*A Mini Project Report on*

# Movie Recommendation System

**T.E. - I.T Engineering**

**Submitted By**

Kevin Thakkar (20104023)
Vinayak Somvanshi (20104003)
Yogesh Kumbhar (20104139)

Under The Guidance Of

**Prof. Manjusha Kashilkar**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

A.P.SHAH INSTITUTE OF TECHNOLOGY

G.B. Road, Kasarvadavali, Thane (W), Mumbai-400615

UNIVERSITY OF MUMBAI

**Academic year: 2022-23**

# CERTIFICATE

This to certify that the Mini Project report on **Movie Recommendation System** has been submitted by Kevin Thakkar (20104023), Vinayak Somvanshi (20104003), and Yogesh Kumbhar (20104139) who are the students of A. P. Shah Institute of Technology, Thane, Mumbai, as a partial fulfilment of the requirement for the degree in **Information Technology**, during the academic year 2022-2023 in the satisfactory manner as per the curriculum laid down by University of Mumbai.

**Prof. Manjusha Kashilkar**
**Guide**

**Dr. Kiran Deshpande**
**Head of Department of Information Technology**

**Dr. Uttam D. Kolekar**
**Prinicipal**

**Examiner(s):**

**1.**

**2.**

**Place**: A.P Shah Institute of Technology, Thane

Date:

# ACKNOWLEDGEMENT

This project would not have come to fruition without the invaluable help of our guide **Prof. Manjusha Kashilkar**. Expressing gratitude towards our HOD, **Dr. Kiran Deshpande**, and the Department of Information Technology for providing us with the opportunity as well as the support required to pursue this project. We would also like to thank our teacher **Prof. Yaminee Patil** who gave us her valuable suggestions and ideas when we were in need of them. We would also like to thank our peers for their helpful suggestions.

# ABSTRACT

In this hustling world, entertainment is a necessity for each one of us to refresh our mood and energy. Entertainment regains our confidence for work and we can work more enthusiastically. For revitalizing ourselves, we can listen to our preferred music or can watch movies of our choice. For searching favorable movies online, we can utilize movie recommendation systems, which are more reliable, since searching for preferred movies will require more and more time which one cannot afford to waste. In this project, movies are recommended to user based on the content they've typed into the search bar and desired list of movies are shown as output. This system relies on API, through which it will be able to recommend a large variety of movies. Since this system uses API, it also helps to keep track of the most recently released movies. The system uses the filter search feature to improve the user experience. Providing different filter options to the users can help them in finding a specific movie to watch. This system has a login/registration feature, which allows registered users to add films to their watchlist so they can keep track of the films they want to watch. The movies will also provide users with additional information like date of release, synopsis, cast, rating, and a frame to play trailer of the same.

# TABLE OF CONTENTS

# Chapter 1

## Introduction:

Movies are a part and parcel of life. There are different types of movies like some for entertainment, some for educational purposes, some are animated movies for children, and some are horror movies or action films. Movies can be easily differentiated through their genres like comedy, thriller, animation, action etc. Other way to distinguish among movies can be either by releasing year, language, director etc. Watching movies online, there are a number of movies to search in our most liked movies. Movie Recommendation Systems helps us to search our preferred movies among all of these different types of movies and hence reduce the trouble of spending a lot of time searching our favorable movies. So, it requires that the movie recommendation system should be very reliable and should provide us with the recommendation of movies which are exactly same or most matched with our preferences.

## 1.1 Purpose:

A recommendation system or recommendation engine is a model used for information filtering where it tries to predict the preferences of a user and provide suggests based on these preferences. These systems have become increasingly popular nowadays and are widely used today in areas such as movies, music, books, videos, clothing, restaurants, food, places and other utilities. These systems collect information about a user's preferences and behavior, and then use this information to improve their suggestions in the future.

A large number of companies are making use of recommendation systems to increase user interaction and enrich a user's shopping experience. Recommendation systems have several benefits, the most important being customer satisfaction and revenue. Movie Recommendation system is very powerful and important system. But, due to the problems associated with pure collaborative approach, movie recommendation systems also suffers with poor recommendation quality and scalability issues.

## 1.2 Problem Statement:

For building a recommender system from scratch, we face several different problems. Currently there are a lot of recommender systems based on the user information, so what should we do if the website has not gotten enough users. After that, we will solve the representation of a movie, which is how a system can understand a movie. That is the precondition for comparing similarity between two movies. Movie features such as genre, actor and director are a way that can categorize movies. But for each feature of the movie, there should be different weight for them and each of them plays a different role for recommendation.

- The goal of the project is to recommend a movie to the user.
- Providing related content out of relevant and irrelevant collection of items to users of online service providers.
- What kind of movie features can be used for the recommender system.
- How to recommend movies when there is no user information.

## 1.3 Objectives:

- Improving the Accuracy of the recommendation system
- Improve the Quality of the movie Recommendation system
- Improving the Scalability.
- Enhancing the user experience.

## 1.4 Scope:

The objective of this project is to provide accurate movie recommendations to users. The goal of the project is to improve the quality of movie recommendation system, such as accuracy, quality and scalability of system than the pure approaches. This is done using approach by combining content based filtering and collaborative filtering, to eradicate the overload of the data, recommendation system is used as information filtering tool in social networking sites. Hence, there is a huge scope of exploration in this field for improving scalability, accuracy and quality of movie recommendation systems Movie Recommendation system is very powerful and important system. But, due to the problems associated with pure collaborative approach, movie recommendation systems also suffer with poor recommendation quality and scalability issues.

# Chapter 2

## Literature Review:

- In [Paper 1] (2019), it was proposed that a movie recommender system can be built using the K-Means Clustering and K-Nearest Neighbor algorithms. The proposed work deals with the introduction of various concepts related to artificial intelligence machine learning and recommendation system. In this work, various tools and techniques have been used to build recommender systems. Finally, in this work for different cluster values, different values of Root Mean Squared Error are obtained. In this proposed work as the no of clusters decreases, the value of RMSE also decreases. The best value of RMSE obtained is 1.081648. (**DOI:** 10.1109/CONFLUENCE.2019.8776969)

