MOVIE RECOMMENDATION SYSTEM

UCS503 Software Engineering Project Report

Mid-Semester Evaluation

Submitted by:

101903390 – GAUTAM AGGARWAL

101903392 – PIYUSH BAWA

101903396 – ABHI GARG

101903397 – SAMEER ALAM

BE Third Year, COE

Group No: 3CO15

Submitted to:

: Dr. ASHIMA SINGH

Computer Science and Engineering Department

TIET, Patiala

October 2021

TABLE OF CONTENTS

1 PROJECT SELECTION PHASE

1.1 SOFTWARE BID PROJECT

Team Name: HACK INVERSION
Team ID (will be assigned by Instructor):

Please enter the names of your Preferred Team Members.

- You are required to form **a three to four person** teams
- Choose your team members wisely. You will not be allowed to change teams.

Name	**Roll No**	**Project Experience**	**Programming Language used**
Gautam Aggarwal	101903390	Robotic Arm	Arduino, C++
Piyush Bawa	101903392	Robotic Arm	Arduino, C++
Abhi Garg	101903396	Robotic Arm	Arduino, C++
Sameer Alam	101903397	Web-development, Machine Learning	HTML, CSS, Python

Programming Language / Environment Experience

List the languages you are most comfortable developing in, **as a team**, in your order of preference. Many of the projects involve Java or C/C++ programming.

1. C/C++
2. Python
3. HTML, CSS, Javascript

Choices of Projects:

Please select **4 projects** your team would like to work on, by order of preference: *[Write at-least one paragraph for each choice (motivation, reason for choice, feasibility analysis, etc.)]*

First Choice	Movie Recommendation system: This project will be our first choice as we would like to create a recommendation system that will provide more precise results as compared to existing systems.
Second Choice	Library Management System (Web-based):

	In this project we intend to provide safety against phishing URL's that try to steal our personal details with the help of software integration and data mining techniques
Third Choice	Intelligent rule based Phishing URL detection: In this project we aim to create a web-application to handle all records on a single platform like, overseeing book's issue and return dates, calculate fines etc..
Fourth Choice	Online Code editor: In this project our aim is provide users a platform where they can write their codes online without having to install any code editor in their machine.

Additional Remarks/ Inputs

Please tell us about any other factors that we should take into consideration (e.g., if you really would like to work on a project for some particularly convincing reason).

……

We would like to work on Movie Recommendation System. In this system we will provide

user a very user friendly interface where he would be provided the recommendations based on his search, whether he searches the movie by the name of its Cast or a single alphabet would be enough to provide the movie names to the user.

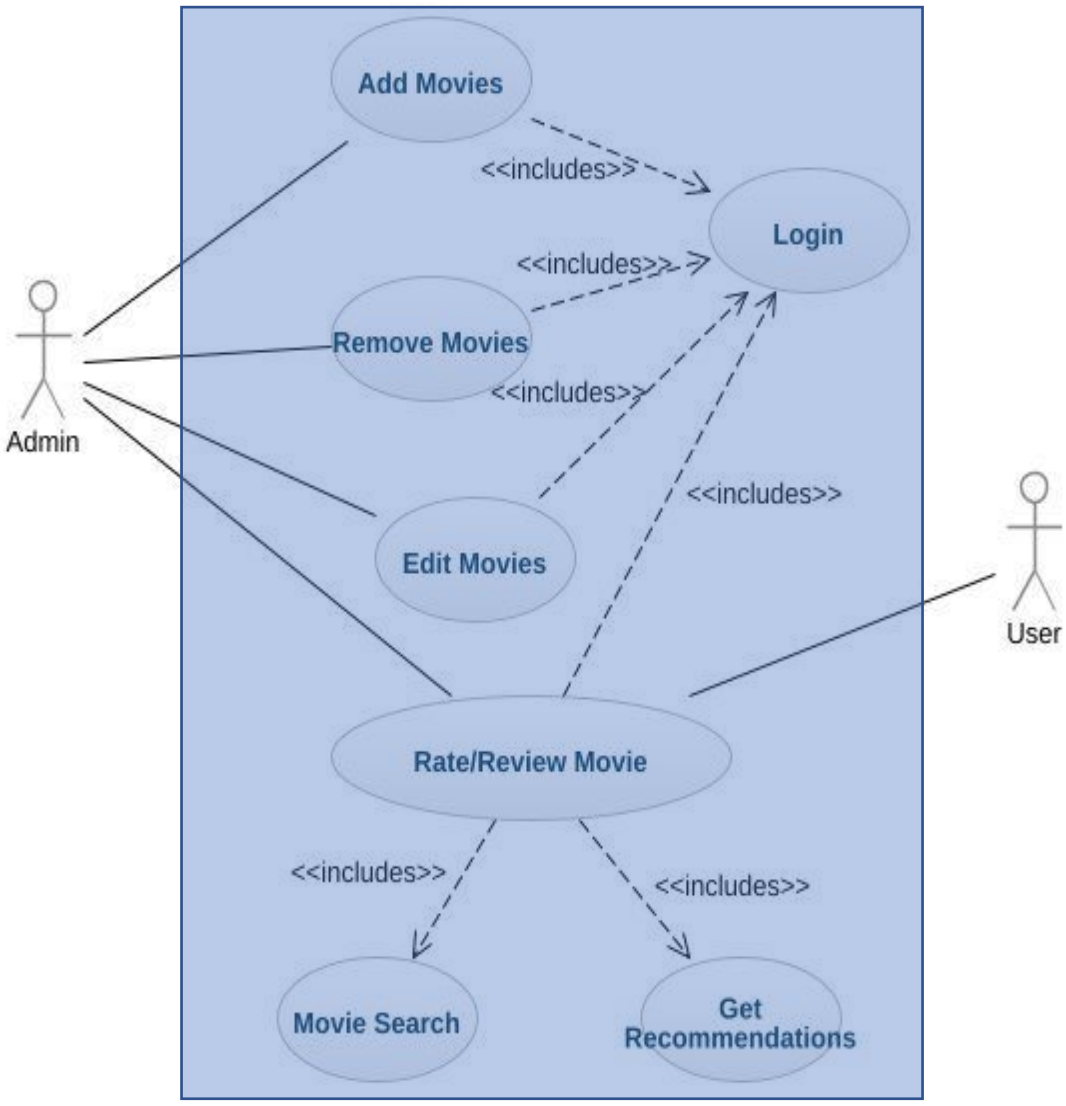
1.2 PROJECT OVERVIEW

The scope of this project is to design and develop a system that is made for easiness of the intended audience so that they can go on to the site and login/signup with their credentials and can get all the kind of movie recommendations under one site. This will help users to select their movie and not waste time searching for different genres over the internet.

2 ANALYSIS PHASE

2.1 USE CASE

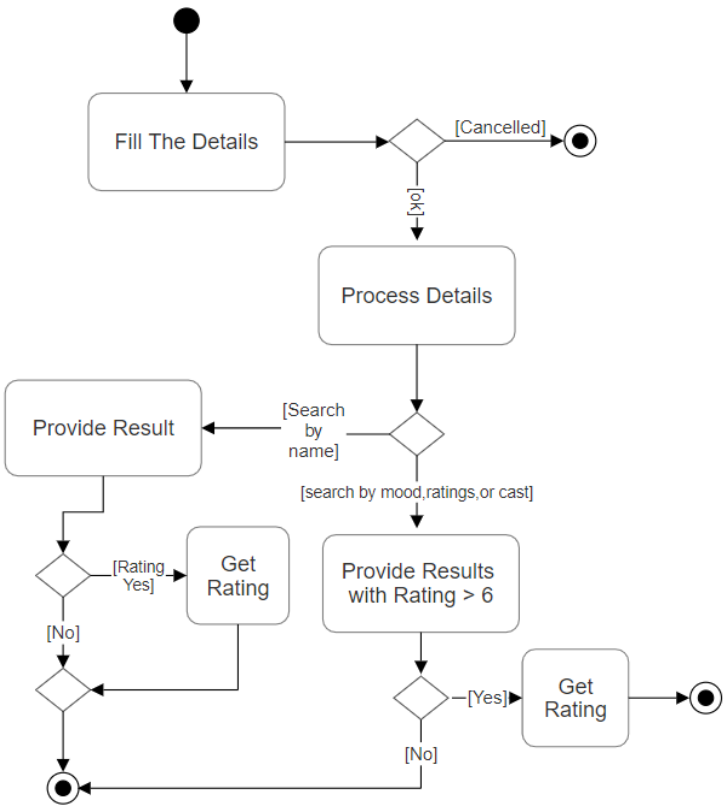
- 2.1.1. USE CASE DIAGRAMS



- 2.1.2. USE CASE TEMPLATES

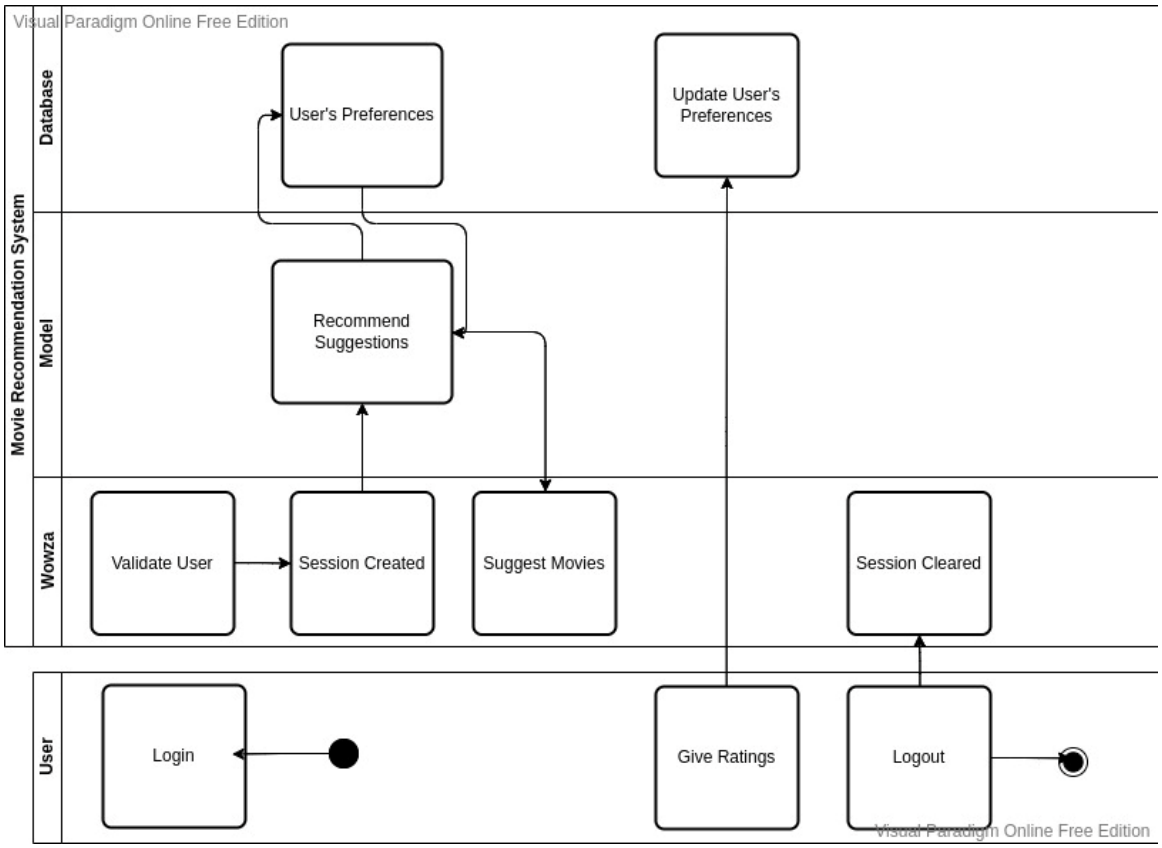
1. **Use Case Title**	Search Movie
2. **Abbreviated Title**	Search Movie
3. **Actors**	Members, Admins
4. **Description** With this search facility, user can use the specified search criteria. For example, Movie name, ratings, and get the desired result.	
5. **Pre-Conditions:** A user must be logged in.	
5.1. **Task Sequence** 1. Search screen will be shown by the system. 2. Enter the required information. 3. On clicking the search button, system will show the results.	
5.2. **Post Conditions** 1. User can view his desired results. 2. User can search any number of times.	
5.3. **Modification History:** Date 12/October/2021	
5.4. **Author:** Team Hack Inversion	

