RV COLLEGE OF ENGINEERING®, BENGALURU-560059

(Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Online Streaming Platform - Movieholic

Mini - Project Report

Submitted by

MAHEK JAIN **USN 1RV18CS082**
SRISHTI MOORTHY **USN 1RV18CS174**

In partial fulfillment for the requirement of 5th Semester

Software Engineering Laboratory Mini Project (18IS55)

Under the Guidance of

Prof. Prapulla S.B.,
Assistant Professor, Department of Computer Science and Engineering,
RV College of Engineering,
Bengaluru

Academic Year 2020- 2021

RV COLLEGE OF ENGINEERING®,BENGALURU - 560059
(Autonomous Institution Affiliated to VTU, Belagavi)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

Certified that the project work titled **'Online Streaming Platform - Movieholic'** is carried out by **Srishti Moorthy, 1RV18CS174** and **Mahek Jain, 1RV18CS082,** who are bonafide students of R V College of Engineering®, Bengaluru, in partial fulfillment of the curriculum requirement of 5th Semester Software Engineering Laboratory Mini Project during the academic year **2020-2021**. It is certified that all corrections/suggestions indicated for the internal Assessment have been incorporated in the report deposited in the departmental library. The report has been approved as it satisfies the academic requirements in all respect laboratory mini-project work prescribed by the institution.

Signature of Faculty In-charge

Head of the Department
Dept. of CSE, RVCE

External Examination

Name of Examiners	Signature with date
1	
2	

ACKNOWLEDGEMENT

Any achievement, be it scholastic or otherwise does not depend solely on the individual efforts but on the guidance, encouragement and cooperation of intellectuals, elders and friends. A number of personalities, in their own capacities have helped me in carrying out this project work. I would like to take this opportunity to thank them all.

I deeply express my sincere gratitude to my guide **Prof. Prapulla S.B., Assistant Professor,** Department of CSE, RVCE, Bengaluru, for his able guidance, regular source of encouragement and assistance throughout this project.

I would like to thank Dr. Ramakanth Kumar P, Head of Department, Computer Science & Engineering, R.V.C.E, Bengaluru, for his valuable suggestions and expert advice.

First and foremost I would like to thank **Dr. Subramanya. K. N**, Principal, R.V.C.E, Bengaluru, for his moral support towards completing my project work.

I thank my Parents, and all the Faculty members of Department of Computer Science & Engineering for their constant support and encouragement.

Last, but not the least, I would like to thank my peers and friends who provided me with valuable suggestions to improve my project.

Abstract

Our project is an Online Streaming Platform system, which is designed to make the streaming experience for the customers a lot more enjoyable and convenient. In the current pandemic situation, it is not advisable to go to the movie theatres and thus our system provides a wide variety of movies and TV shows. Also the current platforms do not offer a personalized experience, rather they cater to the masses, and which is why we have taken customers' ratings and reviews into consideration to provide more accurate recommendations.

The customer is first required to sign up or login to the platform (Figure B.1). There is an attractive home page (Figure B.2), which allows the customers to search for the movies (Figure B.4) based on the movie name. Customers can also update their profile information (Figure B.3). Customers are required to make the payment for the subscription plan to stream the movies (Figure B.6). They are directed to a third party page to complete the payment process. The customer can also rate movies, write reviews and view additional information about the movies (Figure B.5). They can also see a list of similar movies and view their recommended movies list (Figure B.8). Customers are also provided with an option to purchase a movie, if not available on the website (Figure B.7).

The project has been implemented using HTML, CSS, Javascript and PHP, and the databases used are MySQL and MongoDB. NLP has been used to analyse the users' ratings and reviews for better movie recommendations. The website has a simple and user-friendly user interface, which helps the users to navigate through the website easily.

Table of Contents

List of Figures

List of Tables

GLOSSARY

API	:	Application Programming Interface
SRS	:	Software Requirement Specification
BaaS	:	Backend as a Service
DFD	:	Data Flow Diagram
NF	:	Normalized Form
ER	:	Entity Relationship
HTTP	:	Hyper Text Transfer Protocol
REST	:	Representational State Transfer

Chapter 1

Introduction

1. Introduction

1.1 Objective

Our project is an Online Streaming Platform system, which is designed to make the streaming experience for the customers a lot more enjoyable and convenient. In the current pandemic situation, it is not advisable to go to the movie theatres and thus our system provides a wide variety of movies and TV shows. Also the current platforms do not offer a personalized experience, rather they cater to the masses, and which is why we have taken customers' ratings and reviews into consideration to provide more accurate recommendations.

1.2 System Overview

The proposed online streaming platform is an extremely useful alternative as it sorts out movies not only based on the popularity but also the customers' reviews. It delivers a wide range of series and movies based on the likes and dislikes of the customers and provides them a good streaming experience. To achieve this, we will be analyzing not only the star ratings given by the customers to movies, but also their written reviews. This will help us to customize movies and web series recommended to a user, and give them a good viewing experience. We also intend to provide a feature for the user to purchase a movie if it is unavailable.

1.2.1 System Description

The idea of this product was conceived in order to present a more personalized movie and TV show viewing experience to the users. The current platforms do not offer a personalized experience, rather they cater to the masses, and they also show movies which are trending in certain geographical locations. The proposed software aims to provide a personalized viewing experience to users by analyzing how they rate movies, and the reviews they write. The review analyzing software can function as an additional component in existing systems, or it can be independently used in an entirely new platform. However, the proposed system is an online movie streaming system which provides a personalized viewing experience to users.

The proposed system has the following essential features:-

1. The system will require a very simple user interface.
2. NLP analysis to provide recommendations based on the user ratings.

3. NoSQL database to hold all the user comments and reviews. A NoSQL database is ideal for storing such data, since it provides flexibility, because the length of the reviews is not fixed, so it will be difficult to make a rigid schema for the same.
4. Menu bar: This offers basic features such as searching for a movie. The user can directly type in the name of the movie, or the actor to search for. Other basic features like viewing profile information, updating profile information etc. are also provided.

5. When the user chooses a movie to watch, the display switches to the next screen, and the movie begins playing there. The user can also rate movies and write reviews. He can also see a list of similar movies recommended to him in the recommendations page.

1.2.2 Assumptions and Constraints

The assumptions are:

- The coding should be error free.
- The system should be user-friendly so that it is easy to use for the users.
- All the information regarding the star ratings and the reviews must be stored in a database.
- The system should have more storage capacity and provide fast access to the database.
- Users may access from any computer that has Internet browsing capabilities and an internet connection.
- Users must have their correct usernames and passwords to enter into their accounts.

The constraints are:

- The reviews written by the users will be analyzed using NLP, in order to make movie recommendations to the users. This might not give 100% accuracy of predictions
- Storage of huge data.
- The movie/series might not be available to watch or purchase.
- Video streaming constraints.
- Constraints with the resources for high performance.
- SQL commands for above queries/applications
- Implement the database at least using a centralized database management system.

Chapter 2

Software Requirement specification

2.1 Introduction

2.1.1 Purpose

The purpose of the S.R.S is to present a detailed description of the management of the online streaming platform. It will explain the purpose and features of the software, the interfaces of the software, what the software will do and the constraints under which it must operate. This S.R.S is intended for users of the software and also potential developers. This S.R.S will also provide a reference to verify whether the developed software meets the intended objectives, at any phase of development.

2.1.2 Document Conventions

This SRS was created based on the IEEE template for System Requirement Specification Documents.

2.1.3 Intended Audience

- Websites that store private and confidential information of their customers.
- Application Administrators as they are critical to keeping the applications our organization relies on running.
- Customers/clients that will be using the online system for streaming, for eg: students, employees etc.
- Programmers/testers/developers, who are interested in the field of NLP or who want to improve this project by fixing bugs/adding functionalities etc.

