkent</b> Student Paper	<b>&lt;1</b> %
<b>6</b>	<b>Submitted to St. Petersburg High School</b> Student Paper	<b>&lt;1</b> %
<b>7</b>	<b>Submitted to Staffordshire University</b> Student Paper	<b>&lt;1</b> %
<b>8</b>	<b>Carlos Rojas. "Building Progressive Web Applications with Vue.js", Springer Science and Business Media LLC, 2020</b> Publication	<b>&lt;1</b> %