2.2 ACTIVITY DIAGRAM AND SWIMLANE DIAGRAM



*ACTIVITY DIAGRAM

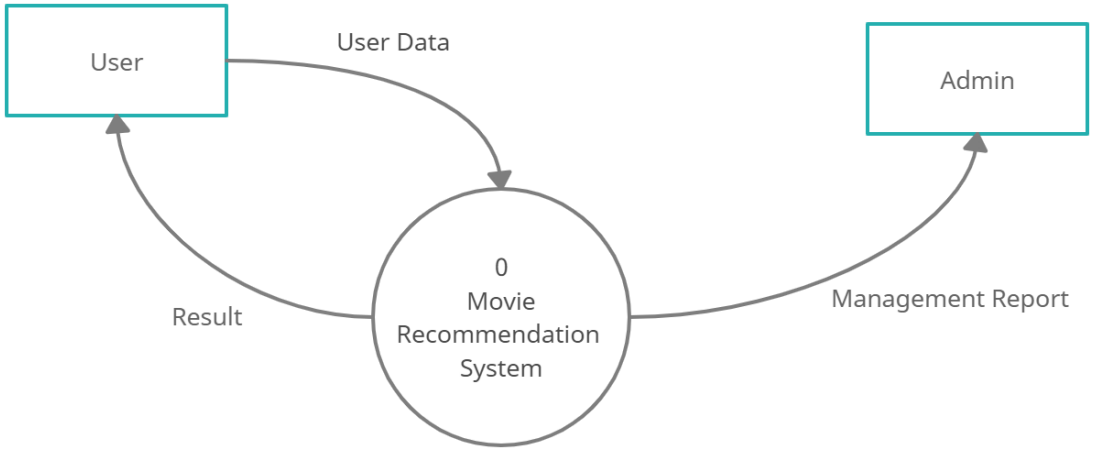
SWIMLANE DIAGRAM



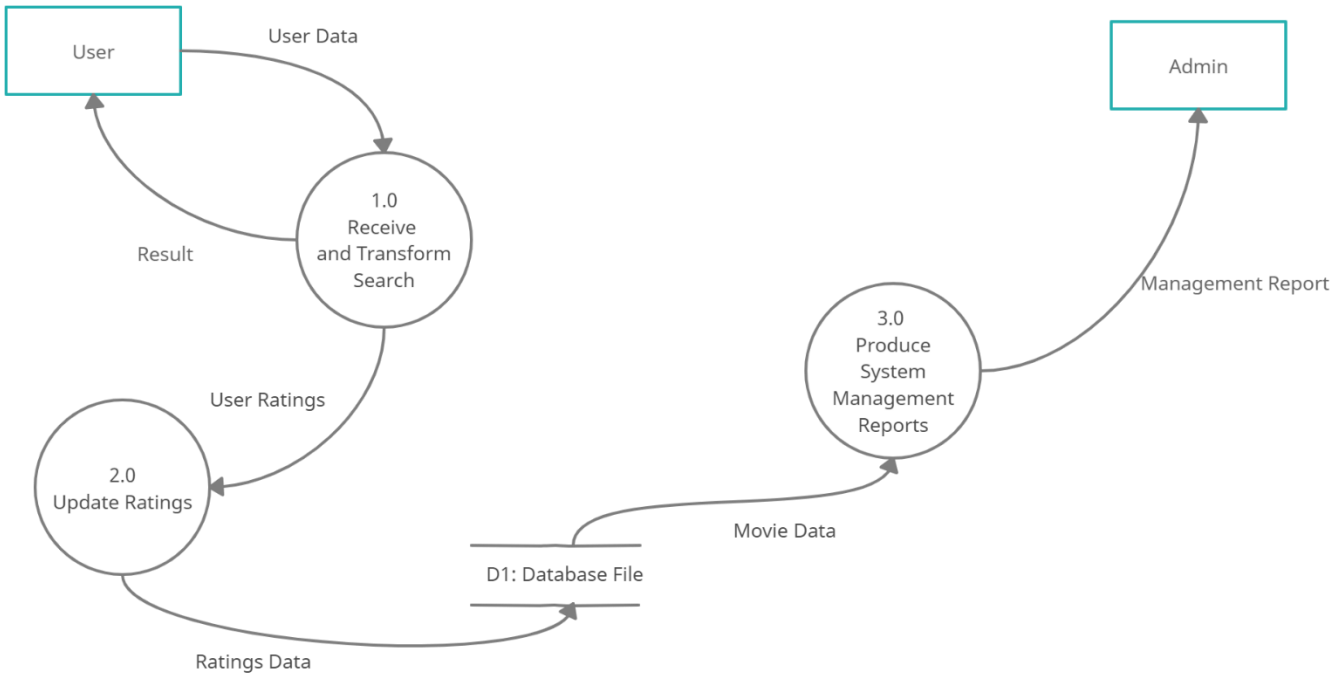
*Swimlane Diagram

2.3 DATA FLOW DIAGRAMS

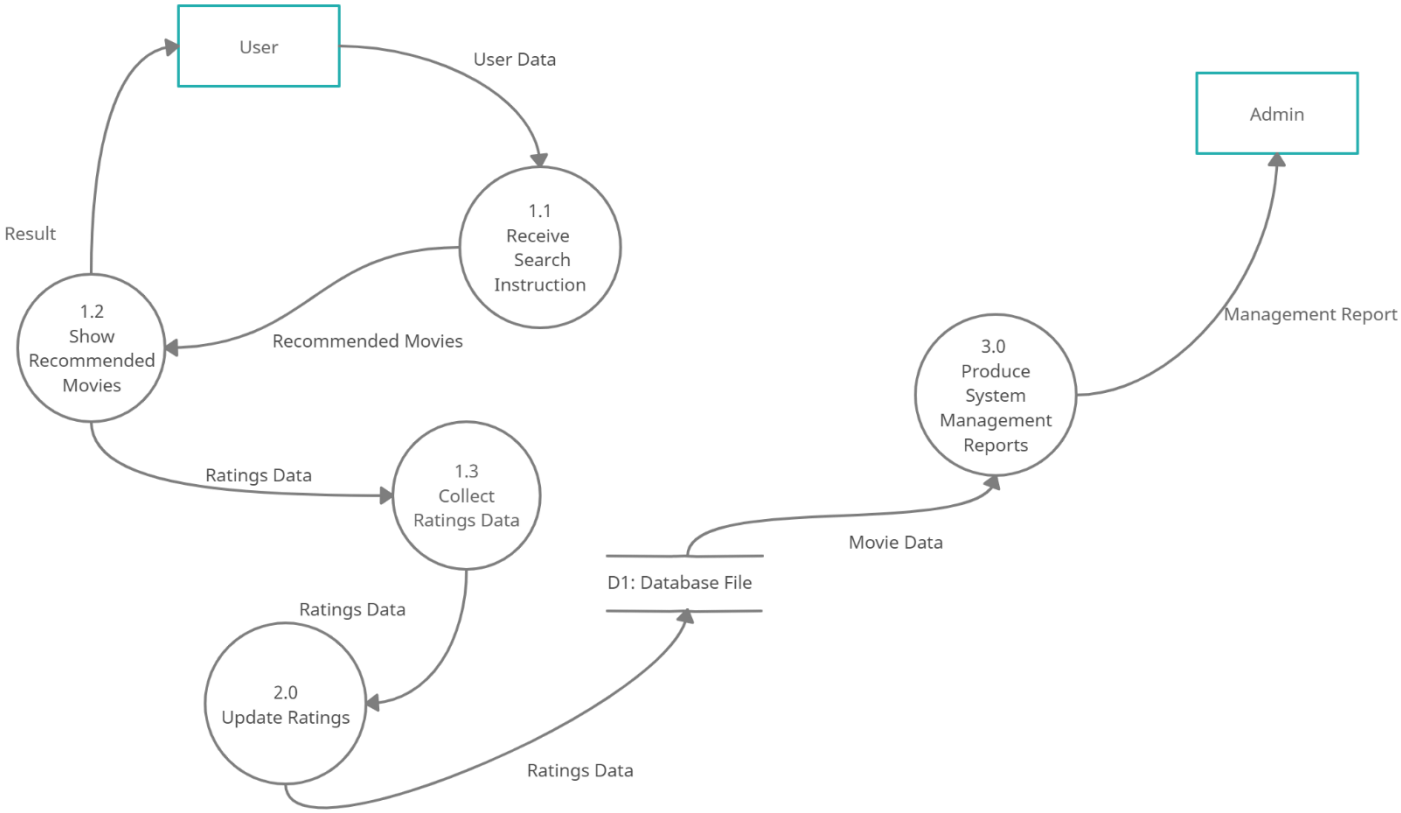
- 2.3.1. DFD LEVEL 0



- 2.3.2. DFD LEVEL 1



- 2.3.3. DFD LEVEL 2



2.4 SOFTWARE REQUIREMENTS SPECIFICATION IEEE FORMAT

SOFTWARE REQUIREMENTS SPECIFICATION

for

Movie Recommendation System

Version 1.0 approved

Prepared by

Abhi Garg

Gautam Aggarwal

Piyush Bawa

Sameer Alam

21-September-2021

Table of Contents

1. INTRODUCTION

1.1 PURPOSE

This document provides a detailed description of Software Requirements Specification (SRS) for "Movie Recommendation System". It is prepared according to ***IEEE*** Recommended Statements for Software Requirements Specification - ***IEEE Standard 830 – 1998***. The system makes an extensive use of user data to come up with reasonable movie predictions about user preferences and aims to improve the existing recommendation systems. This document includes the design, construction, project perspective, data model and future scope of the overall system. The target audience who wants to make use of this system, can find all related requirements information in this document. It assists the software developer team, the stakeholders and the end users.