2.1.4 Reading Suggestions

END USERS: 1.1 purpose, 1.5 Product scope, 1.6 References, 2.2 Product functions, 2.4 Operating environment, 2.6 User documentation, 4 System features, 6 Glossary

PROGRAMMERS: 1.1 purpose, 1.5 Product scope, 2.2 Product functions, 2.4 Operating environment, 2.5 Design and implementation constraints, 2.6 User documentation, 3 External Interface Requirements , 4 System features, 5 Other non functional requirements, 6 Other requirements

ADMINISTRATOR: 1 Introduction, 2.2 Product functions, 2.4 Operating environment, 2.6 User documentation, 3 External Interface Requirements, 4 System features, 5 Other non functional requirements, 6 Other requirements

2.1.5 Product Scope

The proposed online streaming platform is an extremely useful alternative as it sorts out movies not only based on the popularity but also the customers' reviews. It delivers a wide range of series and movies based on the likes and dislikes of the customers and provides them a good streaming experience. To achieve this, we will be analyzing not only the star ratings given by the customers to movies, but also their written reviews. This will help us to customize movies and web series recommended to a user, and give them a good viewing experience. We also intend to provide a feature for the user to purchase a movie if it is unavailable.

2.1.6 References

1. **Fan Qui and Yi Cui(2010).”An analysis of user behavior in online video streaming”, DOI: 10.1145/1878137.1878149**
2. http://courses.ece.ubc.ca/cpen441/preProposal/preproposal_pdfs/yangmichelle_NetflixWebApplicationUIAnalysis.pdf
3. https://www.academia.edu/40417916/Software_Requirements_Specification_For_Netflix_Movies_and_TV_Streaming_School_of_Computer_Science_and_Engineering_LOVELY_PROFESSIONAL_UNIVERSITY
4. https://www.studocu.com/in/document/lovely-professional-university/software-engineering/other/glee-software-requirement-specification-srs/3071300/view

2.2 Overall Description

2.2.1 Product Perspective

The idea of this product was conceived in order to present a more personalized movie and TV show viewing experience to the users. The current platforms do not offer a personalized experience, rather they cater to the masses, and they also show movies which are trending in certain geographical locations. The proposed software aims to provide a personalized viewing experience to users by analyzing how they rate movies, and the reviews they write. The review analyzing software can function as an additional component in existing systems, or it can be independently used in an entirely new platform. However, the proposed system is an online movie streaming system which provides a personalized viewing experience to users.

2.2.2 Product Functions

The major product functions are listed as follows:

- Login module: This allows existing users to login and new users to create an account. Subscription module: The user can purchase a monthly subscription or renew his existing subscription.
- Movie search and streaming: This allows users to search for movies by name, language, genre and actors, and watch the movie. Users are given the option of purchasing unavailable movies from other websites.
- Movie recommendation: Personalized movie recommendations are made to users based on how they rate movies, and the reviews they write.

2.2.3 User Classes and Characteristics

This software is designed for use by both experienced and inexperienced computer users. It is an entertainment system which caters to a wide range of users. The main classes of people who will use this platform are students and the general working population. However, the usage of this platform is not limited to only these classes of users, and the platform is more generally meant for use by the general public. The simple user interface will make it easy to use for all these classes of users.
The general classes of users are:

1. Customers: They create accounts in order to view movies.
2. The database administrator: He has the privilege to update information in the database.

2.2.4 Operating Environment

- Operating system: Windows / MAC / Linux
- Database: MySQL + MongoDB
- Client / server system
- Internet speed: Minimum of 500 kbps
- RAM: Minimum 1GB for streams up to 720p, minimum 2GB for streams up to 1080p.

2.2.5 Design and Implementation Constraints

Following could be the constraints on the design and implementation of the proposed software:

1. The reviews written by the users will be analyzed using NLP, in order to make movie recommendations to the users. This might not give 100% accuracy of predictions.
2. Storage of huge data.
3. SQL commands and managing the database.
4. Video streaming constraints.
5. Accessing of data with low speed internet.

2.2.6 User Documentation -

1. https://docs.mongodb.com/
2. https://dev.mysql.com/doc/
3. https://reactjs.org/docs/getting-started.html
4. https://ieeexplore.ieee.org/document/7522526

2.2.7 Assumptions and Dependencies

It is assumed that the database design for this platform will work appropriately. It is also assumed that the speed of the internet used by the customer is efficient to load data from the database and the server, to the user system or browser, and the browser is able to handle these huge amounts of data.

2.3 External Interface Requirements

2.3.1 User Interfaces

The system shall provide a simple and easy to use Graphical User Interface. The user interface shall consist of a main home page, which links to other pages. This improves ease of use for the user, and allows the user to navigate the platform easily. This allows for higher software flexibility as well. This will appeal to both inexperienced and experienced computer users, with a higher selling market.

The following are the features the system shall provide:

1. Menu bar: This offers basic features such as searching for a movie. The user can type the name of the movie into the search bar, in order to search for a movie, and the user can click the movie which appears in the search result, in order to watch it. Other basic features like viewing profile information, updating profile information etc. are also provided.

2. When the user chooses a movie to watch, the display switches to the next screen, and the movie begins playing there. Here, the user can also rate movies and write reviews. Once the user writes a positive review for a movie, he can navigate to his recommendations page, where he can see a list of similar movies recommended to him.

3. Purchase subscription/movie: The user will be able to purchase a monthly subscription, or purchase a movie which is not available on the streaming platform.

2.3.2 Hardware Interfaces

The external hardware interface used for accessing the online movie streaming platform is the personal computer of the user. The PCs may be laptops, desktops, notebooks or netbooks with an internet connection. The minimum connection speed required is 500 kbps. A minimum of 1GB RAM is recommended for streams up to 720p and 2GB RAM for streams at 1080p. Since the application must run over the internet, all the hardware shall require to connect to the internet will constitute the hardware interface for the system. For eg. modem, WAN – LAN, Ethernet Cross-cable.

2.3.3 Software Interfaces

The software requirements at the user's side are:

1. Windows 7 or higher or MAC OS X 10.3 or higher or Linux
2. Flash player 7.0 + plug-in
3. Google Chrome or Mozilla Firefox browser
4. Internet connection with a minimum speed of 500 kbps.

The following are the interfaces provided:

1. The online movie streaming platform shall communicate with providers, distributors, producers and creators to acquire licensing to stream TV shows and movies.
2. The system shall communicate with systems like Paytm, in order to validate and process payments. The option of a credit or debit card transfer can also be made available.
3. The system shall communicate with a third party software to allow users to complete secured transactions.
4. The system shall communicate with MySQL and MongoDB to access the data regarding the movies and user reviews stored in these databases.

2.3.4 Communications Interfaces

The e-store system shall use the HTTP protocol for communication over the internet. The intranet communication will be through TCP/IP protocol suite.

2.4 System Features

This section demonstrates prominent features of the project and explains how they can be used and the results they will give back to the user.

2.4.1 Account creation, login and authorization

Description and Priority

The first step to access this application is to create an account and login. Therefore this is a high priority task.

Stimulus/Response Sequences

When the user opens this application for the first time, he will be prompted to either sign up or login. He can also update his profile information on logging in.

Functional Requirements

1. The user should be able to create an account by specifying a username, password, email id and mobile number.
2. If any of the details provided are incorrect, or if the username already exists, or the password is too weak, or if there is a password mismatch, the user must be prompted with a suitable error message, and must be prompted to enter the correct details.
3. The user must be able to login using the username and password provided by him. If an incorrect username or password is typed, the user must be prompted with a suitable error message, and must be prompted to enter the correct login details.
4. On the completion of the login, the user must be taken to the home page, where he can purchase a subscription, purchase a movie, update his profile information, search for movies etc.

2.4.2 Subscription plan

Description and Priority

The user should be able to purchase or renew his monthly subscription plan. This is one of the relatively high priority tasks, since having a subscription plan is important for the user to be able to watch movies and TV shows on the streaming platform. However, a subscription is not required for the user to purchase a movie.

Stimulus/Response Sequences

If the user is a new user, he is prompted with an option to purchase a monthly subscription plan. He need not purchase a plan if he just wants to purchase a single movie. If the user chooses to purchase a subscription plan, he is then prompted to make a payment via Paytm. A confirmation message is displayed once the payment is made.

Functional Requirements

1. A user logging in for the first time should be prompted to purchase a monthly subscription plan.
2. A user whose subscription is about to expire, should be prompted to renew it 2 days before it expires.
3. In both cases, the user must be able to make a payment via Paytm, and a confirmation message should be displayed if the payment is successful. An error message should be displayed if the transaction fails.