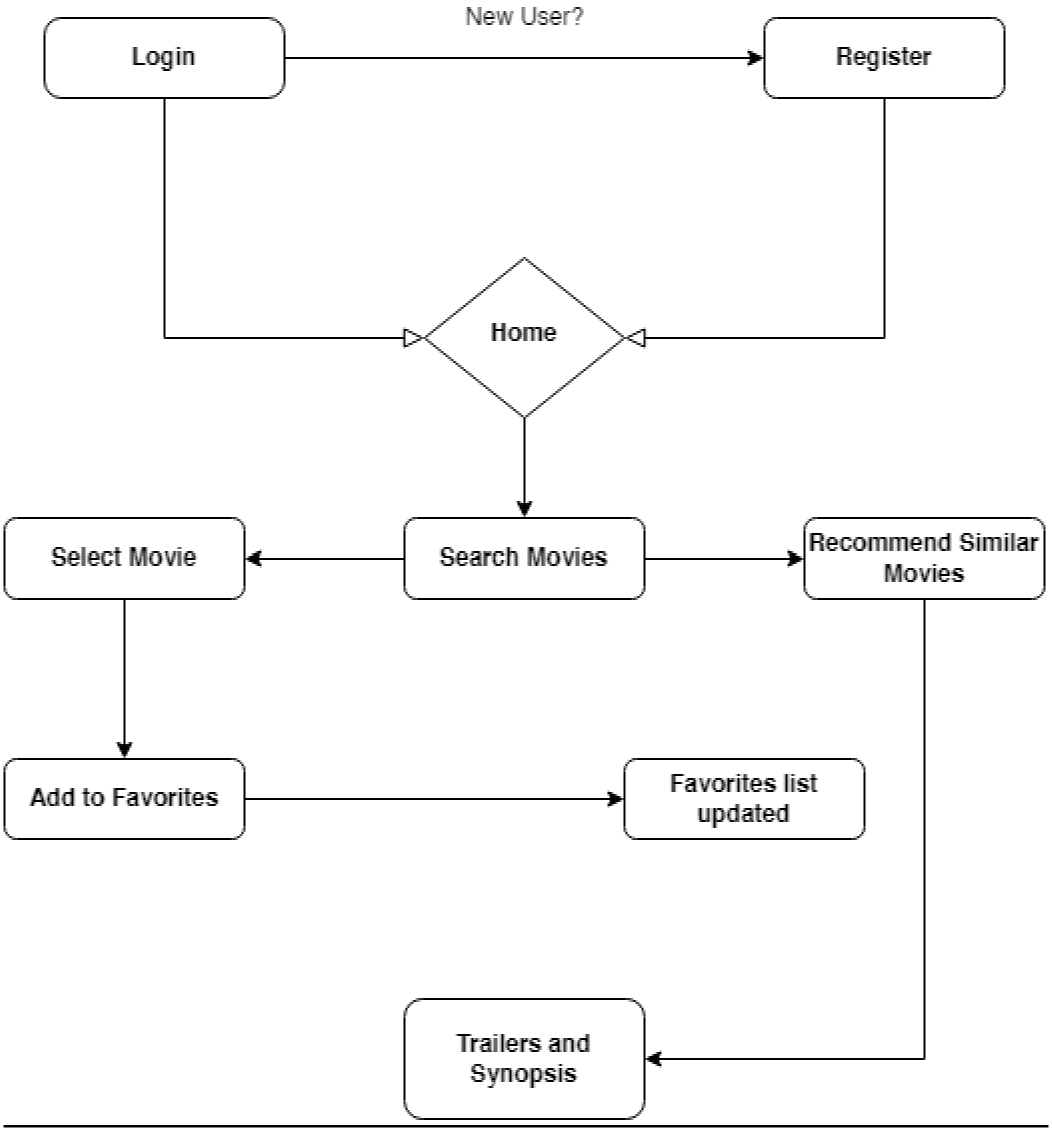
- In [Paper 2] (2020), it proposed that although varieties of methods are used to implement a recommendation system, Content-based filtering is the simplest method. Which takes input from the users, rechecks his/her history/past behavior, and recommends a list of similar movies. In this paper, to prove the effectiveness, K-NN algorithms and collaborative filtering are used to mainly focus on enhancing the accuracy of results as compared to content-based filtering. This approach is based on cosine similarity using k-nearest neighbor with the help of a collaborative filtering technique, at the same time removing the drawbacks of the content-based filtering. Although using Euclidean distance is preferred, cosine similarity is used as the accuracy of cosine angle and the equidistance of movies remain almost the same. (**DOI:** 10.1109/ICESC48915.2020.9155879)

- In [Paper 3] (2015), it proposed a novel recommendation algorithm called STPMF based on neighborhood model and matrix factorization model, where complementary roles of similarity relationships and trust relationships to the user model by means of a weight w are considered simultaneously. This algorithm was proposed because traditional collaborative filtering approaches are often confronted with two major problems: data sparsity and cold-start. Along with the rise of social media, social network is producing a large and rich set of social data, which provides a new way to solve the problems of collaborative filtering, namely, can make use of social data to enhance the recommendation accuracy. However, traditional recommendation algorithms may only consider either the influence of similarity relationships or trust relationships to the user model but fail to take full advantage of the implications of social data. Compared with existing recommendation algorithms, the proposed method can effectively alleviate the problems of collaborative filtering and enhance the recommendation accuracy. (**DOI:** 10.1109/CIC.2015.11)

# Chapter 3

## Proposed System:

The aim of the project is to develop an application for recommending users a movie as per their requirement. The proposed system can overcome all the limitations and errors of the manually selecting a particular movie to watch.



## 3.1 Features and Functionalities:

1. **Sign in/ Sign up/ Forget password:**

   User can be able to login the system and if the user is new, he/she should signup first with credentials first. If the user forgets password his/her password they can reset it with their username and password. This provides the information of the user and acts as a security to the data. The user can register and use the software.

2. **Display latest and trending movies on the application's main page:**

   The user can choose the required movies which are available on the system's main page. The movies vary from latest and trending and all in between. This will ensure that the user could choose a movie easily without hassling through a humongous list of content.

3. **Allow the user to see the detailed description of a particular movie:**

   User has to know what exactly they are watching and this is fulfilled by the detailed description of movie. Along with the information about the movies user can also note the cast and crew along with the plot and age ratings of the movie.

4. **Trailers:**

   Once the user has found their ideal movie, they should be able to watch a preview or moreover the trailer for the particular movie. User can simply click on the movie and the trailers will be displayed on the screen. After the desired trailer is watched the user can decide for himself what to watch.

5. **Like and comment:**

   The user will be able to use the like button for his/her favourite movies. Also the user can write comments for any movie so that it can be viewed by other users that might interest them. Comments section can help other users to get a better idea about the specific movie.

6. **Watchlist:**

   Watchlist helps user in keeping track of movies they want to watch in future. The user will be able to add movies to their watchlist and remove as they wish.

# Chapter 4

## Requirement Analysis:

- **Performance Requirements:**

The application should be responsive and recommend movies based on searched keyword.

- **Design Constraints:**

The application should be able to run on any Pc or Laptop.

- **Availability:**

The application should be always available whenever user wants to use.

## Hardware requirements:

- **RAM:**

The application requires a device with a minimum of 512MB RAM while running.

- **Processor speed:**

The application requires a device with a minimum processor speed of 1GHz while running.

## Software requirements:

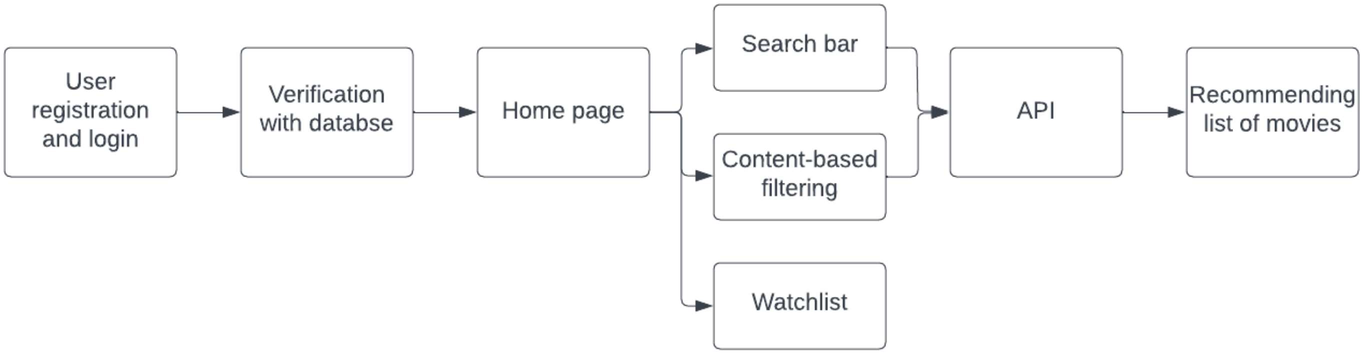
- **Operating system:**

The application must run on any Operation System.