1.2 DOCUMENT CONVENTIONS

This document uses the following document conventions:

1) Section headings and sub-sectional headings have been emphasised

in *Times New Roman Bold* with font-size 18 and 16 respectively.

2) Standards and Specifications have been italicized.

3) Keywords have been underlined.

1.3 INTENDED AUDIENCE AND READING SUGGESTIONS

The system is worth using by an audience that is interested to watch movies of their interest and benefit from facilities offered in such a case. Facilities include saving time by selecting the best movie according to the mood and comfort of the person and maintaining history to recommend more movies related to user's history by maintaining their own user profiles.

1.4 PROJECT SCOPE

The scope of this project is to design and develop a system that is made for easiness of the intended audience so that they can go on to the site and login/signup with their credentials and can get all the kind of movie recommendations under one site. This will help users to select their movie and not waste time searching for different genres over the internet. Also, the user profile feature in the system will maintain their history of movies and will recommend more by learning through this training set of each user.

Project scope from user perspective, limits the range of users to only those who have an internet connection.

1.5 REFERENCES

Please consult the subsection 1.4: Project Scope for information regarding project scope. A Use Case diagram and a Block Diagram has been added to accompany sections 2.1: Product Perspective and 2.2: Product Features.

2. OVERALL DESCRIPTION

2.1 PRODUCT PERSPECTIVE

The basics of the Movie Recommendation System are dependent on the researches in Cognition Science and Information Retrieval. The Recommendation system that we are going to develop is a follow-on member of present recommendation systems. It will take answers to specific questions and provide results based upon that. It will be an alternative to IMDB or Jinni, which will give exact results as searched by the user but is highly dependent on platforms like Prime Video, Netflix, or Hotstar for video streaming purposes.

2.2 PRODUCT FEATURES

The Recommender system is provided with a database of movies by the developer at the backend. Now a user comes, he first logs in into his account and proceeds with his work. The functionality provided is such that a user can search for a movie, which helps the system predict the right set of films that the user might watch. Also, he can even rate the movie on some scale.

The results are shown based on the movie's ratings, with the highest rated at the top.

2.3 USER CLASSES and CHARACTERISTICS

Remote Users most frequently use the device to get Recommendations for the movies. The customers are expected to have an Android Handset and a Know-How of English Language. Hence, the User Interface is kept pretty simple and available in English Language.

2.4 OPERATING ENVIRONMENT

The software will operate with following software components and applications:

The Frontend of the software is being developed by the use of JQuery and the Backend will be developed by the use of DJango. The Software after completing will be hosted locally.

2.5 DESIGN AND IMPLEMENTATION CONSTRAINTS:

We'll create a machine learning model to recommend acceptable shows to users depending on their preferences. We'll employ lighter models that are easier to adopt, as well as precise and quick, so that end customers get a quick response.

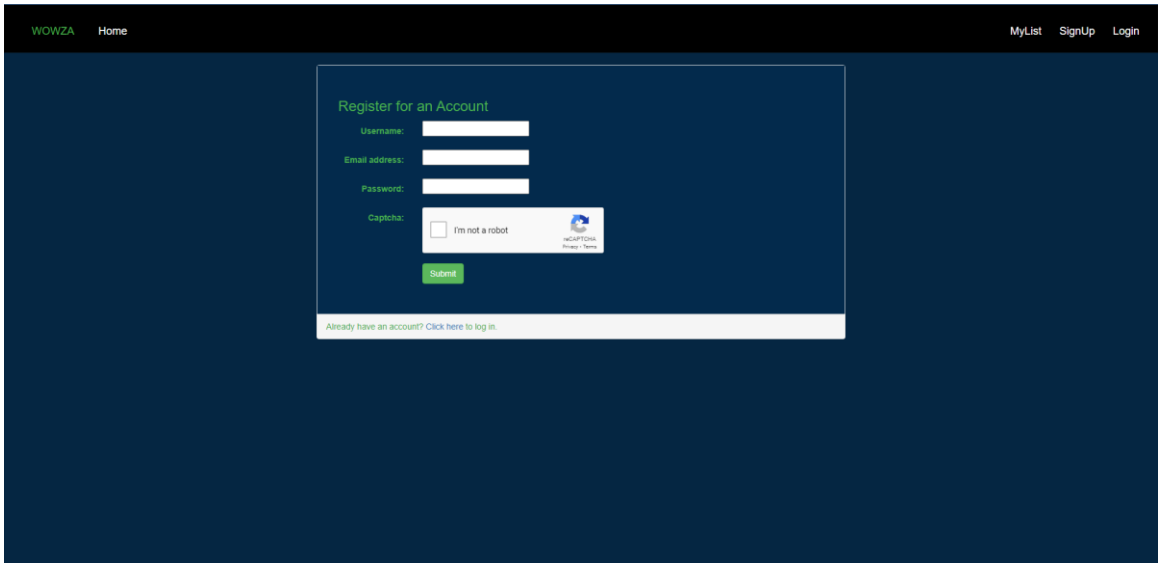
We'll also employ a secure login system and maintain track of a user's previously watched episodes.

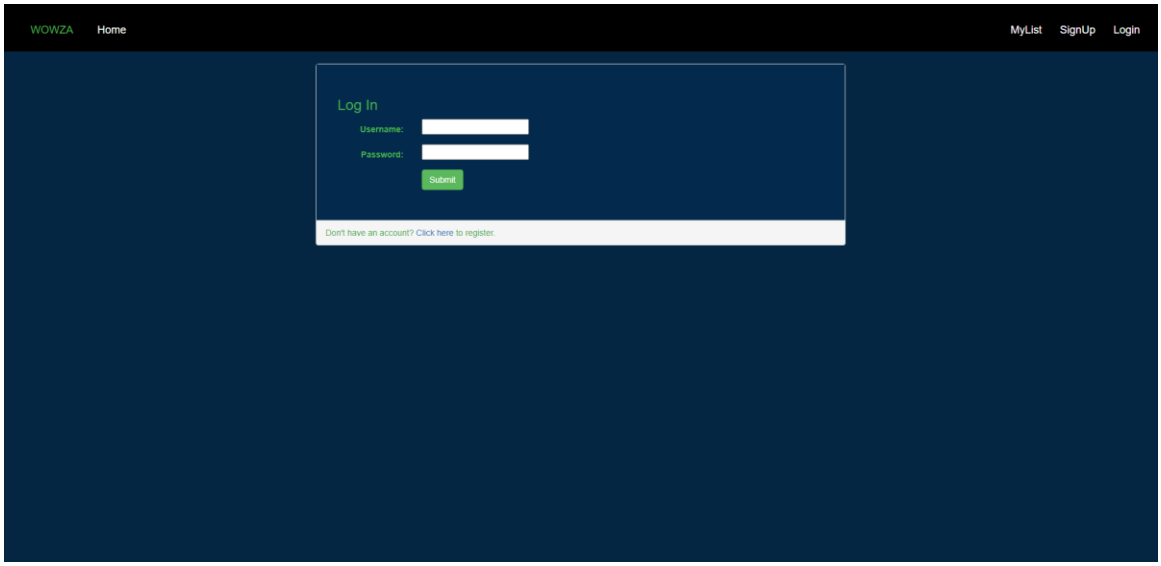
The number of users who can access or be online at the same time is referred to as server capacity. The higher the number of users, the higher the network traffic, and thus the service goes down. Personal firewall and updating is a difficult task; it should not block network traffic, causing the system to slow down. The server's firewall should not interfere with the user's firewall.

2.6 USER DOCUMENTATION

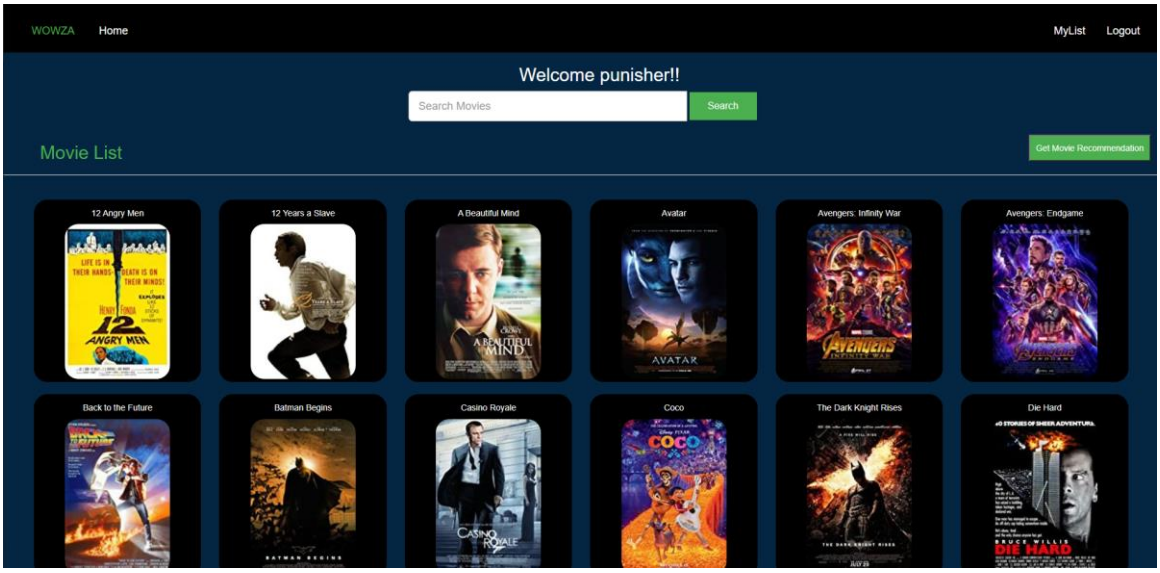
The User Interface is very simple and interactive. User is provided with set of fields to fill and will get output accordingly.
The website consists of the following pages:

1. User has to login/signup accordingly to have access to wowza

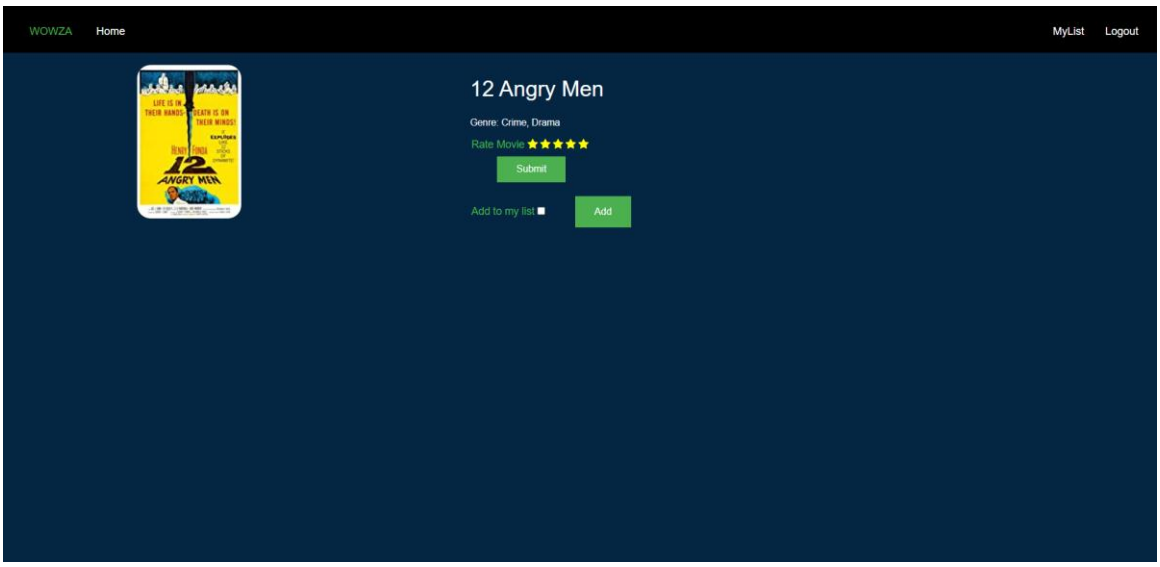




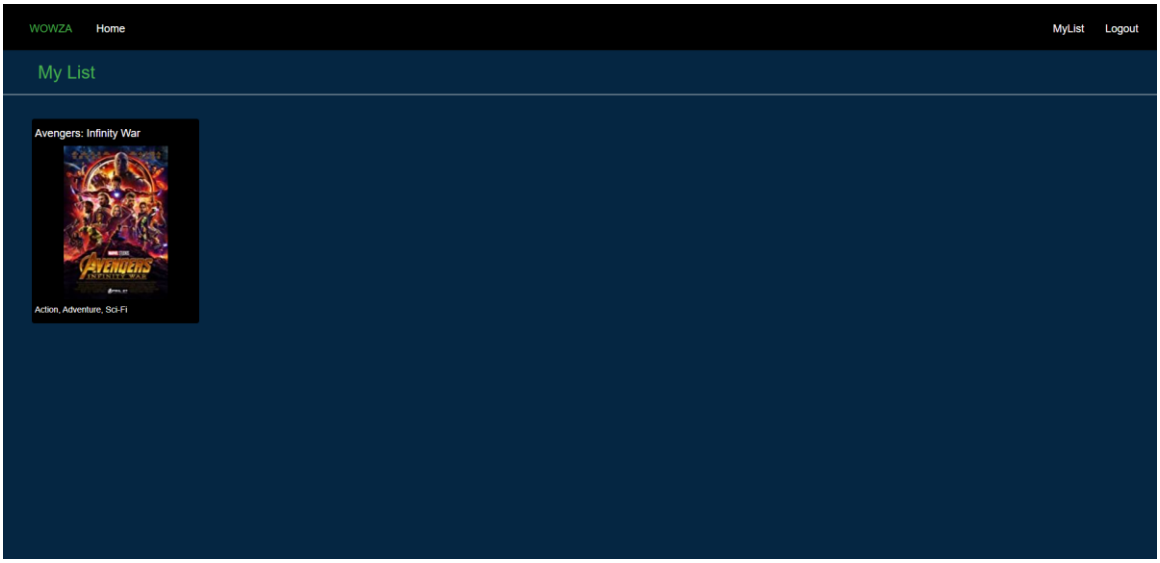
2. After login/signup user can have access to movies



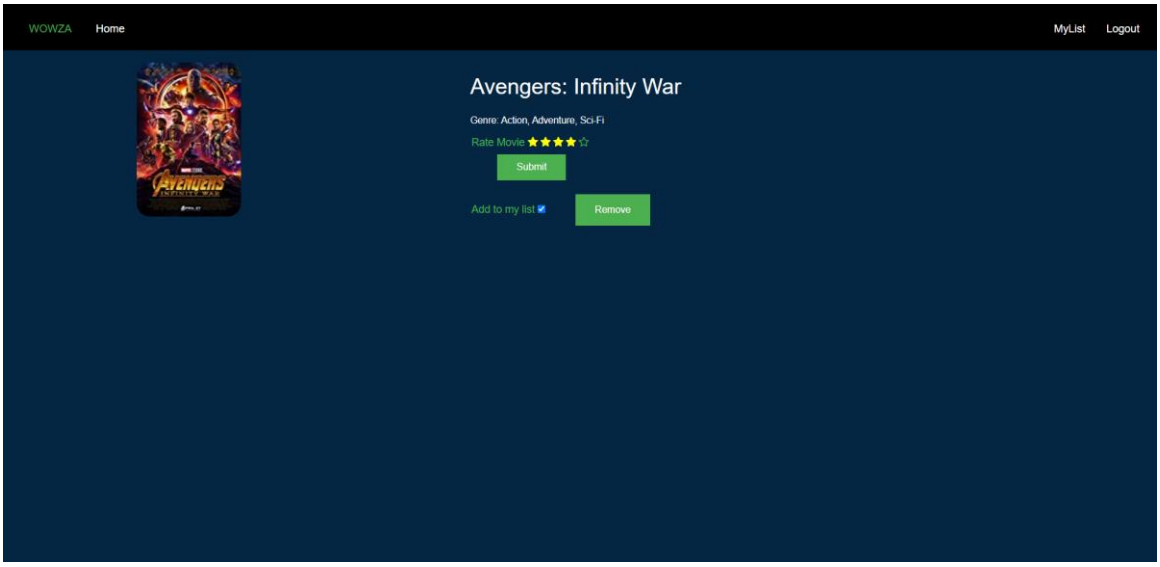
3. To rate or add movie to your list user can click on the thumbnail of the movie or can search in the search bar



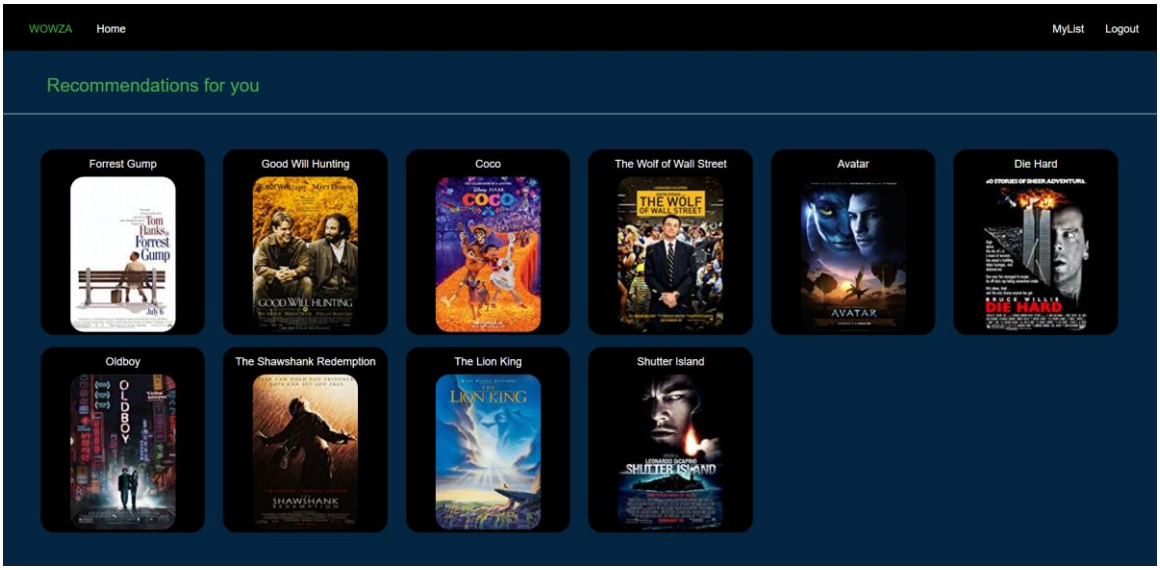
4. To view all movies added to your list user can click ‘MyList’ on top right



5. User can also remove movies from the list by clicking movies in ‘MyList’
   And then clicking the ‘Remove’ button



6. User can get movie recommendation by going back to ‘Home’ and clicking ‘Get Movie Recommendation’ button.



2.7 ASSUMPTIONS AND DEPENDENCIES:

We estimate that the number of consumers we will be interacting with within the early stages of this product will be approximately 1000. We will host our website on appropriate servers using free cloud services such as Heroku. If we wish to introduce this as our own product on the market, we'll also need to invest money in domain names.

To have the proper rights to use the shows that we would suggest to our users, we must obtain authorization from the respective proprietors.

3. SYSTEM FEATURES

3.1 DATABASE

The database is being handled by the admins who can add, remove or edit movies. It must be updated time-to-time to keep the latest movie name with their ratings available to the user.

3.2 LOGIN PAGE

Each user has to sign up/login with their credentials which will help the users get good recommendations related to their choice of movies and this will also maintain a history of each user by their unique login ids.

3.3 SEARCH AND GET RECOMMENDATIONS

Once the user has logged in this system will allow them to search movies. They can also rate the movie after watching to help our system learn and provide better recommendations in future.

4. EXTERNAL INTERFACE REQUIREMENT

4.1 User Interfaces

The Movie Recommendation System user interface has been specifically designed keeping in mind the user's comfort by allowing user to search movies according to their own taste and also get recommendations related to their previous history by maintaining a database.

This system allows the user to get ratings and streaming platform names so that the user just needs to go to the online platform and watch the movies rather then wasting time on different platforms searching for it.

4.2 HARDWARE INTERFACES

The hardware requirements include a good internet connection and a machine: -

Processor – Dual Core

Ram – 256MB or more

4.3 SOFTWARE INTERFACES

For frontend, JQuery will be used as framework for this project due to its high performance. The core of the framework offers a virtual DOM program and server-side rendering, which makes complex apps run extremely fast. This approach, as a result, helps to maintain high app performance and guarantees a better user experience.

For backend, Django will be used as our framework for this project due to its immense scalability and minimum web hosting costs. Also, Django hides the website's source code thus providing protection against XSS and CSRF attacks, SQL injections, clickjacking, etc. This was another important consideration we kept in mind while choosing framework for backend.

4.4 COMMUNICATION INTERFACES

This system is a web application; therefore network connection with TCP/IP protocol is required.