2.4.3 Movie search and streaming

Description and Priority

The user should be able to search for a movie or TV show by typing the movie name into the search box. He should then be able to watch the movie or TV show. This is also a high priority task, since the entire platform revolves around streaming movies and TV shows. The functionality of purchasing an available movie does not require a subscription plan, and it is a low priority task.

Stimulus/Response Sequences

The user should see a few movies which he can watch, on the home page, when he logs into his account. A search bar present at the top of the page should allow him to search for. The user should be able to watch any movie he chooses. If the movie is unavailable, a message should be displayed, informing him of which website the movie is available on.

Functional Requirements

1. A search bar present on the home screen shall allow the user to search for movies by movie name.
2. When an available movie is clicked on, the user should be able to watch the movie
3. In case a movie is unavailable, a message must be displayed, informing the user of which website the movie is available on, and giving him an option to purchase it.
4. The user should also be able to download a movie and watch it when he has no internet connection.

2.4.4 Movie recommendation

Description and Priority

The user is recommended movies or TV shows he might like based on how he rates movies, and also the reviews he writes. This will be implemented using NLP. This is of relatively lower priority, because although this is a unique feature of our system, it is not crucial to the functioning of the system. Also, it needs the other features to be completed before it can work.

Stimulus/Response Sequences

After the user has watched and rated a few movies, and also written a few reviews, recommended movies start showing up in his recommended movies list on the recommendations page. The recommendations are further tuned according to the user's preferences depending on how he further rates movies.

Functional Requirements

1. The user should be able to see a recommended movies list on his recommendations page..
2. In case the user is new, and hasn't seen any movies, he can be recommended movies which have a high IMDB rating, and his recommendations can be tuned further to his preferences as he keeps watching more movies.
3. The user must have the option of rating movies and writing a review about it.

2.5 Other Nonfunctional Requirements

2.5.1 Performance Requirements

The product shall be based on web and has to be run from a web server.
The product shall take initial load time depending on internet connection strength, and the quality of the video being streamed.
The performance shall, to a large extent, depend upon hardware components, internet connection and RAM of the user.
The system must be interactive and the delays involved must be minimal. The delays in case of opening windows forms, of popping error messages and saving the settings or sessions must be minimized. Even operations like opening databases, sorting questions and evaluation must not have significant delays.

2.5.2 Safety Requirements

Consent must be taken from every user creating an account that he is over 18 years of age, or has a parent or guardian's consent. Also, movies must be provided age ratings, so that children don't end up watching violent movies. These are required to ensure the safety of the user.

2.5.3 Security Requirements

The username and password of every user must be stored in a database. However, the password must be stored in a hashed form to ensure security. Other information such as the email id and mobile number of the user must also be kept confidential. Also, when purchasing a movie or subscription, it must be ensured that the user's transaction is secure.

2.5.4 Software Quality Attributes

The software must be reliable and available. While it is not a safety critical application, it must still be available throughout the day. Both experienced and inexperienced computer users must be able to use the application easily, and its user interface must be simple enough for anyone to understand.

2.5.5 Business Rules

Any application that stores confidential information of the users will have some rules regarding who can access what. Only authorized users (the administrators of the application) must be able to access the database. Most of the processes, however, must be automated.

2.6 Other Requirements

Database requirement: The application must be able to store every format of data. The application should be frequently connectable to the database.

Appendix A: Glossary

- **Python:** Python is an interpreted, high-level, general-purpose programming language
- **Web framework:** A web framework (WF) or web application framework (WAF) is a software framework that is designed to support the development of web applications
- **MySQL:** MySQL is an open-source relational database management system
- **MongoDB:** MongoDB is a cross-platform document-oriented database program. Classified as a NoSQL database program.
- **ML:** ML is an application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed.
- **NLP:** It is a subfield of linguistics, computer science, and artificial intelligence concerned with the interactions between computers and human language, in particular how to program computers to process and analyze large amounts of natural language data.
- **Test Suite:** It is a collection of test cases of a particular component or unit

Chapter 3

Design

3.1 Overview

After reviewing the Use Case analysis, following are the basic classes and actions that emerge:

Classes: (Basic building blocks of Online Streaming Platform (Movieholic))

Sl no.	Class	Principle Responsibility
1	User	Avail functionalities of the platform to watch movies and TV shows.
2	Movie	To be streamed to users for them to watch.
3	Review	To contain information about users' reviews and ratings for various movies.

Note: *Other subsidiary classes may get added to the list in course of implementation for the purpose of load balancing and modularity.*

Actions:

Sl. no.	Action
1	Watch movie
2	Search for movie
3	Validate user
4	Write review
5	Modify user details
6	View recommendations

Note: *There are other minor actions that does not play major role in modeling.*

3.2 System Structure

Here we describe the final structure. It should, however, be kept in mind that the obtaining the final structure is an iterative exercise – an initial structure is refined as the design progresses. In particular, the dynamic modeling has an impact on the structure.

3.2.1 Inheritance Structure

There is one evident case of an inheritance structure that is between the types of videos that the user can stream. The video being streamed can either be a movie or TV Show.

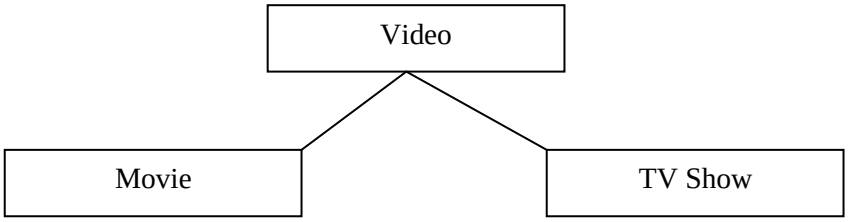


Figure 3.1: Inheritance structure

Both movies and TV shows have most of their features in common, but have a few different attributes like duration.

3.2.2 Entity Relationship Diagram for Movieholic

ER diagram: An Entity–Relationship diagram is a high-level conceptual data model diagram. It is based on the notion of real-world entities and the relationships between them. ER modeling helps in analyzing data requirements systematically to produce a well-designed database. So, it is considered a good practice to complete ER modeling before implementing the database.

Figure 2 shows the entity-relationship diagram for our project. The entities (described in rectangles) are the customer user account, user profile, movie, review, subscription and payment. All these entities have attributes (described in ovals), with one attribute being the key attribute. The relationships between these entities are described using diamonds, and the multiplicity of these relationships is also shown.