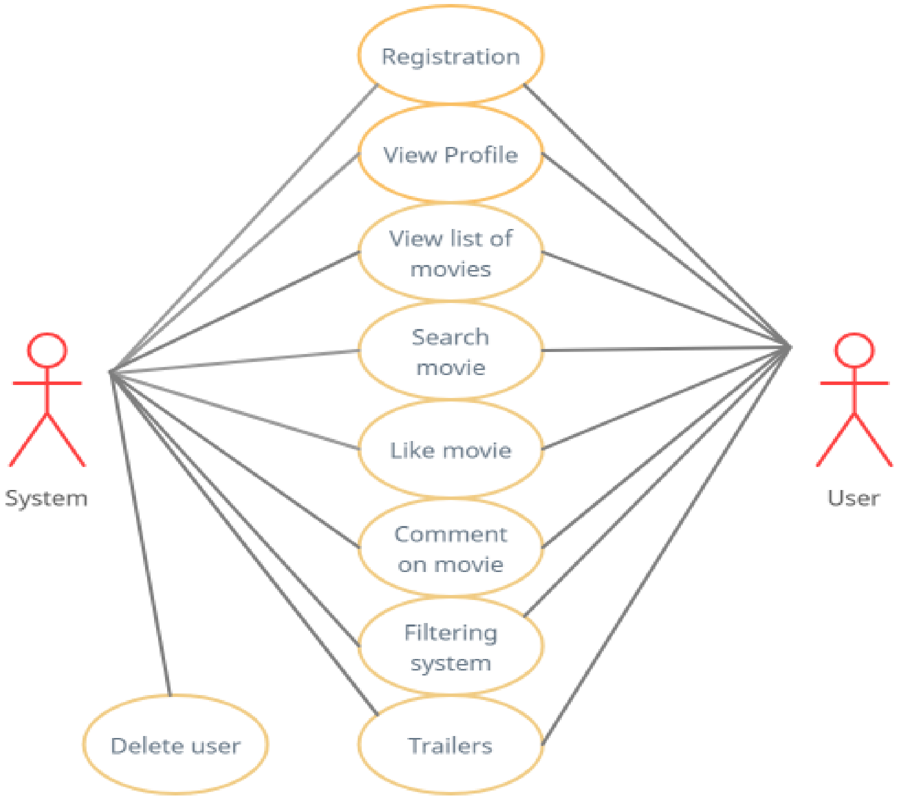
# Chapter 5

## Project Design:

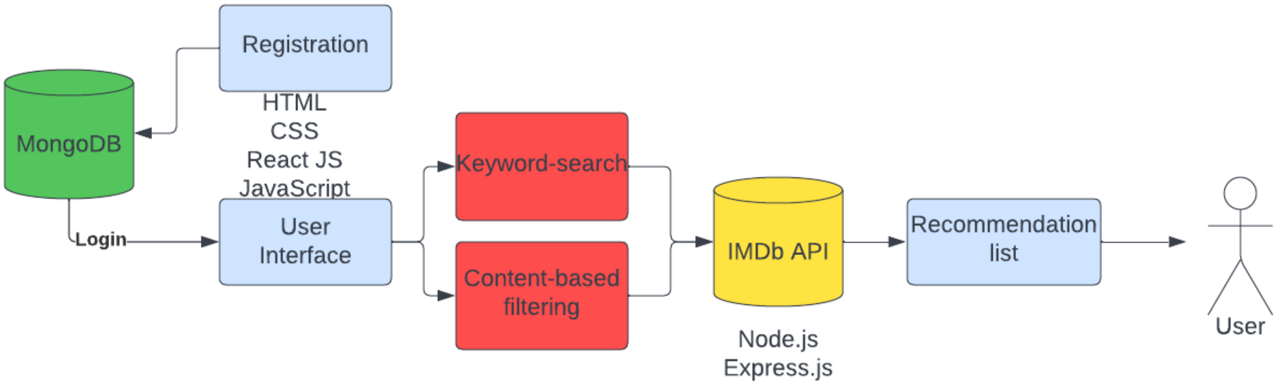
### 5.1 Data Flow Diagram (DFD):


User registration and login
Verification with databse
Home page
Search bar
Content-based filtering
Watchlist
API
Recommending list of movies

### 5.2 Use Case Diagram:



### 5.3 System Architecture:



# Chapter 6

## Technical Specifications:

- **Frontend:**

- **HTML**

HTML (Hypertext Markup Language) – It's a markup language for creating and displaying electronic documents (web pages). They are the backbone in the organization and placement of content on a web page.

- **CSS**

CSS (Cascading Style Sheets) – It is responsible for the format and layout of the web pages. It includes the font styles, sizes, layout, color, and other aspects of a web page. Developers commonly use frameworks like SASS and LESS to make CSS more manageable and more dynamic.

- **React JS**

ReactJS tutorial provides basic and advanced concepts of ReactJS. Currently, ReactJS is one of the most popular JavaScript front-end libraries which has a strong foundation and a large community. ReactJS is a declarative, efficient, and flexible JavaScript library for building reusable UI components. It is an open-source, component-based front-end library which is responsible only for the view layer of the application.

- **JavaScript**

JavaScript is a light-weight object-oriented programming language which is used by several websites for scripting the webpages. It is an interpreted, full-fledged programming language that enables dynamic interactivity on websites when applied to an HTML document. It was introduced in the year 1995 for adding programs to the webpages in the Netscape Navigator browser. Since then, it has been adopted by all other graphical web browsers. With JavaScript, users can build modern web applications to interact directly without reloading the page every time. The traditional website uses JavaScript to provide several forms of interactivity and simplicity.

## **Backend:**

- **MongoDB**

MongoDB is a source-available cross-platform document-oriented database program. Classified as a NoSQL database program, MongoDB uses JSON-like documents with optional schemas. MongoDB is a non-relational document database that provides support for JSON-like storage. The MongoDB database has a flexible data model that enables you to store unstructured data, and it provides full indexing support, and replication with rich and intuitive APIs.

- **Node.js**

Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on a JavaScript Engine and executes JavaScript code outside a web browser, which was designed to build scalable network applications. Node.js is primarily used for non-blocking, event-driven servers, due to its single-threaded nature. It's used for traditional web sites and back-end API services, but was designed with real-time, push-based architectures in mind.

- **Express.js**

Express.js, or simply Express, is a back-end web application framework for building RESTful APIs with Node.js, released as free and open-source software under the MIT License. It is designed for building web applications and APIs. It has been called the de facto standard server framework for Node.js.

## **API:**

The IMDb-API is a web service for receiving movie, serial and cast information. APIs results is a JSON and includes items such as movie specifications, images, posters, trailers, ratings, Wikipedia page content and more.