5. OTHER NONFUNCTIONAL REQUIREMENTS

5.1 PERFORMANCE REQUIREMENTS

- Fault Tolerant: System will be highly resilient and system components will be built such that they can handle failure of other components they are co-dependent on.

- Accuracy: System will provide accurate predictions about movie recommendations for the user and will accurately send and retrieve the necessary date to and fro from the system database.

- Security: Sensitive information and login credentials of the users will be highly secured.

5.2 SAFETY REQUIREMENTS:

The product is overflowing with kid-friendly entertainment, ranging from reboots to originals. Of course, it also includes a variety of age-inappropriate content that children may come across. We provide a few alternatives for parents to limit their children's access to explicit content. Parental controls are one approach to assist your child to learn to make good decisions and manage their screen time on their own. Here's how to use parental controls:

- Scroll down to Settings.
- Click on Parental Controls.
- Create a PIN.
- Scroll to Restrict by Maturity Level and click on the appropriate option.

We can also eliminate shows that contain some specific titles.

5.3 SECURITY REQUIREMENTS:

We understand that consumers are concerned about the security of their accounts, and we share that concern. Here are our top recommendations for keeping your account and personal information safe, whether you received an email from us advising you to change your password or you've seen suspicious activity.

- Use a password unique to this specific product and change it periodically.
- Add a phone number to your account for additional security.
- Sign out of unused devices.
- Be aware of possible phishing attempts.
- Keep your computer safe.
- Report fraudulent or suspicious activity.
- Report security flaws to us.

5.4 SOFTWARE QUALITY ATTRIBUTES:

Diverse and easy to use the product:

We provide users with both simple and advanced features. Due to its well-designed and easy-to-use interface, it can be used by both experts and typical users.

Security:

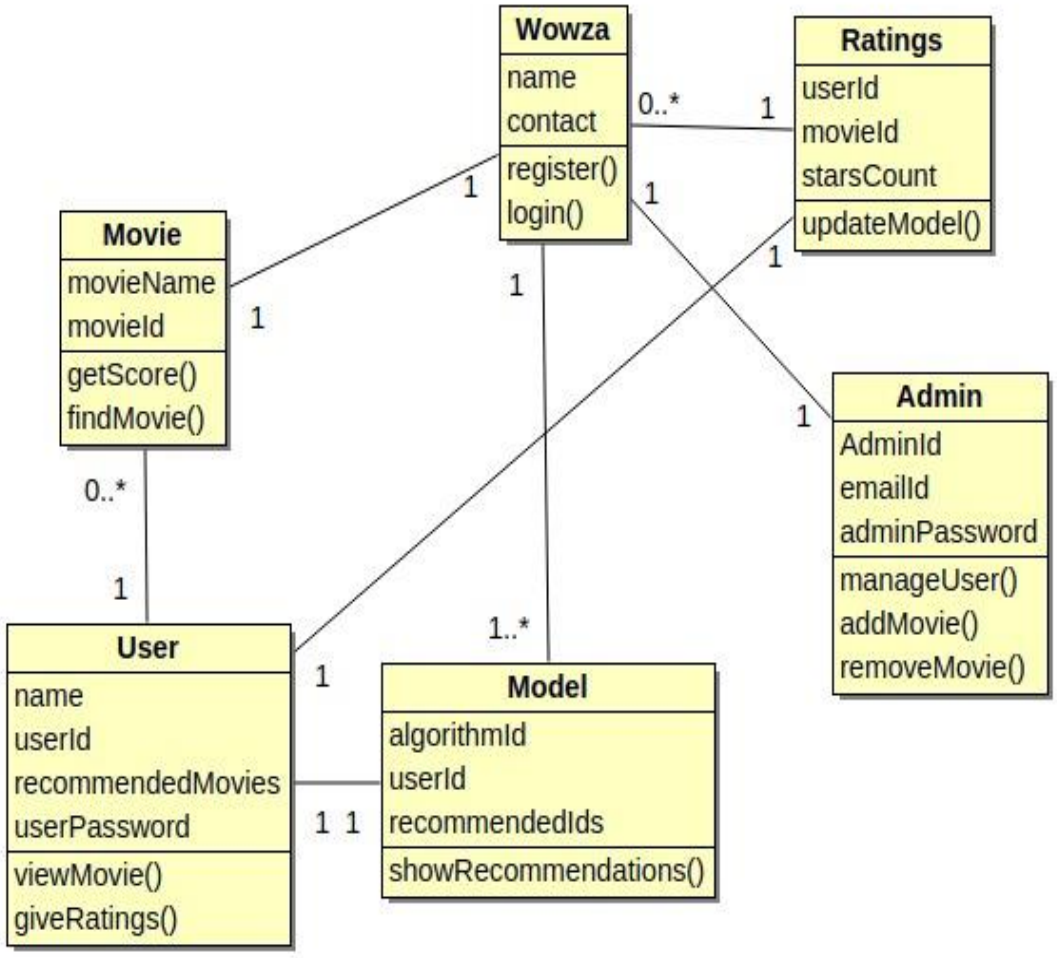
Because the key security concern is the user's account, a robust login process should be implemented to prevent hacking.

Usability:

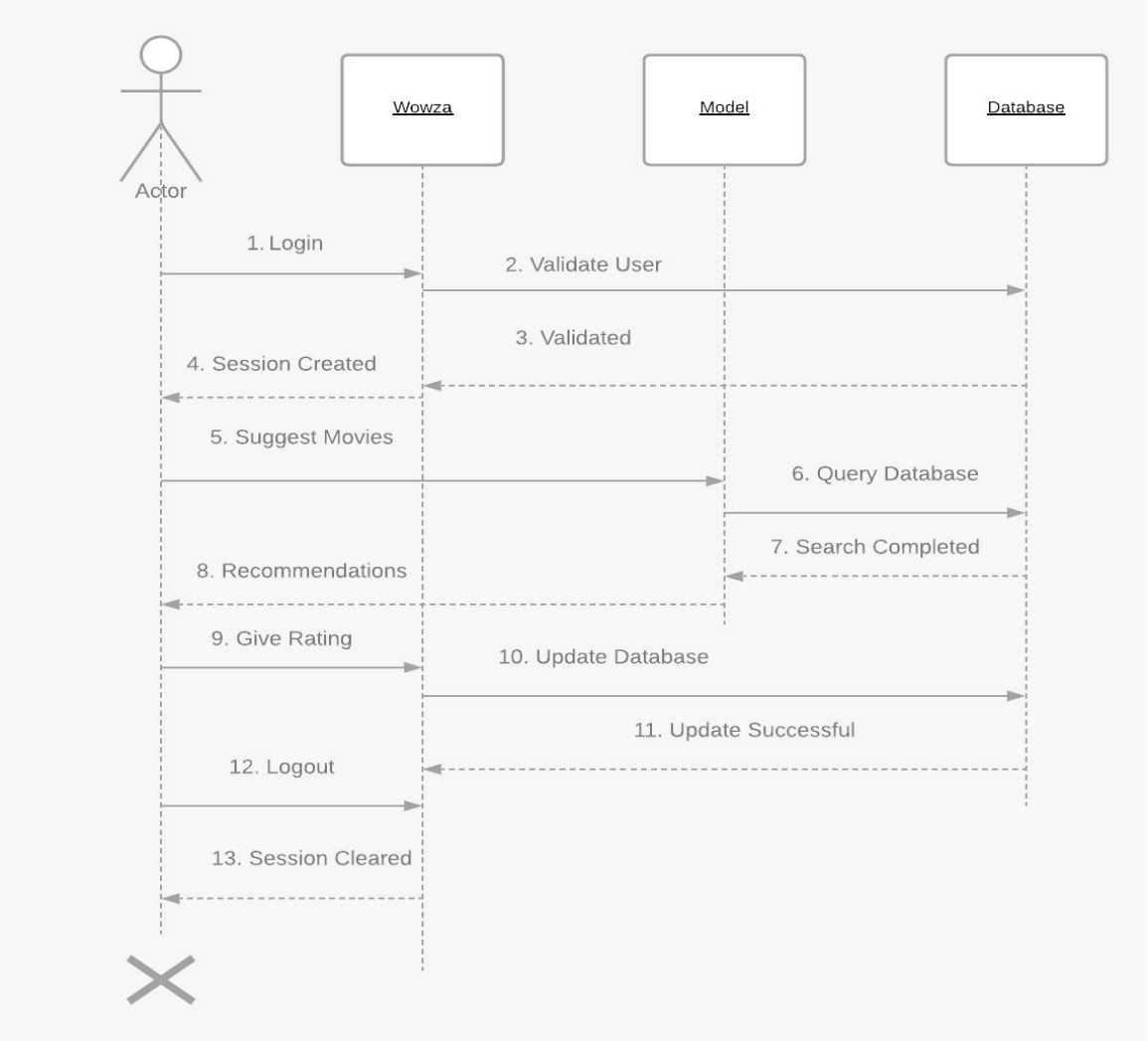
Because the system is simple to use and navigates in a predictable manner with minimal delays. In that instance, the system program reacts appropriately and switches between states fast.