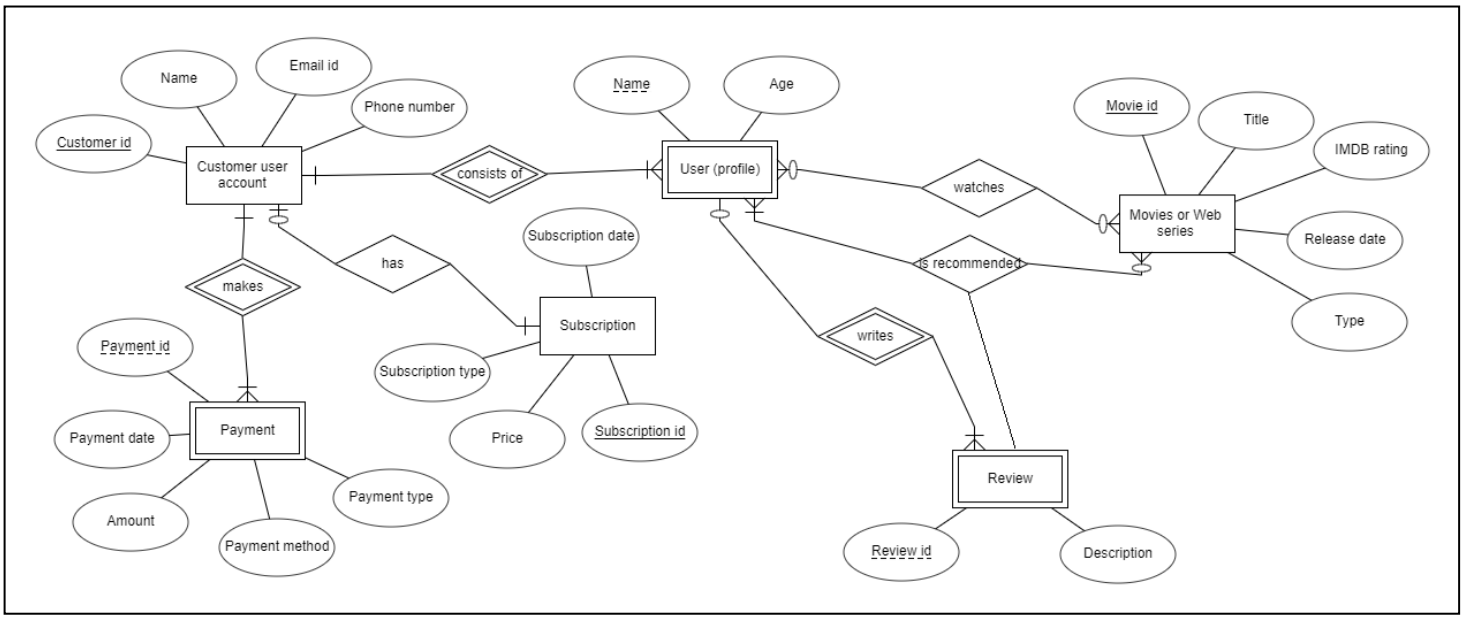


Figure 3.2: ER Diagram

3.2.3 Complete class diagram

Finally, after considering all the major actions, the complete association and aggregation structure is arrived at.

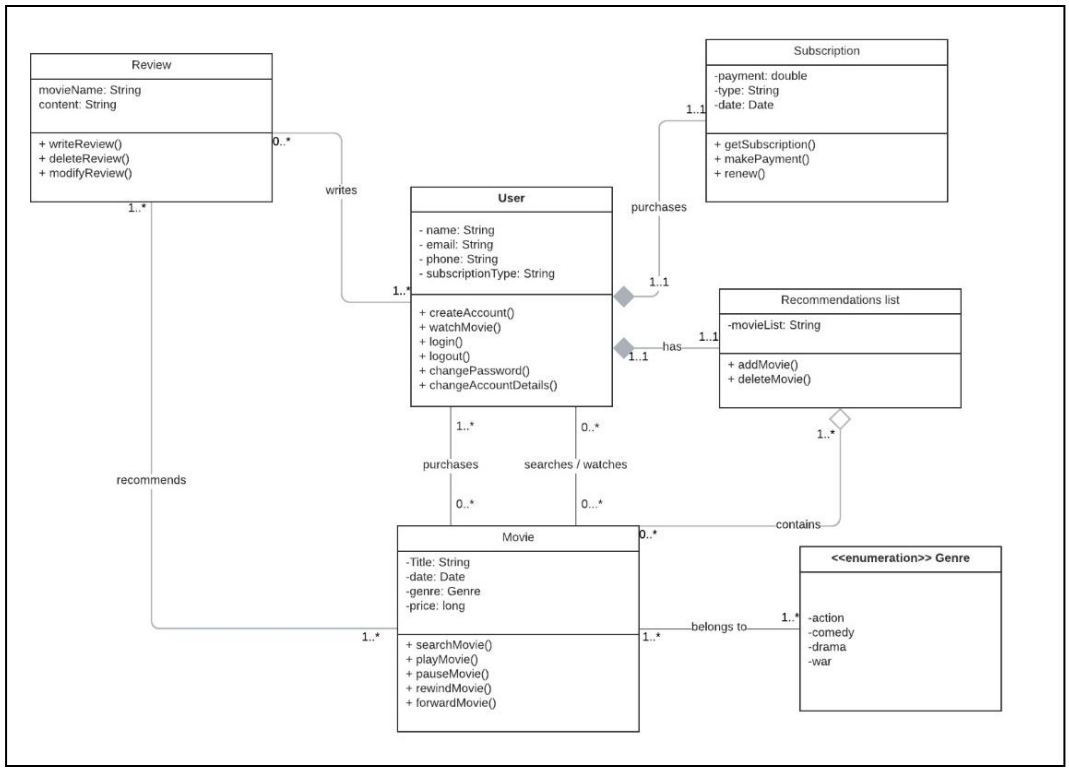


Figure 3.3: Class diagram

3.2.4 Structure Chart

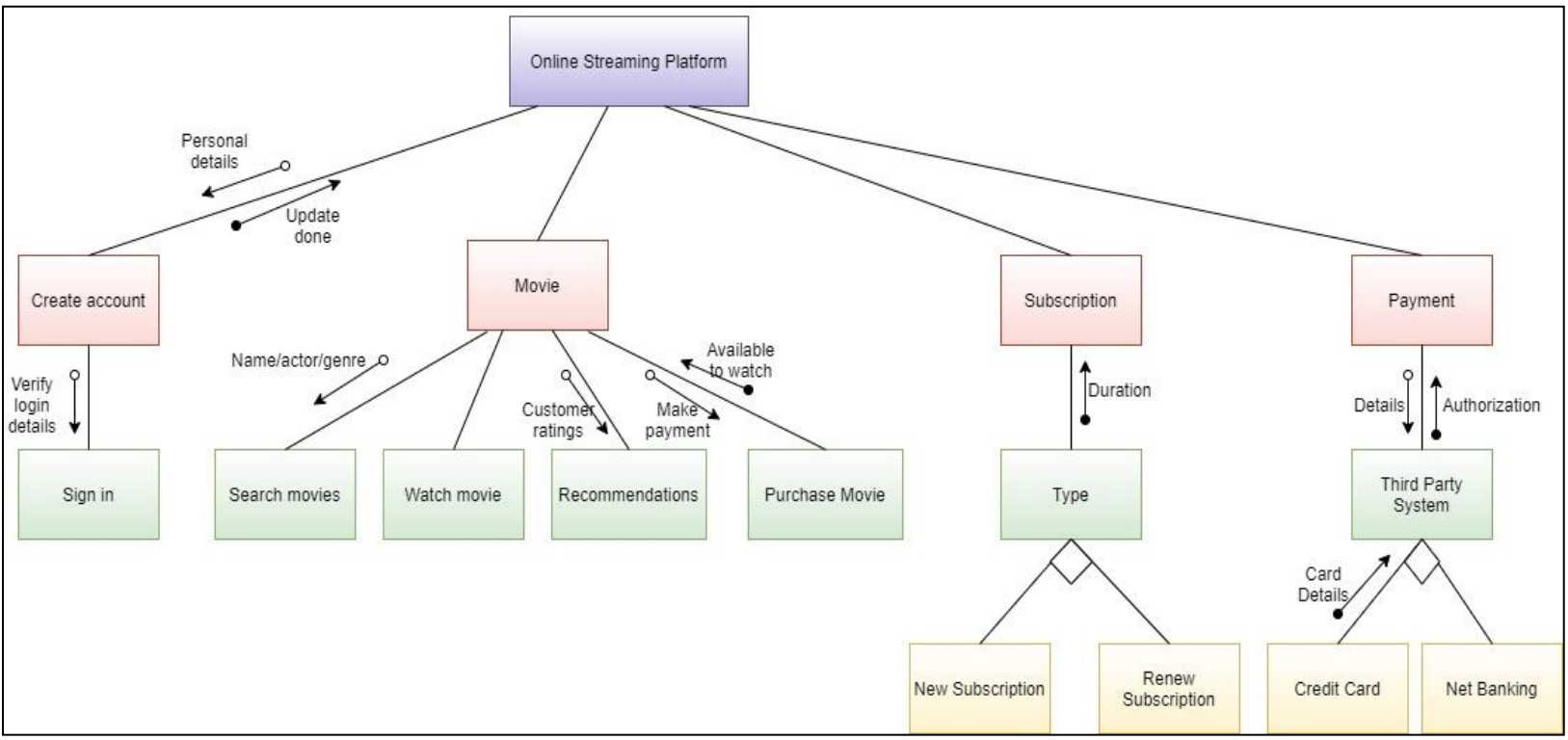


Figure 3.4: Structure chart

3.3 System behavior

3.3.1 Sequence Diagram

The dynamic behavior of the system is modeled by representing the interactions between the classes involved in each principal action. We are showing the final diagrams here. It should be remembered that these models have an impact in refining and enhancing the class diagrams – we are not discussing these aspects here.