# Chapter 7

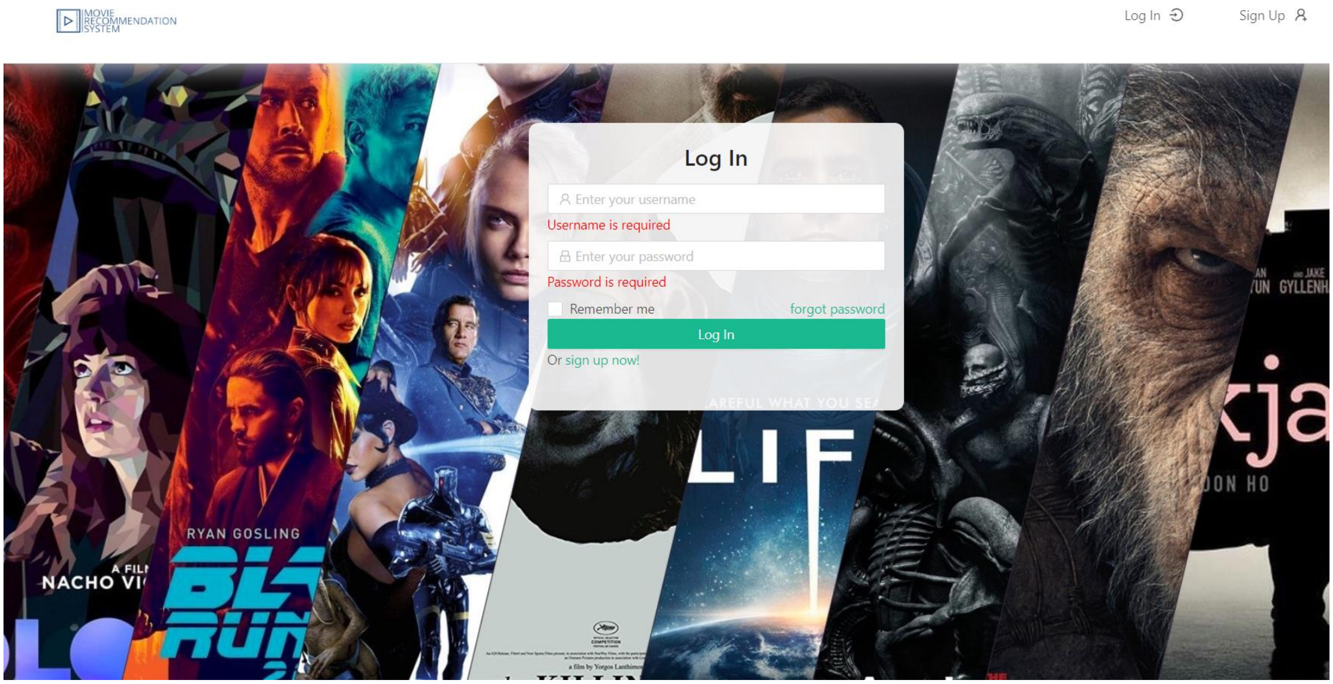
**Project Scheduling:**

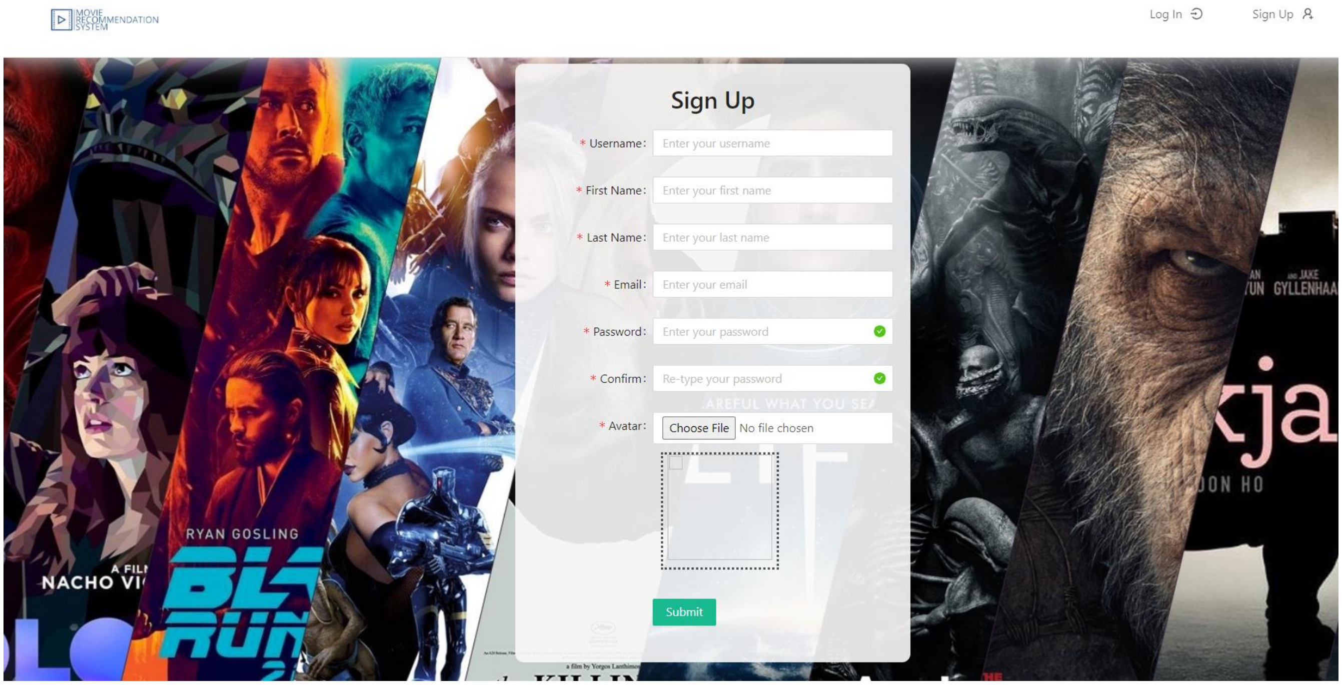
| Sr No. | Group Member | Time Duration | Work to be done |
|---|---|---|---|
| **1** | Yogesh Kumbhar | 1st Week of September | Login/Sign-up page and database connection |
| **2** | Vinayak Somvanshi | 3rd Week of September | Home page, Search bar and Filter option |
| **3** | Kevin Thakkar | 2nd Week of October | Trailer video embedding, watchlist and like/comment option |

# Chapter 8

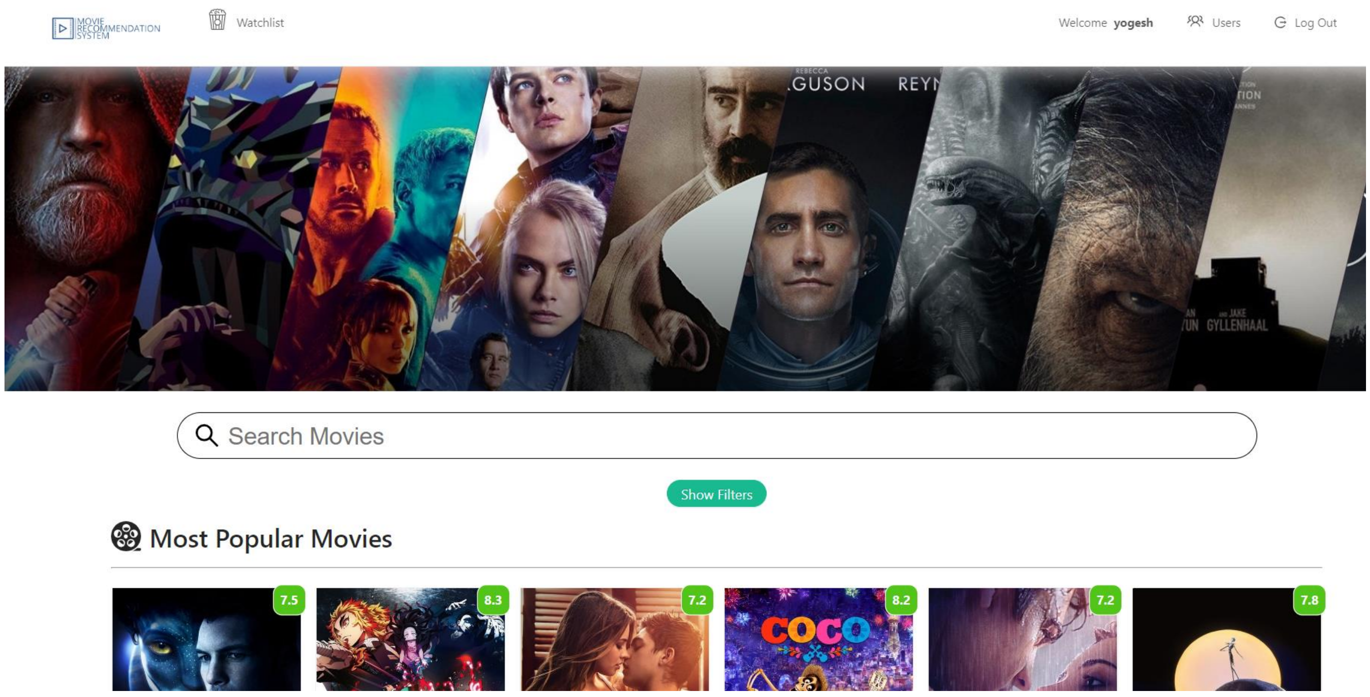
## Implementation:

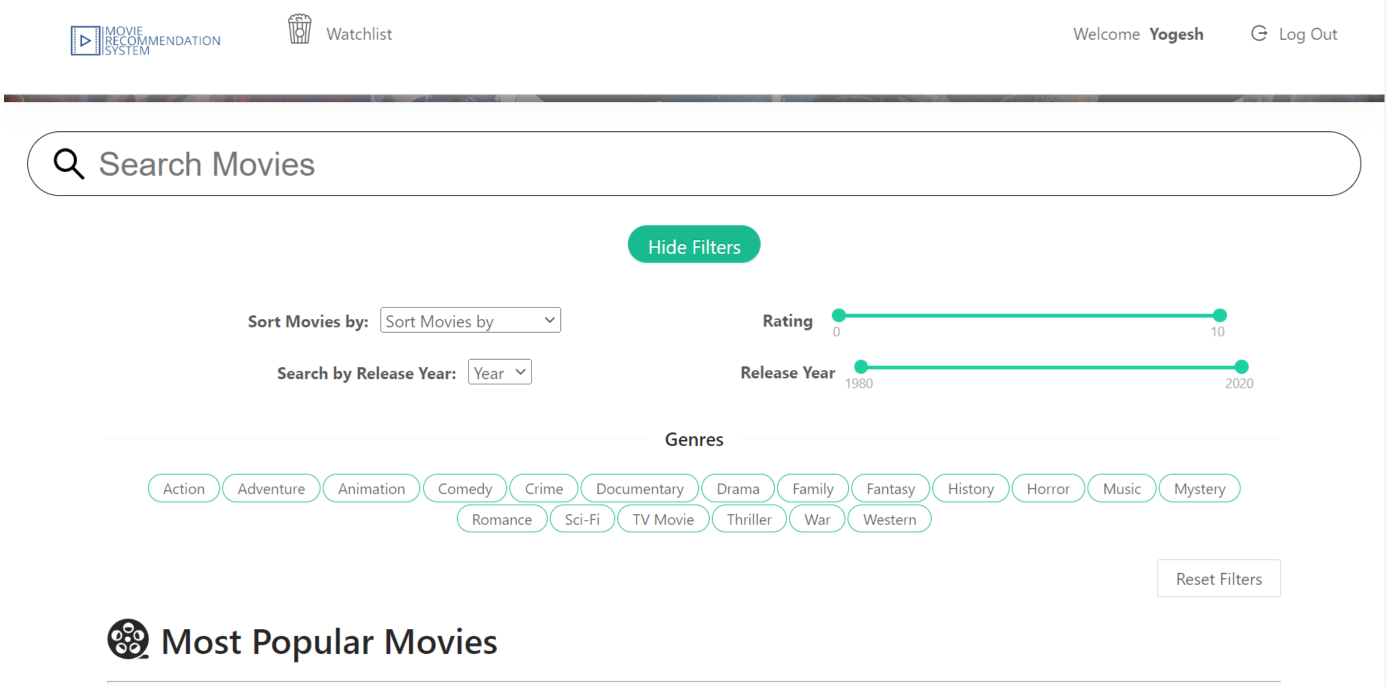
## Login/Sign-up Page:



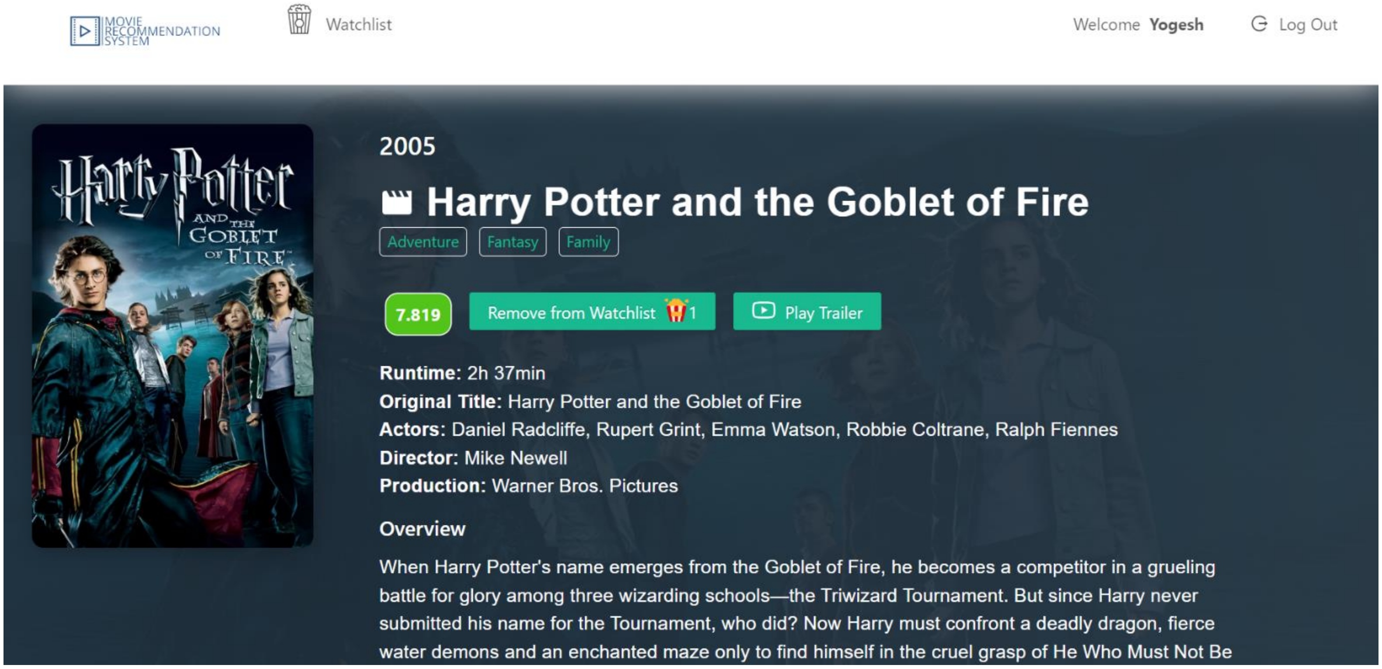


# Home page:


MOVIE RECOMMENDATION SYSTEM
Watchlist
Welcome yogesh
Users
Log Out
Search Movies
Show Filters
Most Popular Movies
7.5
8.3
7.2
8.2
7.2
7.8


MOVIE RECOMMENDATION SYSTEM
Watchlist
Welcome Yogesh
Log Out
Search Movies
Hide Filters
Sort Movies by:
Sort Movies by
Rating
0
10
Search by Release Year:
Year
Release Year
1980
2020
Genres
Action
Adventure
Animation
Comedy
Crime
Documentary
Drama
Family
Fantasy
History
Horror
Music
Mystery
Romance
Sci-Fi
TV Movie
Thriller
War
Western
Reset Filters
Most Popular Movies

## Search results:



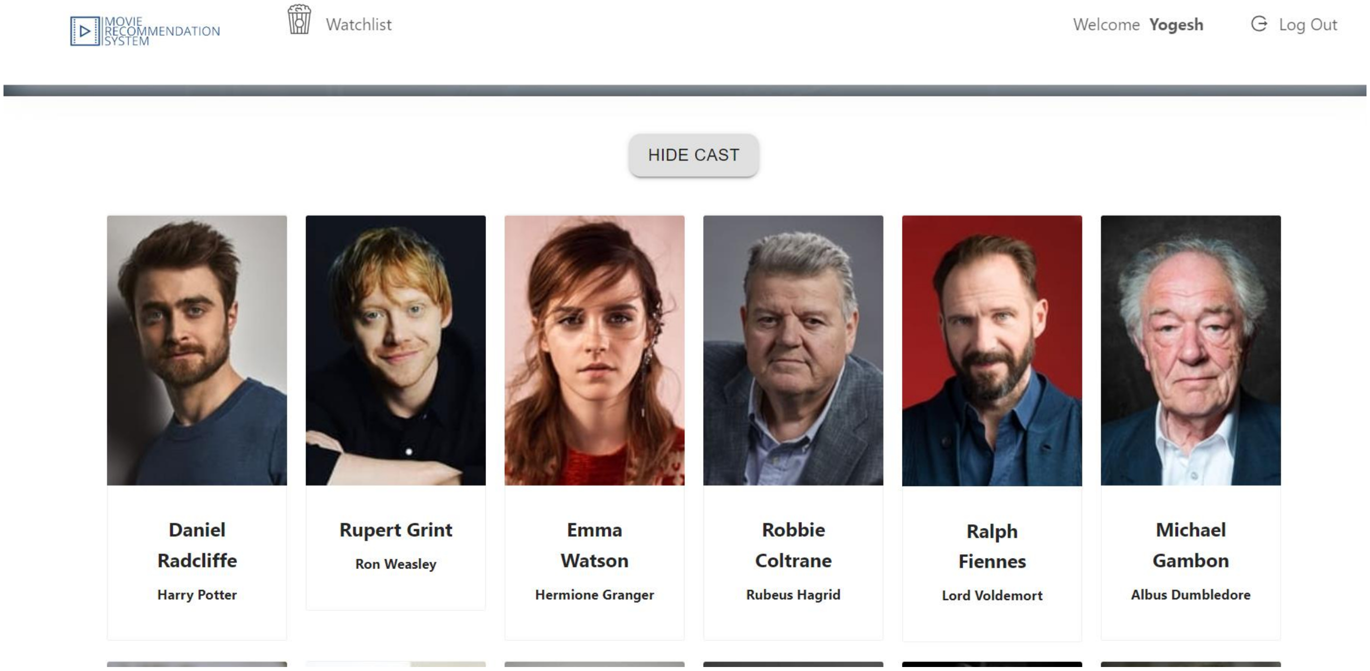
2005

Harry Potter and the Goblet of Fire

**Runtime:** 2h 37min
**Original Title:** Harry Potter and the Goblet of Fire
**Actors:** Daniel Radcliffe, Rupert Grint, Emma Watson, Robbie Coltrane, Ralph Fiennes
**Director:** Mike Newell
**Production:** Warner Bros. Pictures