3 DESIGN PHASE

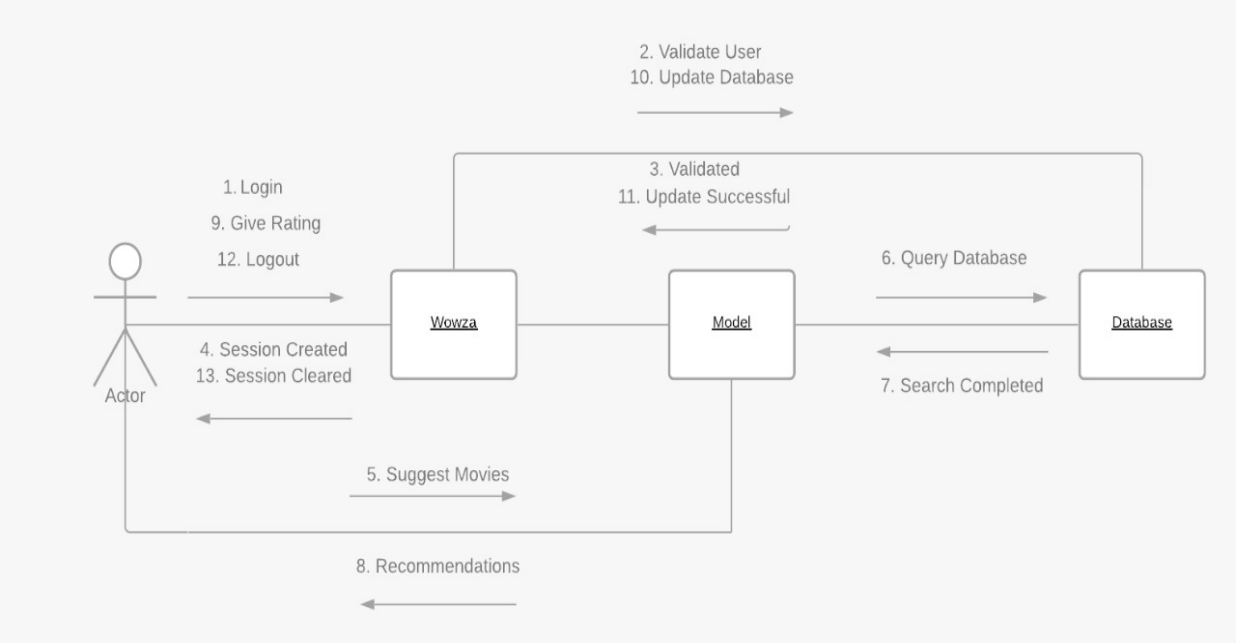
3.1 CLASS DIAGRAM



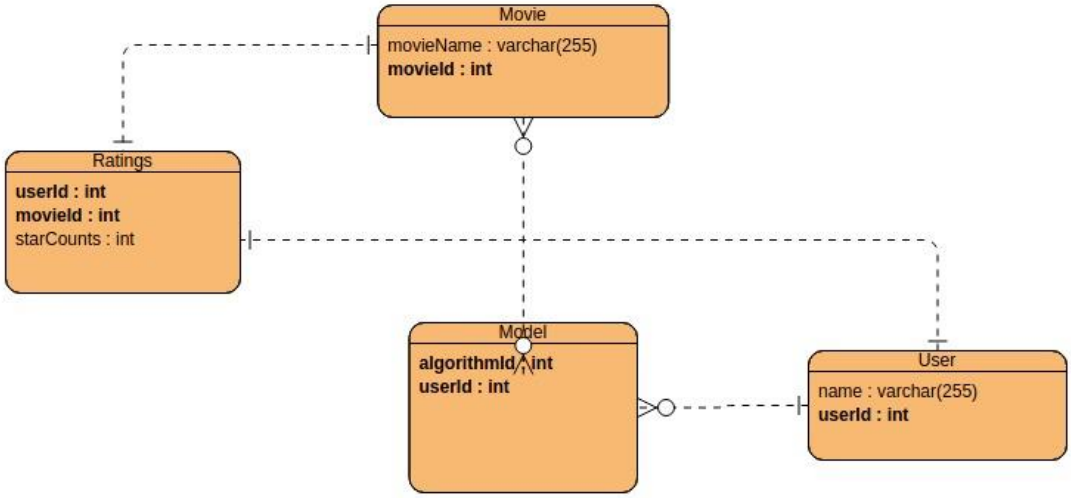
3.2 SEQUENCE DIAGRAM



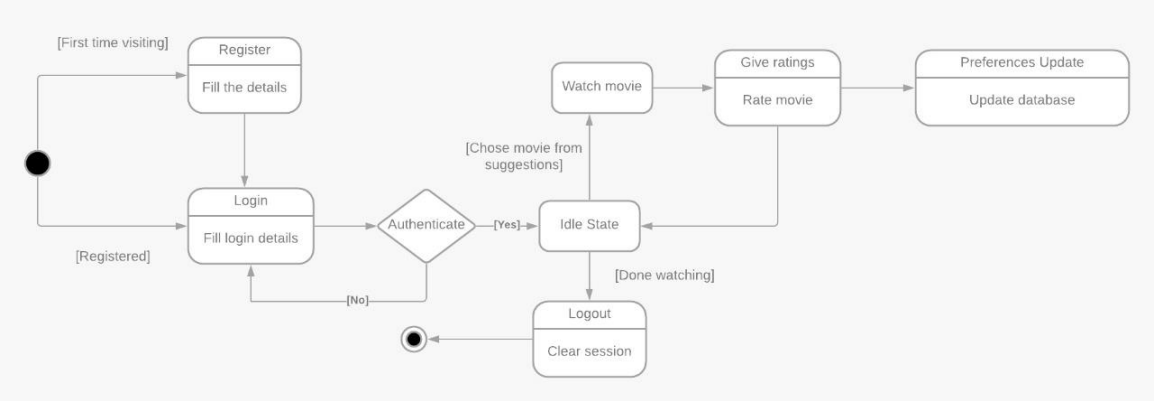
3.3 COLLABORATION DIAGRAM



3.4 ER DIAGRAM

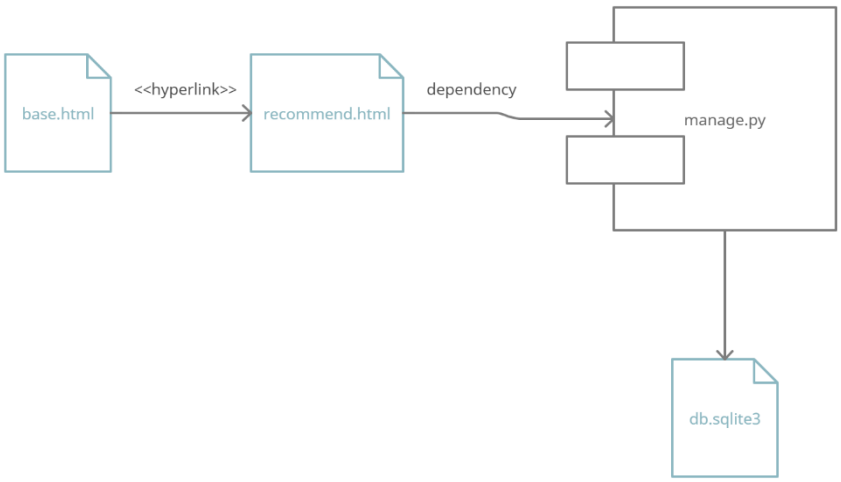


3.5 STATE DIAGRAMS

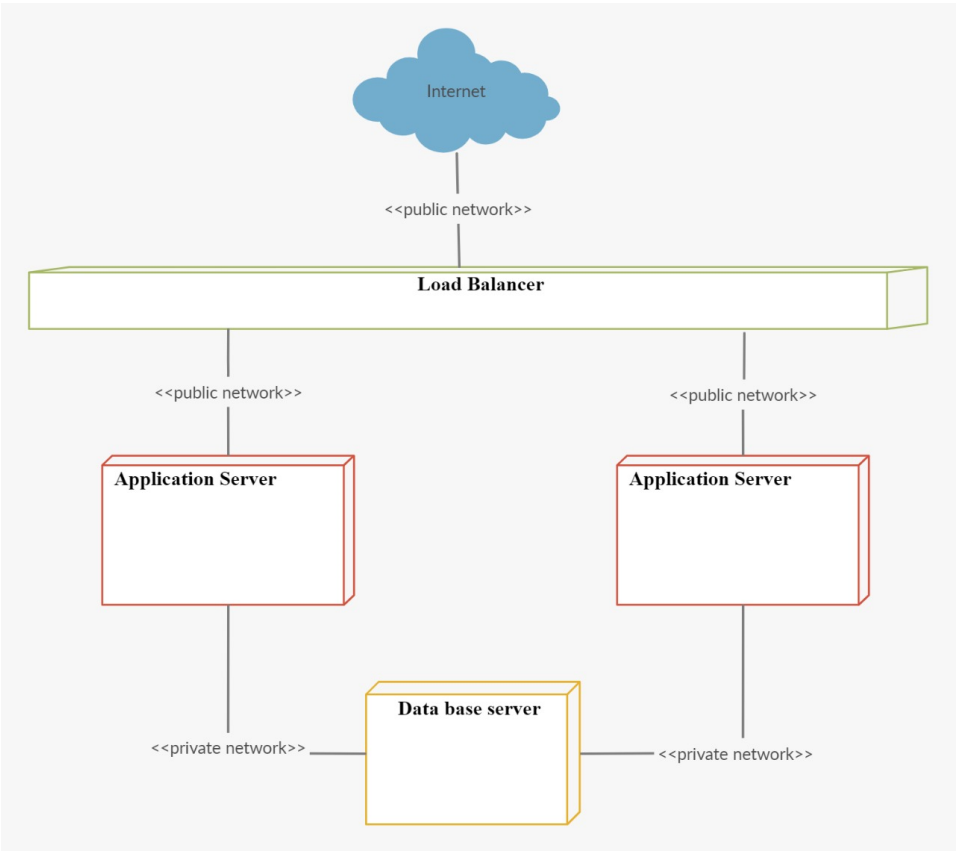


4 Implementation

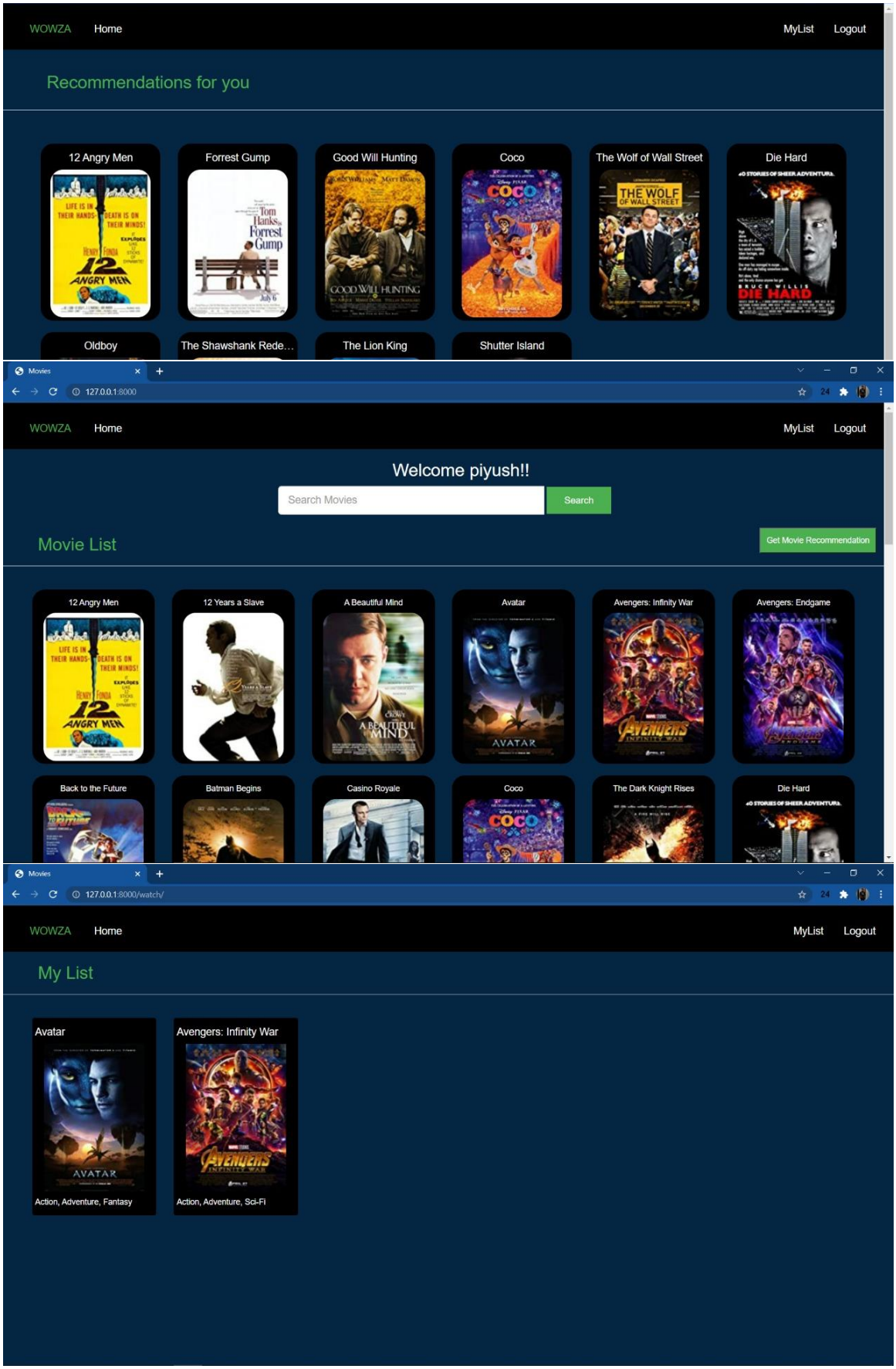
4.1 COMPONENT DIAGRAM

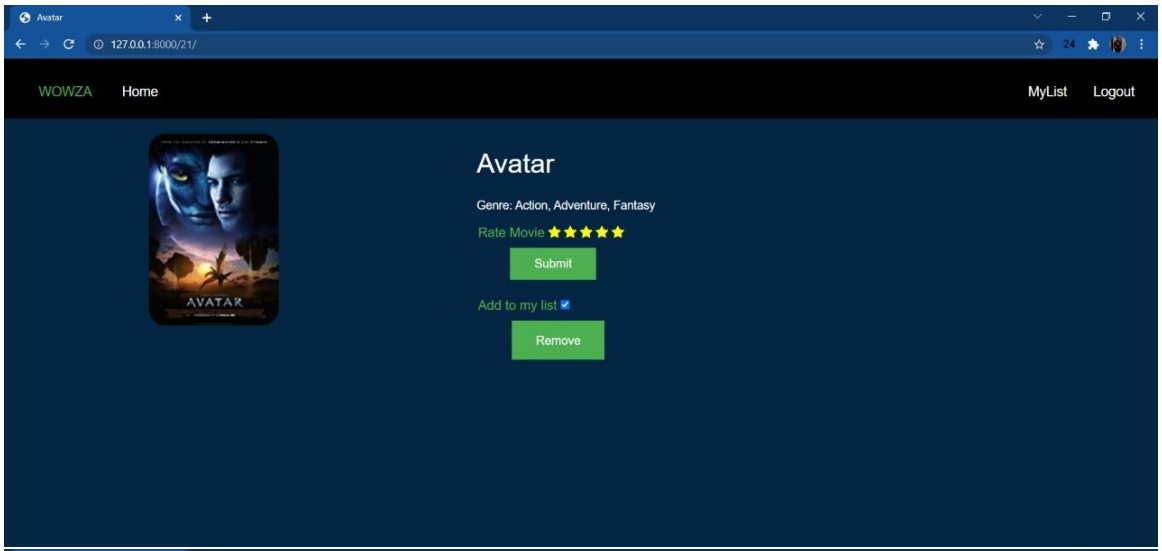


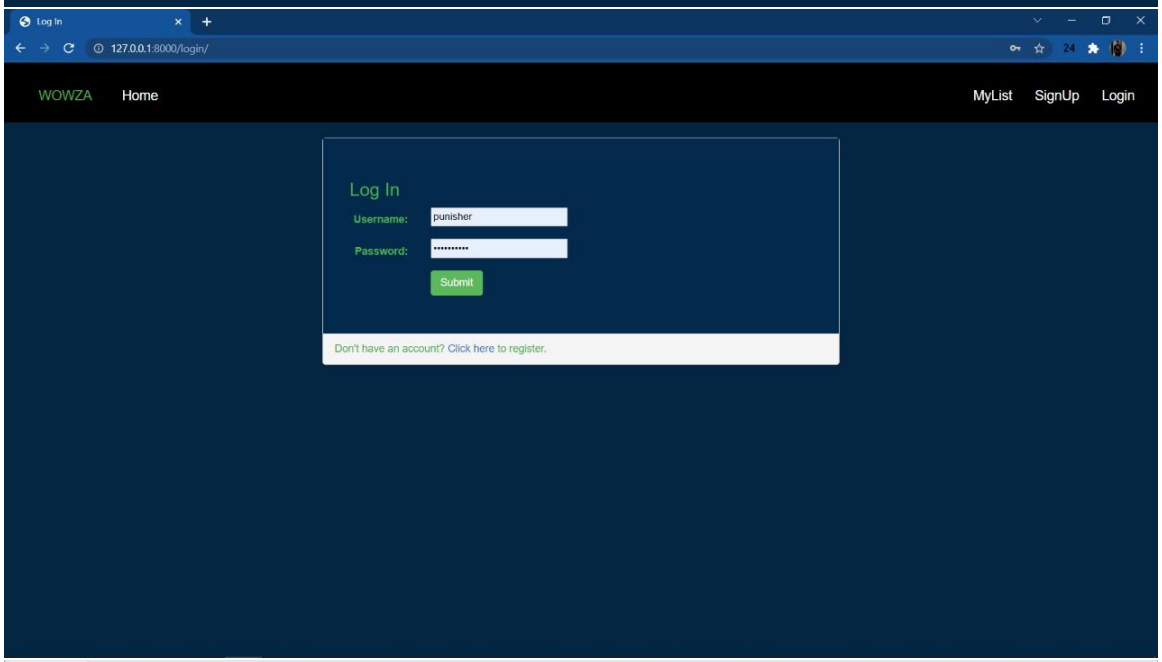
4.2 DEPLOYMENT DIAGRAMS

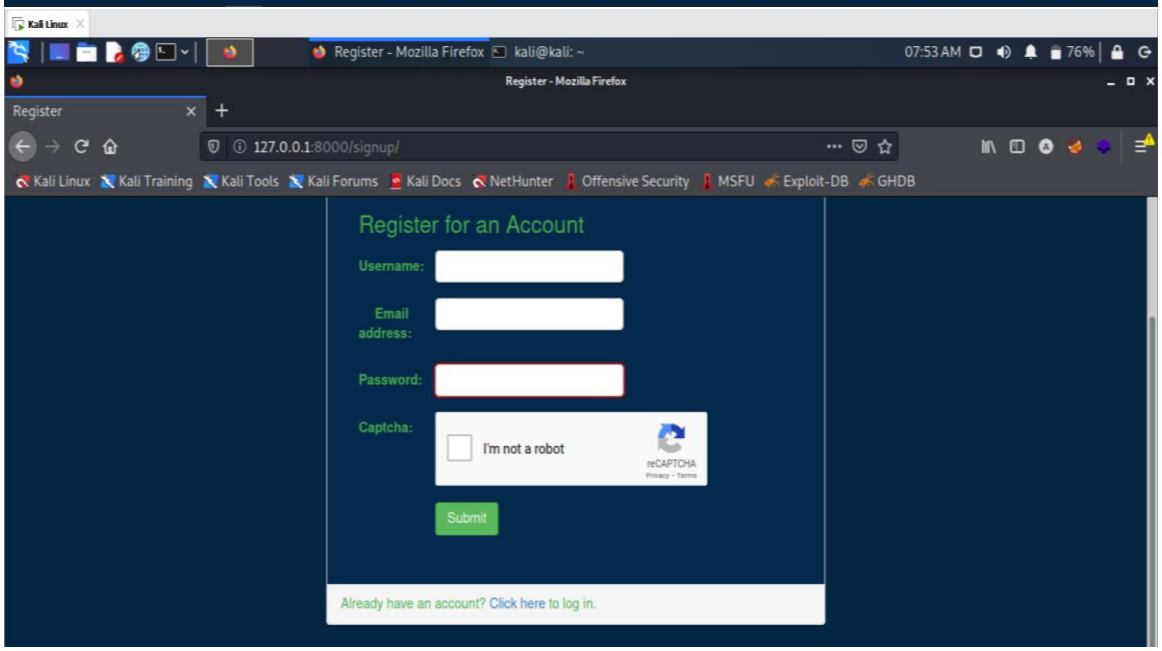


4.3 SCREENSHOTS




Avatar
127.0.0.1:8000/21/
WOWZA
Home
MyList
Logout
AVATAR
Avatar
Genre: Action, Adventure, Fantasy
Rate Movie
Submit
Add to my list
Remove


Log In
127.0.0.1:8000/login/
WOWZA
Home
MyList
SignUp
Login
Log In
Username:
punisher
Password:
Submit
Don't have an account? Click here to register.


Kali Linux
Register - Mozilla Firefox
kali@kali: ~
07:53 AM
76%
Register - Mozilla Firefox
Register
127.0.0.1:8000/signup/
Kali Linux
Kali Training
Kali Tools
Kali Forums
Kali Docs
NetHunter
Offensive Security
MSFU
Exploit-DB
GHDB
Register for an Account
Username:
Email address:
Password:
Captcha:
I'm not a robot
reCAPTCHA
Submit
Already have an account? Click here to log in.

5 Testing

5.1 TEST PLAN

The system type that is going to be tested is a Django application, which is hosted on IP 127.0.0.1. The aim of the test is simple, i.e., the team will test the working of some modules, their function and some of the related performance. If any of the issue is detected will be reported back to the team immediately.