Registration module

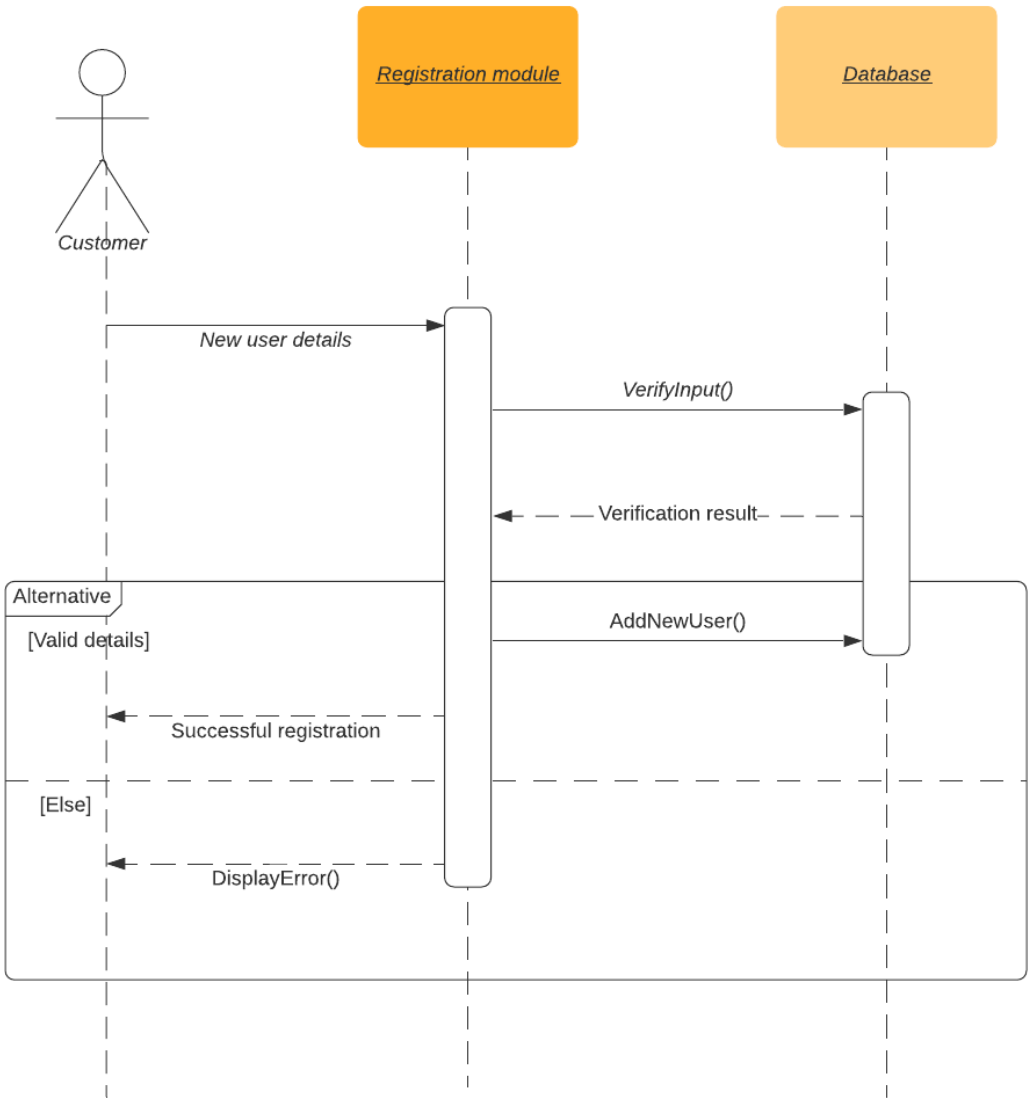


Figure 3.5: Sequence diagram – registration module

Login module

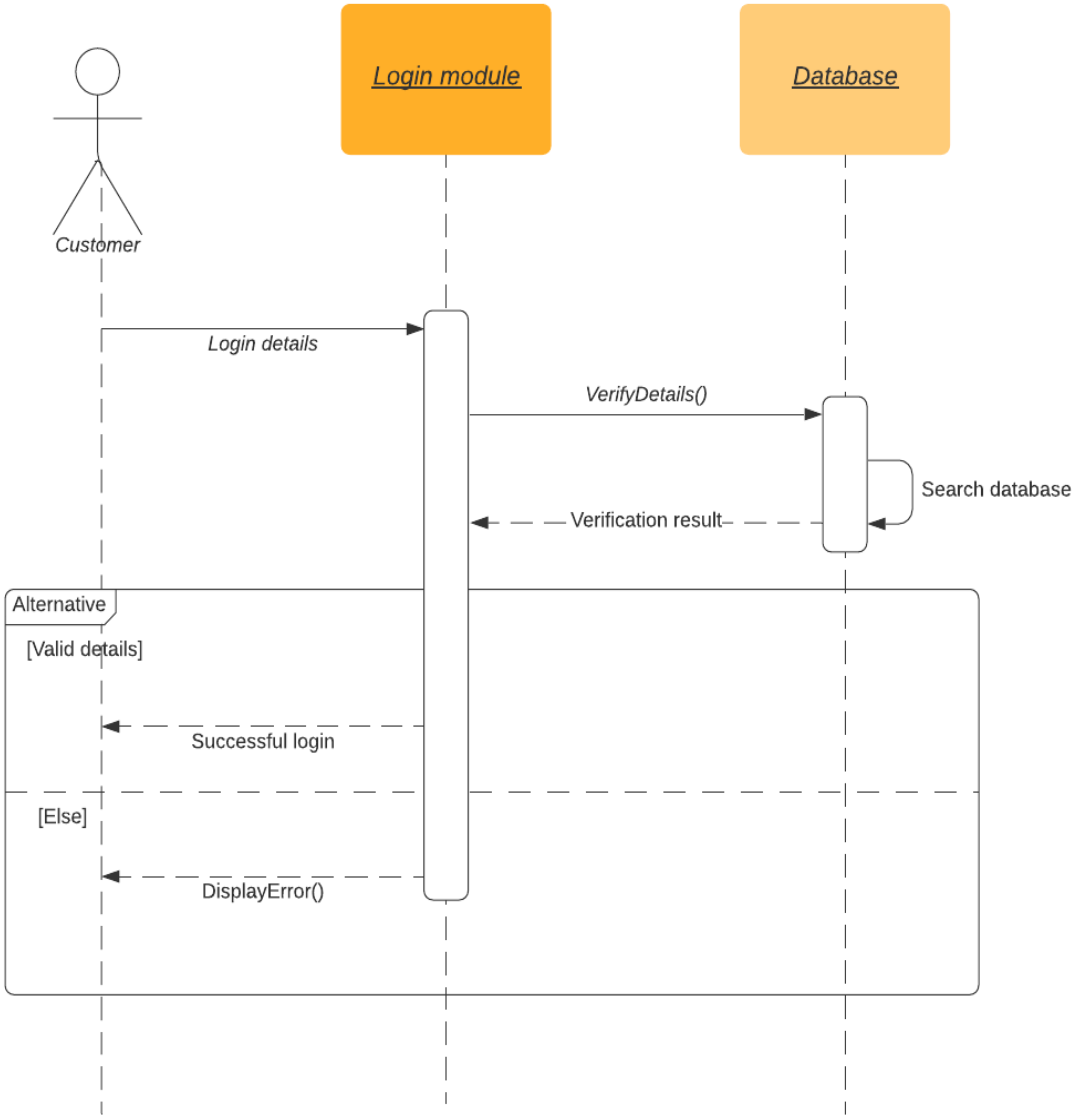


Figure 3.6: Sequence diagram – login module

Subscription/purchase movie module

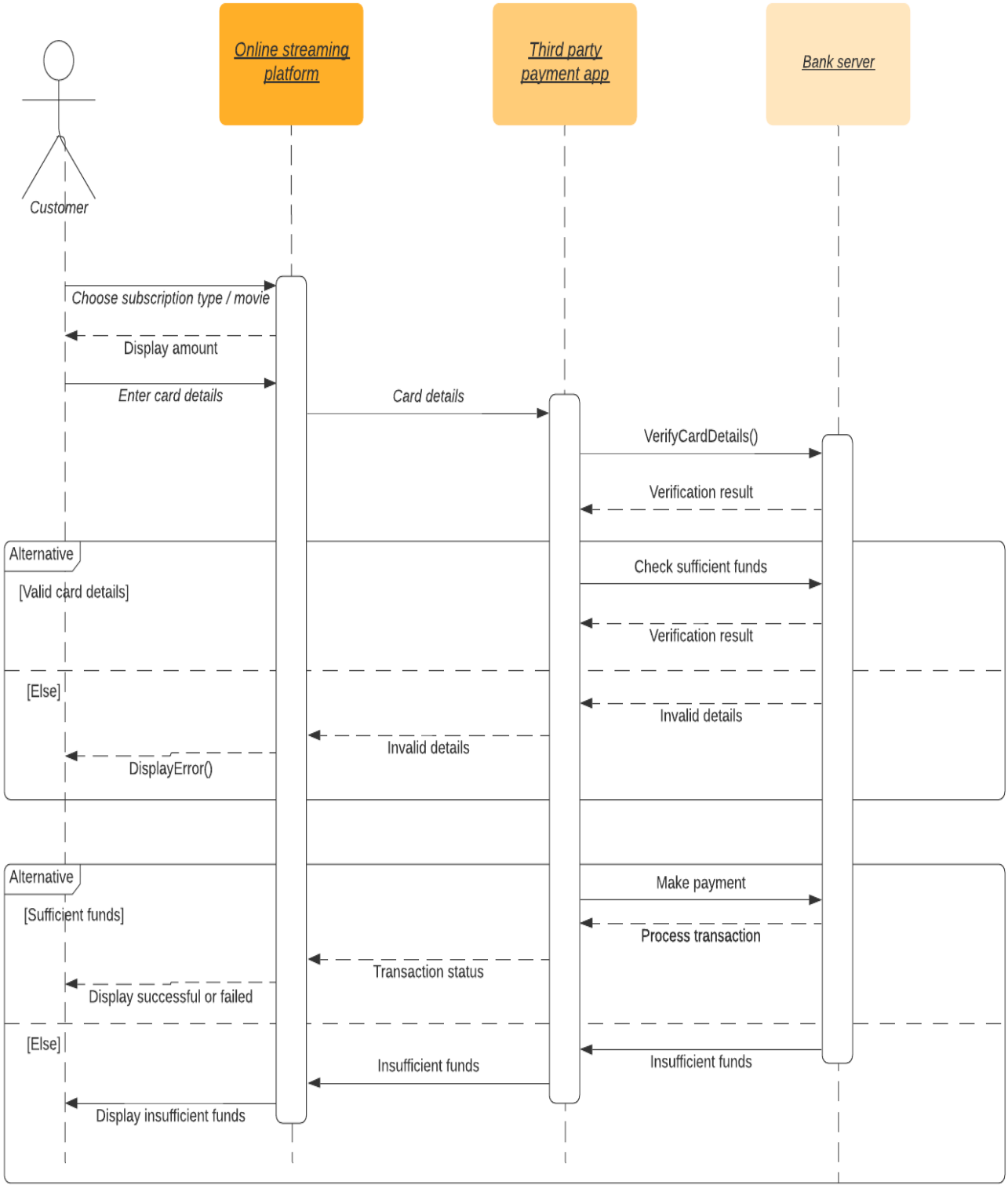


Figure 3.7: Sequence diagram – subscription/purchase movie module