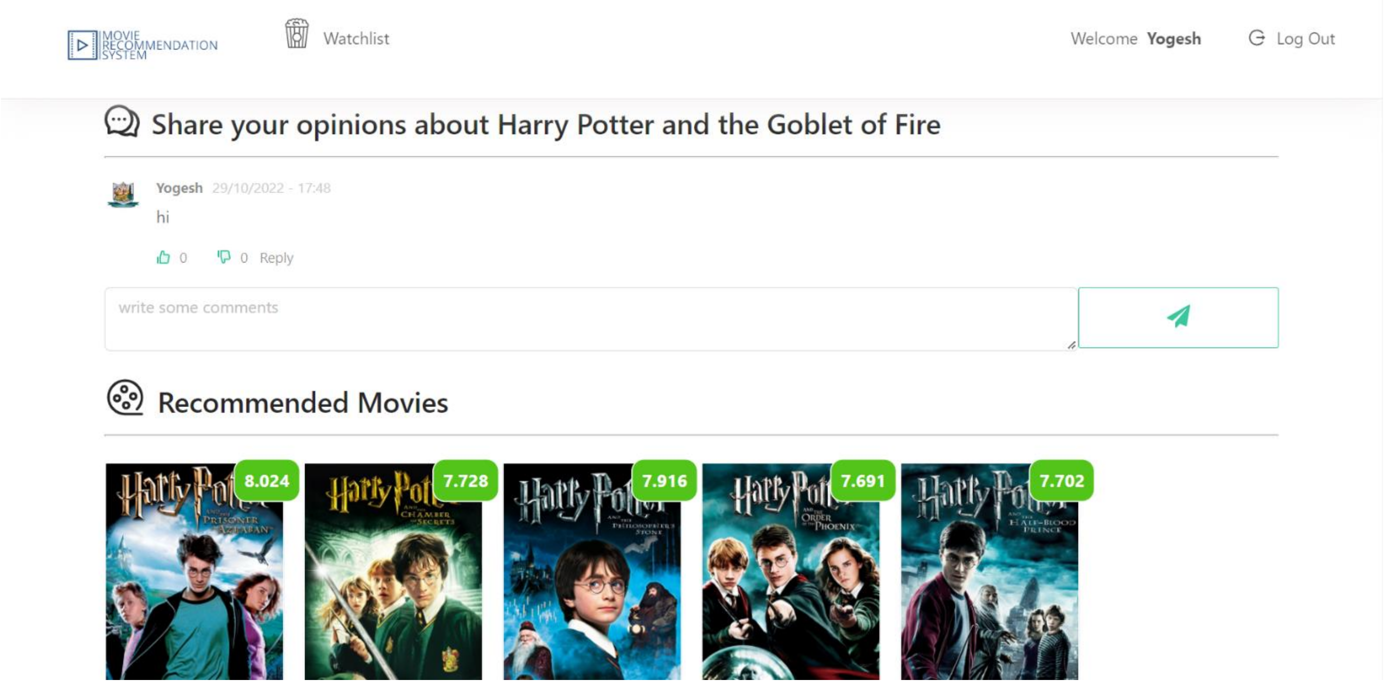
**Overview**

When Harry Potter's name emerges from the Goblet of Fire, he becomes a competitor in a grueling battle for glory among three wizarding schools—the Triwizard Tournament. But since Harry never submitted his name for the Tournament, who did? Now Harry must confront a deadly dragon, fierce water demons and an enchanted maze only to find himself in the cruel grasp of He Who Must Not Be



| | |
|---|---|
| Daniel Radcliffe | Harry Potter |
| Rupert Grint | Ron Weasley |
| Emma Watson | Hermione Granger |
| Robbie Coltrane | Rubeus Hagrid |
| Ralph Fiennes | Lord Voldemort |
| Michael Gambon | Albus Dumbledore |