In our report the testing type used is 'Specific Test Plans'. The modules and the areas that the team is going to test are-

- Security
- Performance
- SignUp

The resources that we are going to use are-

- Portswigger repository
- Linux OS
- Google Analytics

The activities are scheduled such that overall mechanism of the stated modules is measured.

- First, Security testing is implemented to test the system against a BOT or CSRF attack.
- The second test is implemented to check if the valid user is able to SignUp or not.

In the above process the requests and responses are checked thoroughly and any error is noted.

The results of the testing will tell the development team about the performance of the system, how secure it is, and any errors that emerged during test runs.

5.2 TEST CASES

Test Case #: 1.1 **Test Case Name:** reCAPTCHA Test
System: Movie Recommendation system **Subsystem:** SignUp
Designed by: Abhi Garg **Design Date:** 04/12/2021
Executed by: Abhi Garg **Execution Date:** 04/12/2021
Short Description: reCAPTCHA Security Testing on SignUp

Pre-conditions:

- The user is a BOT or there's been a CSRF Attack
- The BOT has credentials for a valid SignUp
- The system is running passively

Step	Action	Expected System Response	Pass/ Fail	Comment
1	A SignUp request is sent from a script to system with credentials and cookies/tokens	The system accepts the information.	Pass	All the information is gathered.
2	'Submit' button is pressed	System should display 'This field is required'.	Pass	The script won't be able to create a bogus user till the CAPTCHA field is not clicked upon.
3	Check Post-condition			

Post-conditions:

- No new user is created.