Watch movie module

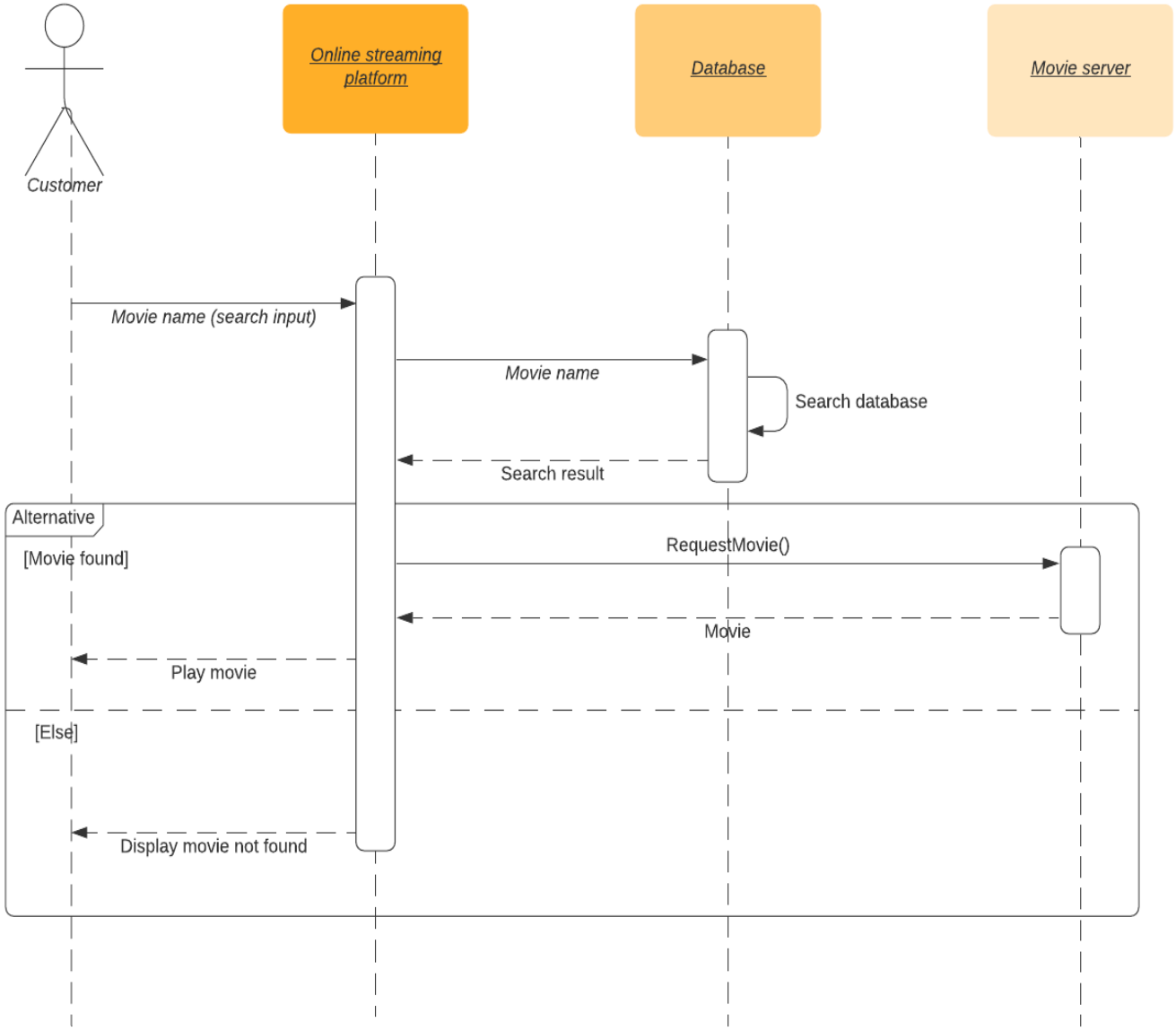


Figure 3.8: Sequence diagram – watch movie module

Review/recommendation module

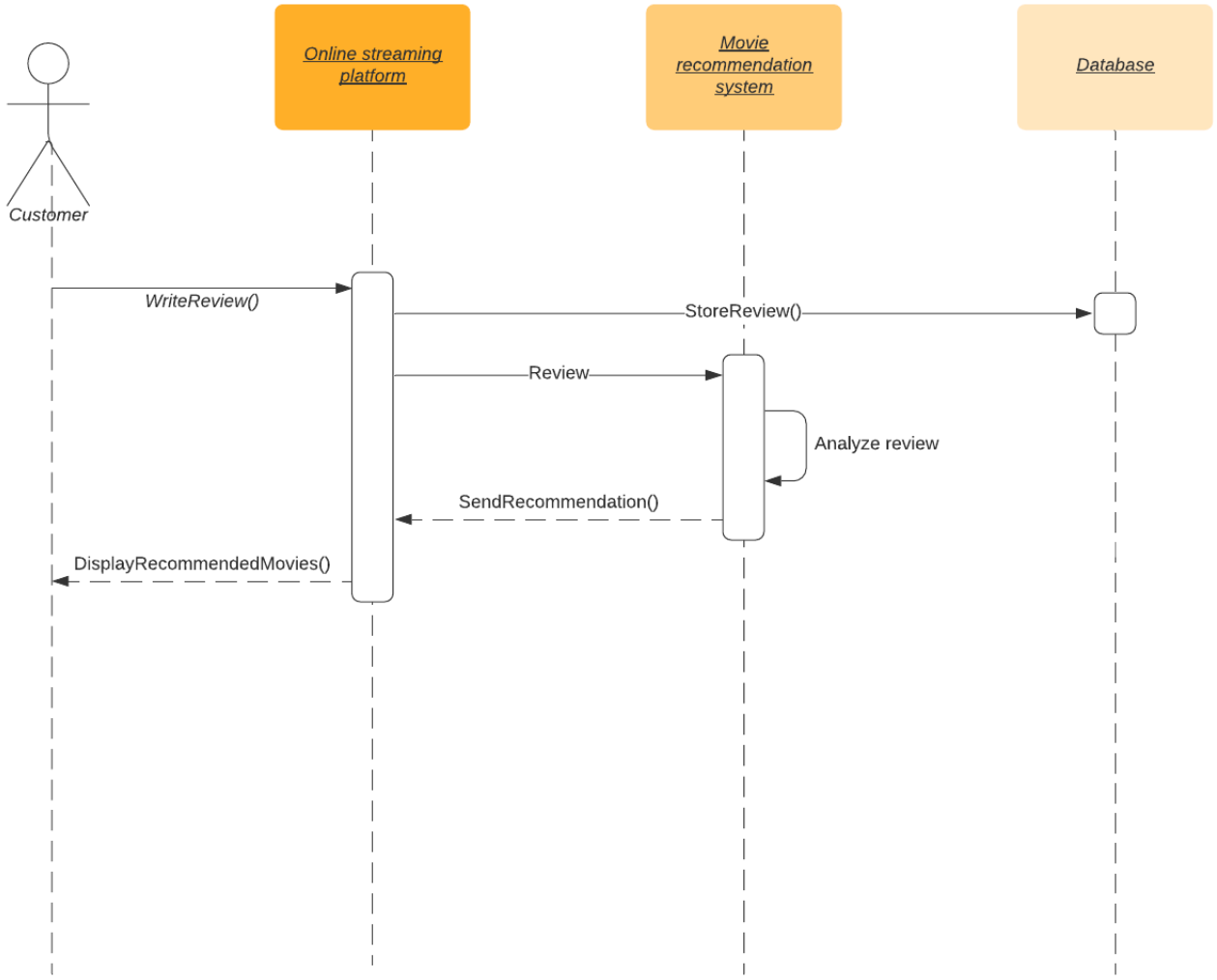


Figure 3.9: Sequence diagram – review/recommendation module

3.3.2 Data flow diagram