## Recommendations:



Share your opinions about Harry Potter and the Goblet of Fire

Yogesh 29/10/2022 - 17:48

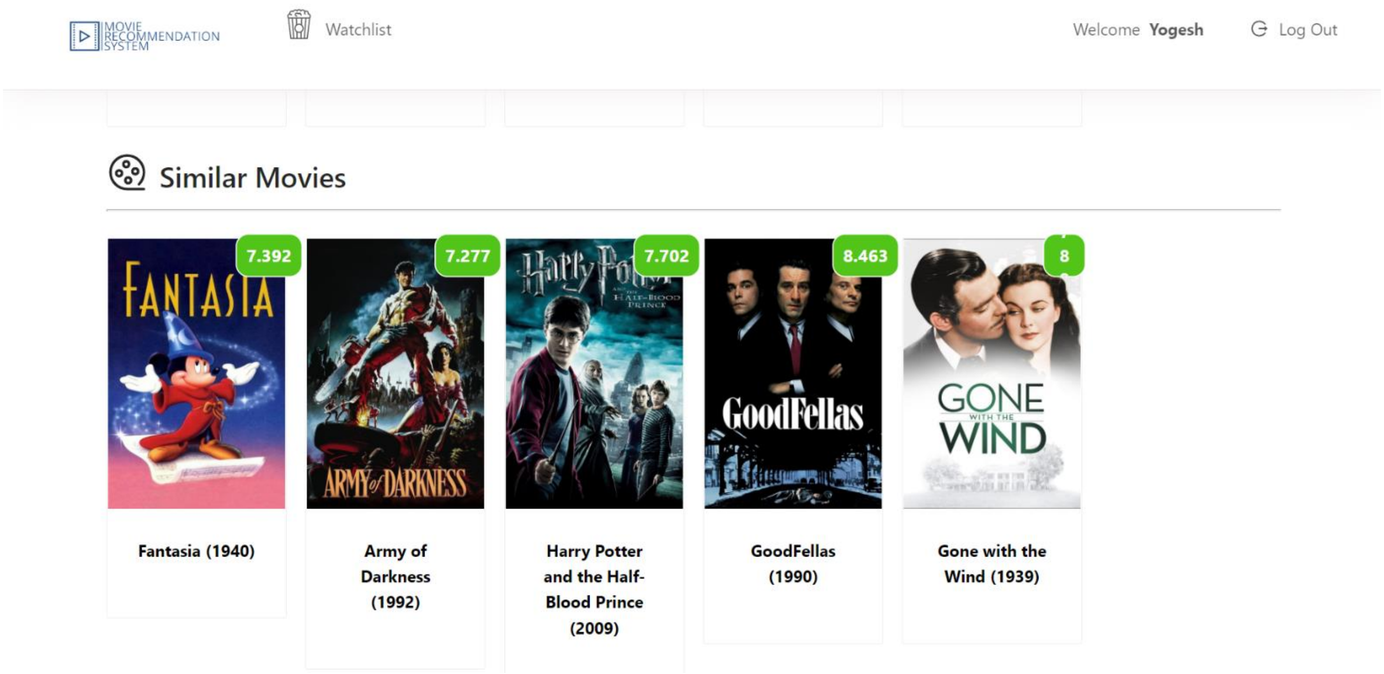
hi

0 0 Reply

write some comments

Recommended Movies

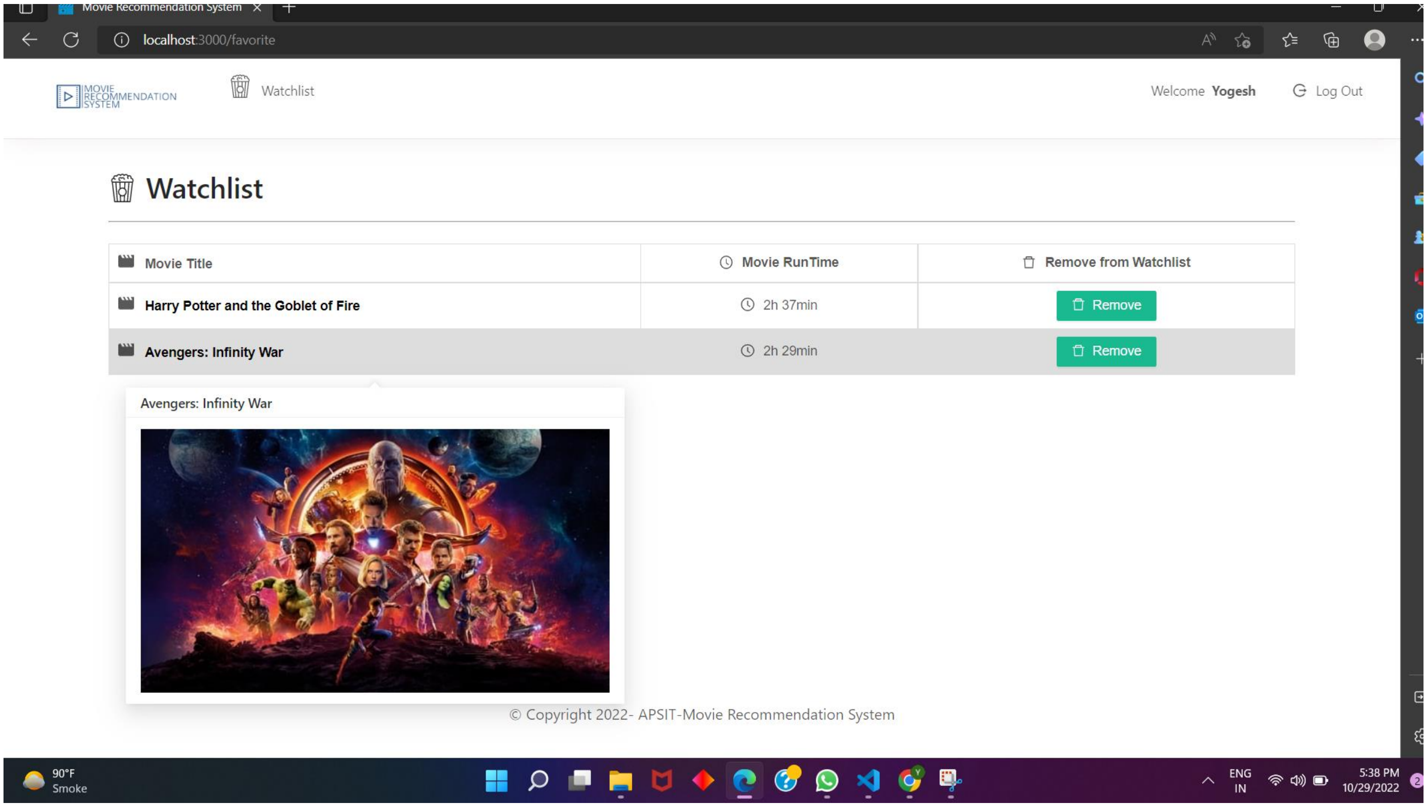
8.024 7.728 7.916 7.691 7.702



Similar Movies

| 7.392 | 7.277 | 7.702 | 8.463 | 8 |
|---|---|---|---|---|
| Fantasia (1940) | Army of Darkness (1992) | Harry Potter and the Half-Blood Prince (2009) | GoodFellas (1990) | Gone with the Wind (1939) |

# Watchlist:



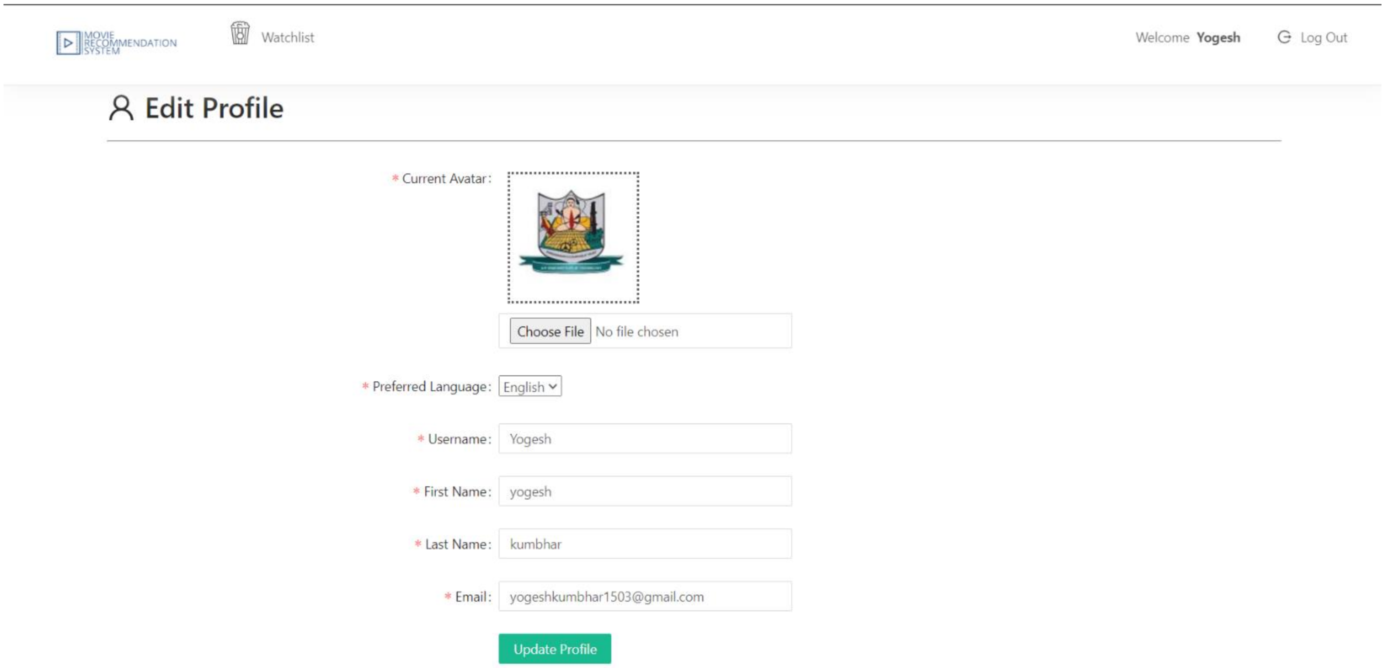
## Watchlist

| Movie Title | Movie RunTime | Remove from Watchlist |
|---|---|---|
| Harry Potter and the Goblet of Fire | 2h 37min | Remove |
| Avengers: Infinity War | 2h 29min | Remove |

Avengers: Infinity War

© Copyright 2022- APSIT-Movie Recommendation System

# User profile:



## Edit Profile

* Current Avatar:

Choose File No file chosen

* Preferred Language: English

* Username: Yogesh

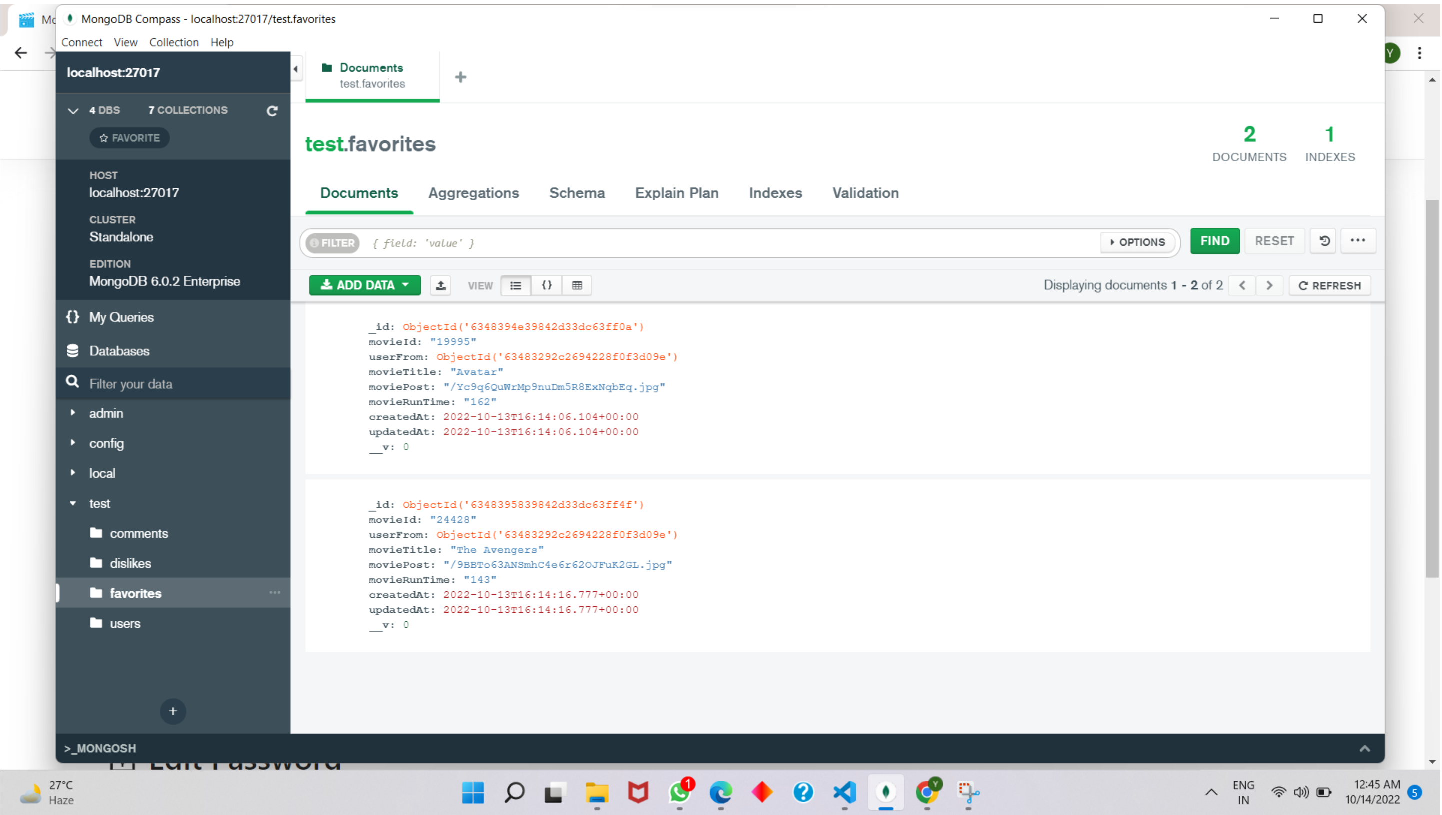
* First Name: yogesh

* Last Name: kumbhar

* Email: yogeshkumbhar1503@gmail.com

Update Profile

# Database containing information about the registered users:



MongoDB Compass - localhost:27017/test.favorites

HOST
localhost:27017

CLUSTER
Standalone

EDITION
MongoDB 6.0.2 Enterprise

test.favorites

2 DOCUMENTS 1 INDEXES

Displaying documents 1 - 2 of 2

```
_id: ObjectId('6348394e39842d33dc63ff0a')
movieId: "19995"
userFrom: ObjectId('63483292c2694228f0f3d09e')
movieTitle: "Avatar"
moviePost: "/Yc9q6QuWrMp9nuDm5R8ExNqbEq.jpg"
movieRunTime: "162"
createdAt: 2022-10-13T16:14:06.104+00:00
updatedAt: 2022-10-13T16:14:06.104+00:00
__v: 0
```

```
_id: ObjectId('6348395839842d33dc63ff4f')
movieId: "24428"
userFrom: ObjectId('63483292c2694228f0f3d09e')
movieTitle: "The Avengers"
moviePost: "/9BBTo63ANSmhC4e6r62OJFuK2GL.jpg"
movieRunTime: "143"
createdAt: 2022-10-13T16:14:16.777+00:00
updatedAt: 2022-10-13T16:14:16.777+00:00
__v: 0
```

# Chapter 9

## Result and Discussion:

Since our project is movie recommendation system. One can develop a movie recommendation system by using either content based or collaborative filtering or combining both. In our project we have used content based filtering. Both the approaches have advantages and dis-advantages in content-based filtering, it is based on the user ratings or user likes only such kind of movie will recommended to the user.

The model can only make recommendations based on existing interests of the user. In other words, the model has limited ability to expand on the users' existing interests. In Collaborative filtering the recommendation is comparison of similar users.

No need of domain knowledge because the embeddings are automatically learned. The model can help users discover new interests. In case of ML based system, it may not know the user is interested in a given item, but the model might still recommend it because similar users are interested in that item.

The prediction of the model for a given (user, item) pair is the dot product of the corresponding embeddings. So, if an item is not seen during training, the system can't create an embedding for it and can't query the model with this item. This issue is often called the cold-start problem. The hybrid approach will resolve all these limitations by combining both content and collaborative filtering.

# Chapter 10

## Conclusion and future scope:

In this project, to improve the accuracy, quality and scalability of movie recommendation system, content-based filtering is implemented. The proposed system works on user input to facilitate the recommendations. Content filtering helps in curating better movies to recommend to the user. Posting of comments can help other users to get a better review of the movie they might be interested in. In the implemented system, filtering options such as rating of movie, year of release and genre are our provided to the user.

In the proposed approach, it has considered genres of movies but, in future we can also consider age of user as according to the age movie preferences also changes, like for example, during our childhood we like animated movies more as compared to other movies. There is a need to work on the memory requirements of the proposed approach in the future. The proposed approach has been implemented here on movie API only. It can also be implemented on the datasets and the performance can be computed in the future.

# Chapter 11

## References:

**[1]** Movie Recommender System Using K-Means Clustering AND K-Nearest Neighbor | DOI: 10.1109/CONFLUENCE.2019.8776969 | 10-11 January 2019.

**[2]** Movie Recommender System Using Collaborative Filtering | DOI: 10.1109/ICESC48915.2020.9155879 | 02-04 July 2020.

**[3]** Probabilistic Matrix Factorization Based on Similarity Propagation and Trust Propagation for Recommendation | DOI: 10.1109/CIC.2015.11 | 27-30 October 2015.

**[4]** A Movie Recommender System: MOVREC | International Journal of Computer Applications Volume 124 – No.3 | August 2015 | DOI:10.5120/ijca2015904111

**[5]** An Efficient movie recommendation algorithm based on improved k-clique | Human-centric Computing and Information Sciences volume 8, Article number: 38 | December 2018

**[6]** A Comprehensive Survey of Neighborhood-based Recommendation Methods | DOI: 10.1007/978-0-387-85820-3_4 | January 2010

**[7]** Content-Based Movie Recommendation System Using Genre Correlation | DOI: 10.1007/978-981-13-1927-3_42 | November 2018

**[8]** Movie Recommendation System based on users' similarity | IJCSMC, Vol. 3, Issue. 4, Page: 765-770 | April 2014

**[9]** Movie Recommendation System Using Genome Tags and Content-Based Filtering | DOI: 10.1007/978-981-10-8360-0_8 | April 2018

**[10]** Comparing LDA and LSA Topic Models for Content-Based Movie Recommendation Systems | DOI: 10.1007/978-3-319-27030-2_16 | December 2015