Test Case #: 1.2 **Test Case Name:** SignUp Test
System: Movie Recommendation system **Subsystem:** SignUp
Designed by: Abhi Garg **Design Date:** 04/12/2021
Executed by: Abhi Garg **Execution Date:** 04/12/2021
Short Description: Valid User SignUp with reCAPTCHA

Pre-conditions:

- The user is not a BOT
- The system is displaying Home Page

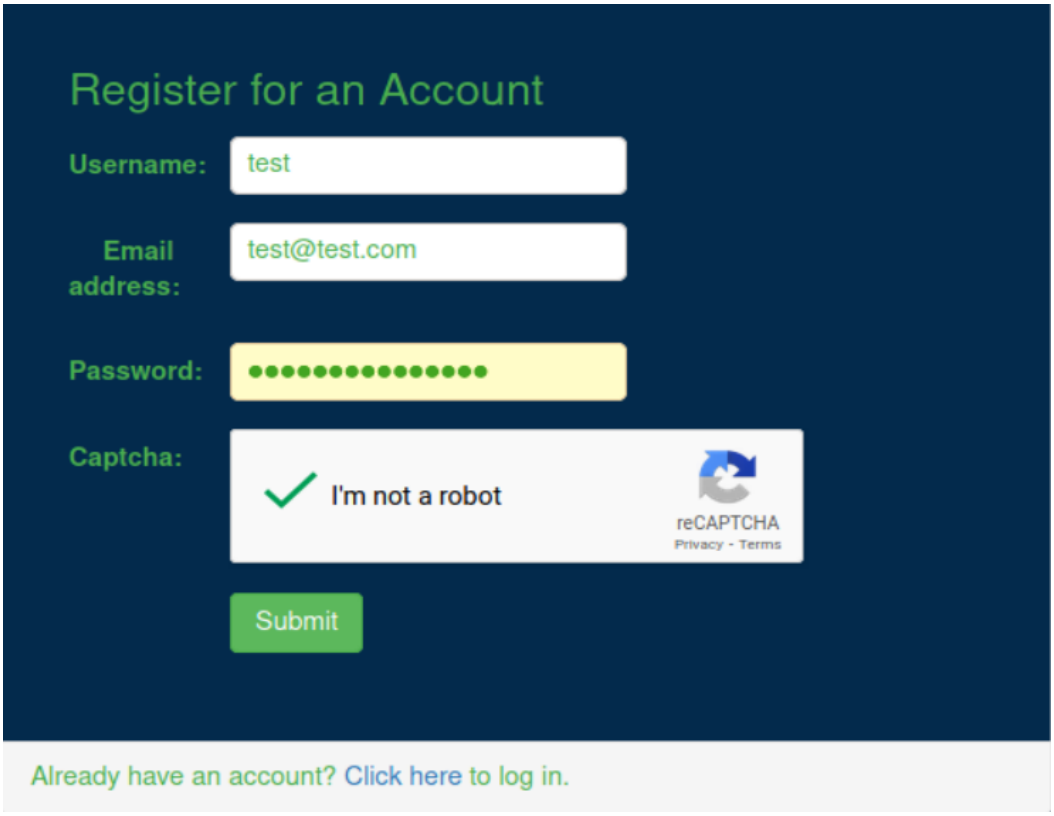
Step	Action	Expected System Response	Pass/ Fail	Comment
1	'SignUp' button is pressed.	The system displays the SignUp form.	Pass	All the information is gathered.
2	'reCAPTCHA' checkbox is filled.	Security check is done to detect a BOT or human.	Pass	A report is made by google and sent to owner.
3	'Submit' button is clicked.	System checks ambiguity in username and access is granted.	Pass	
4	Check Post Condition			

Post-condition:

- A new user is created a saved in the database.

5.3 TEST REPORT

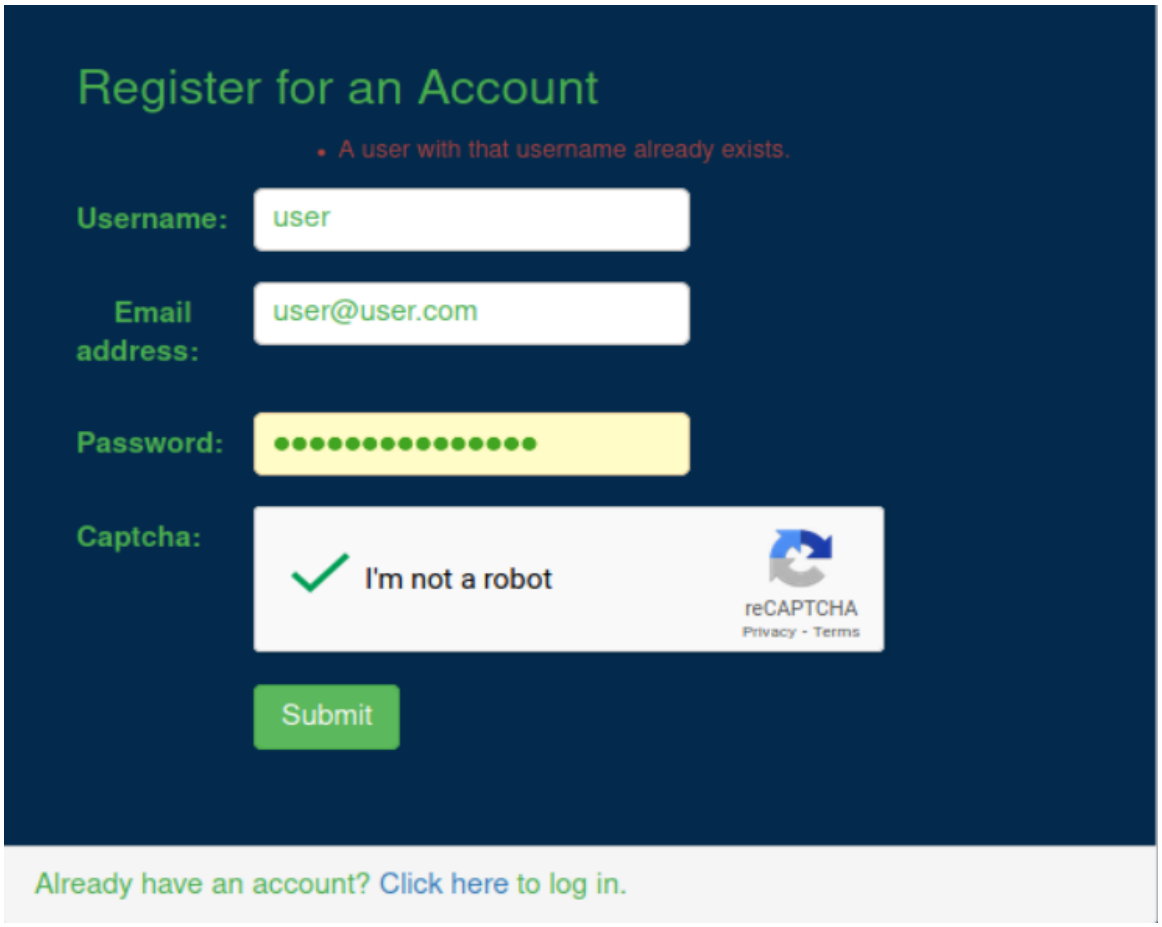
The following results are based upon Test Case- 1.1

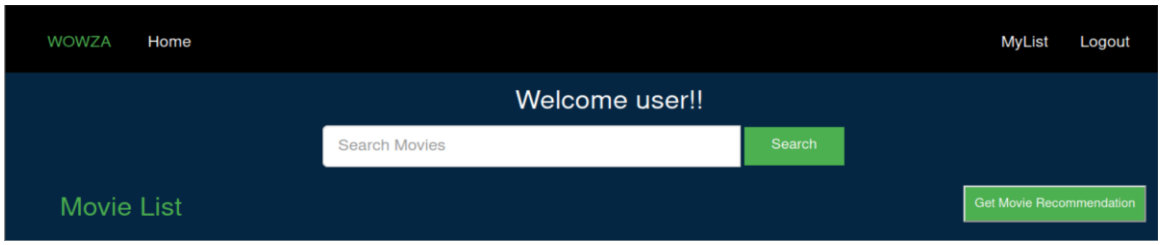



```
----------------------------152456325527232386642305124524
Content-Disposition: form-data; name="csrfmiddlewaretoken"

FnyjF4yyTUh2EPmjCSvefH6JgOnsmvojPTtiVmLIUpWxu2UI2hRfd9dzZx2Ossv8
----------------------------152456325527232386642305124524
Content-Disposition: form-data; name="username"

test
----------------------------152456325527232386642305124524
Content-Disposition: form-data; name="email"

test@test.com
----------------------------152456325527232386642305124524
Content-Disposition: form-data; name="password"

Axt7Bure3gJTpWp
----------------------------152456325527232386642305124524
Content-Disposition: form-data; name="g-recaptcha-response"

03AGdBq25bM7yVPDnPzyzj8wZXD_bYUkC59ZzEU8Ctg5FRL7gw6gRgPd0gfnvLIGx-eZvNryrOkim-t-bcc31K0feXbExZtYrBqoJmuTSLTSY2S4StG4ITSNbFHeLMcDBl2xE7--A2Azy8RgX3fCXvlBWQ33lnxOC-7JwUVBZAjFs
DFXOxPZOrenASyWnw9_TQfwn8ftghJtjW6vXOs5RDvYzJHl3IabXvQwrGCREYCW-wG9AM4-LhFe3UADLBhdwRfnLEhVIRnOYJvQkFAhg-8JPSqpOq-uzA6JFIra7QN7jWv48xR8JdJarOtihs954Unns294hUtKIhGfRqi43W4mfF
jWQWaYHjNtRDRblKX9WwZ5nQ41dSqVITFXledOB4WwBP6weW9PfxeYq5Ne3BsFIOtiE7YKOKkkGITUxMbPW1XX16_-tv4upktRhqoBJ5X5D-GafmBMSPGEwZWYur8y2pvxe_mtli9nitMUjYLOIW3nz-JE8y2X4
----------------------------152456325527232386642305124524--
```

So, above is the snippet of the SignUp request with credentials. Here, at the end is the reCAPTCHA key that will be authenticated. The key is different at all the time and hence, it is nearly impossible for a BOT or attacker to guess it. So, SignUp module is Secure and working all fine.

The following are the test results of Test Case- 1.2





Here, we can see that a normal user can signin to the system very easily.

Now, the results of the test cases conclude that the tested modules are working fine and no error has been detected.