Such a diagram represents the dynamic flow of data in the system, between various entities, processes and data stores.

Level 0 DFD

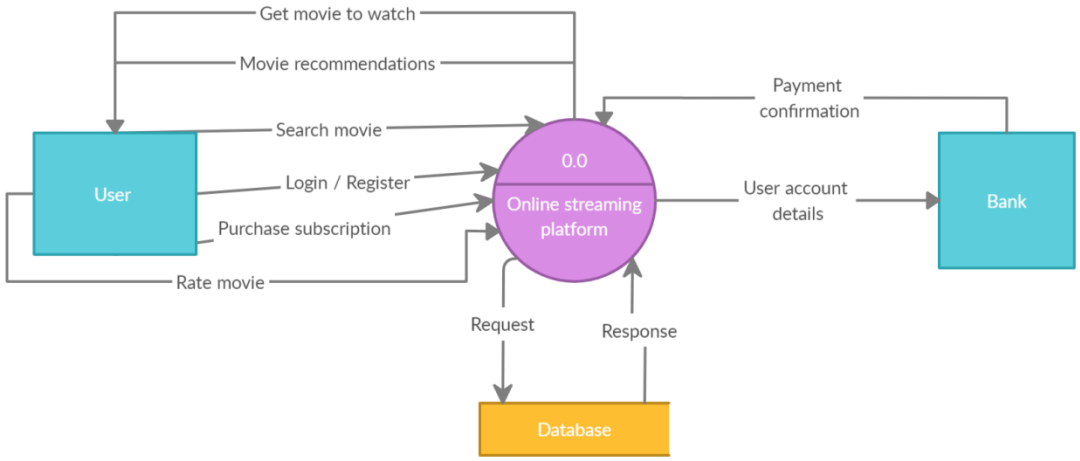


Figure 3.10: Level 0 DFD

Level 1 DFD

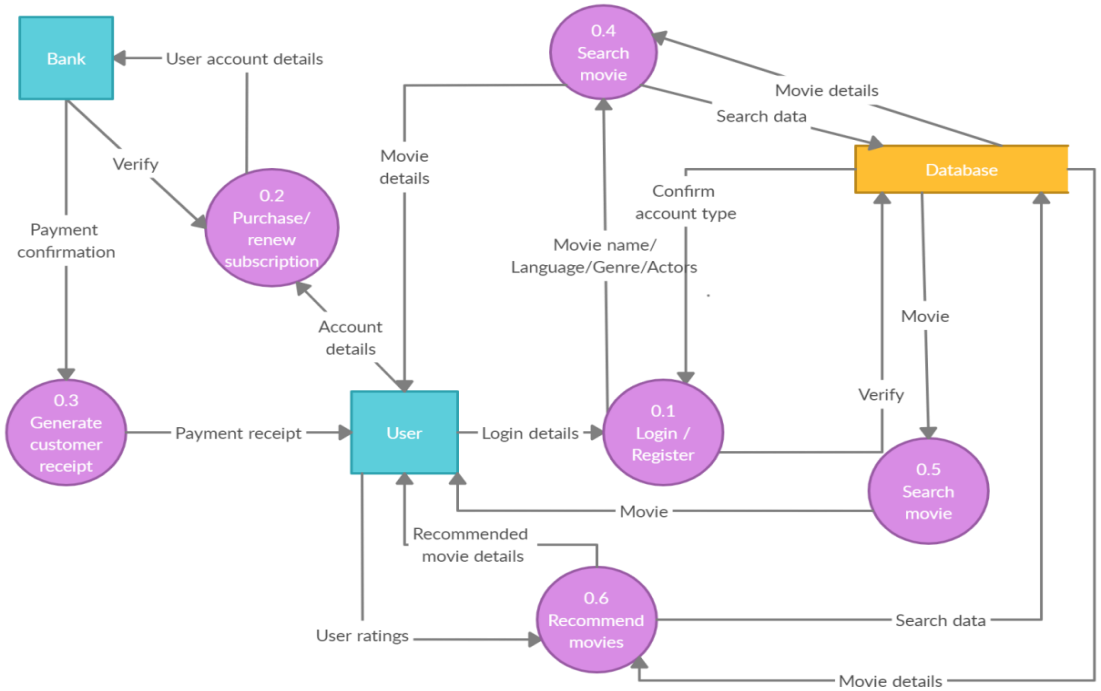


Figure 3.11: Level 1 DFD

3.4 Detailed Design Specification:

It consists of a list of main classes and their attributes and methods.

1. class **User**{
   //**attributes**//
   String name;
   String password;
   String email;
   String phone; //Mobile number
   String subscriptionType;
   //**methods**//
   void createAccount()
   void login()
   void logout()
   void watchMovie()
   void changePassword()
   void changeAccountDetails()
   }

2. class **Movie**{
   //**attributes**//
   String title;
   Date date;
   Long price;
   //**methods**//
   void searchMovie()
   void playMovie()
   void pauseMovie ()
   void forwardMovie()
   void rewindMovie()
   }

3. class **Review**{
   //**attributes**//
   String movieName;
   String content;
   //**methods**//
   void writeReview()
   void deleteReview()
   void modifyReview()
   }

4. class **Subscription**{
   //**attributes**//

Double payment;
String type;
Date date;
//**methods**//
void getSubscription ()
void makePayment()
void renew()
}

5. class **RecommendationsList**{
//**attributes**//
String movieList;
//**methods**//
void addMovie()
void deleteMovie()
}

3.5 Use Cases:

The complete use case diagram describes the entire system as a whole, and how the various actors interact with the system. It is as shown below:

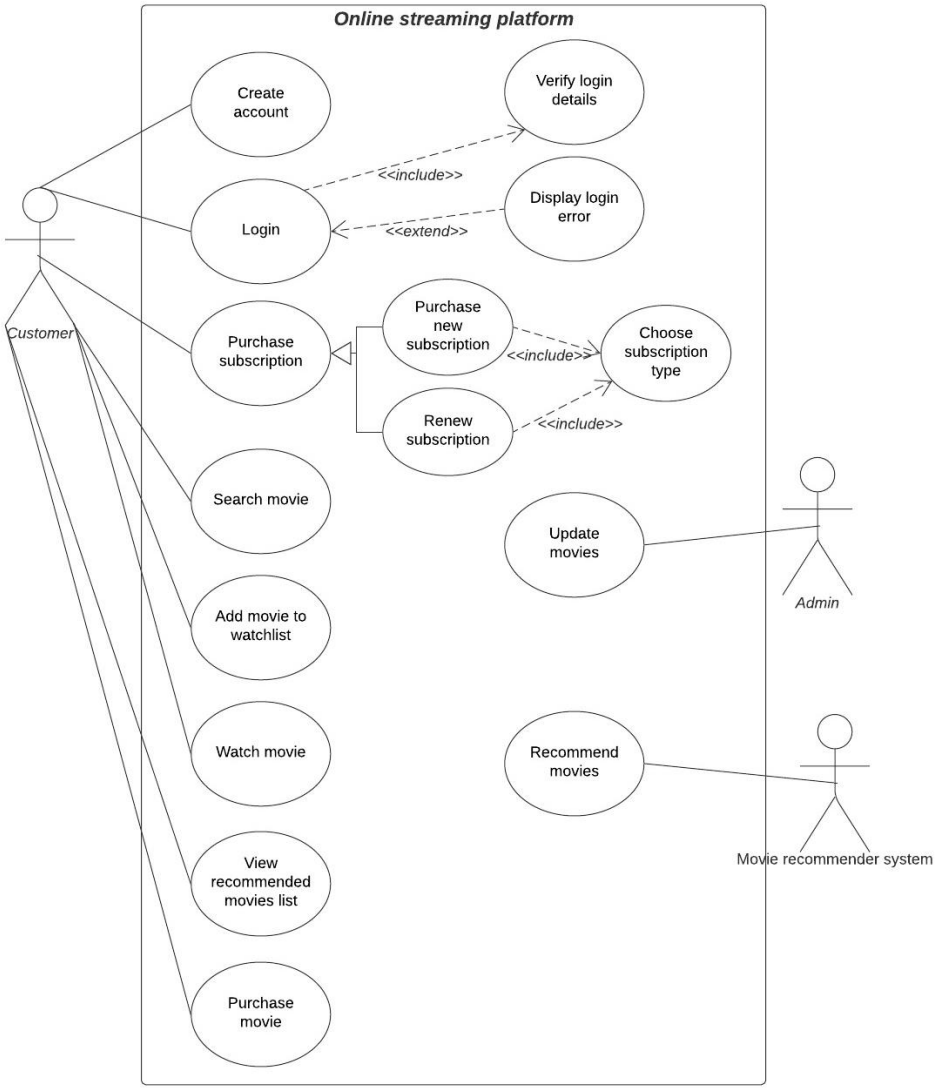


Figure 3.12: Use case diagram of the complete system

Each use case is also described module wise, along with a use case diagram for each module.

3.5.1 Use case 1: Login and purchase subscription

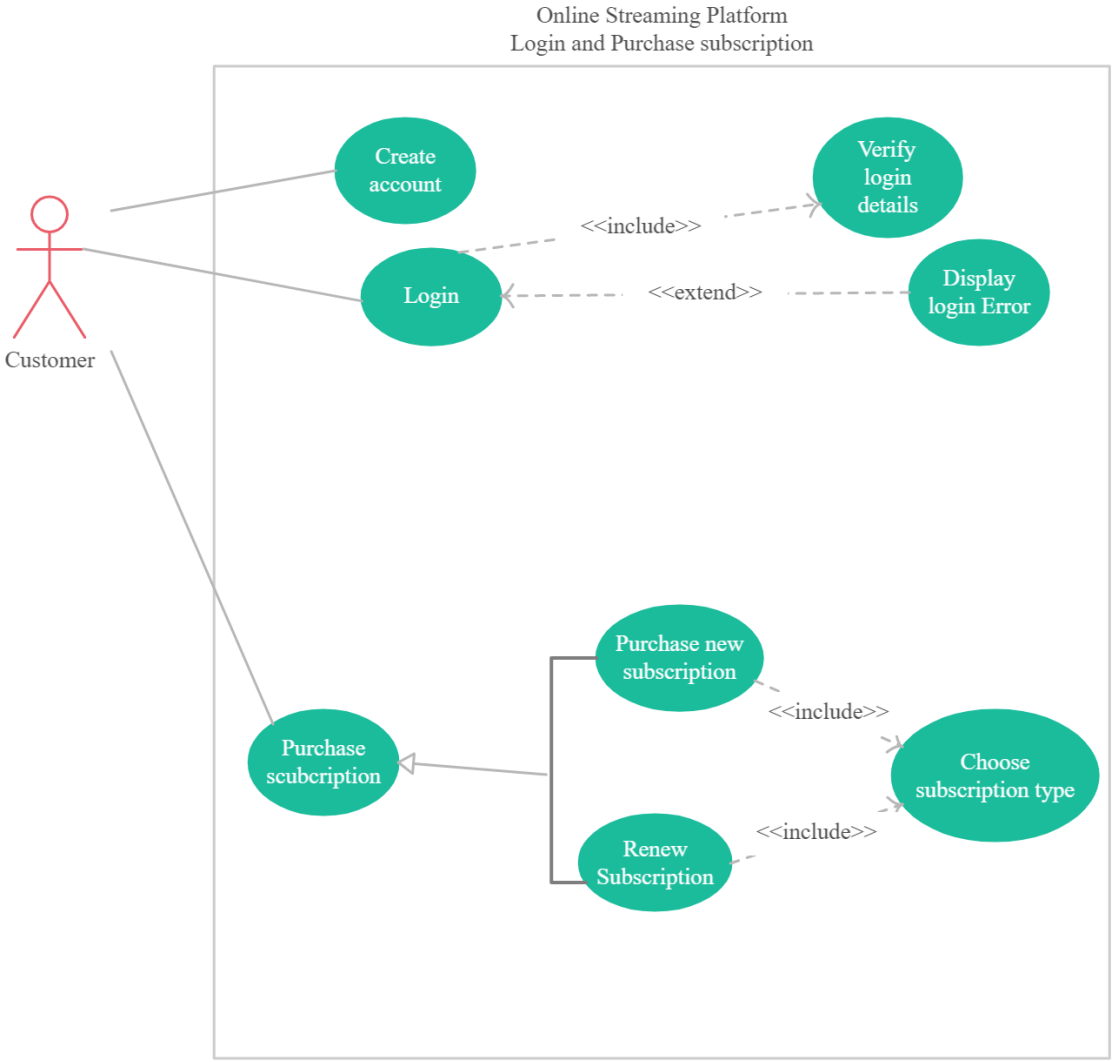


Figure 3.13: Use case diagram for login module

Use case description:
Use case 1: Login and Purchase Subscription
Primary actor: Customer
Goal:

- Login/Create account
- Purchase subscription

Main Success Scenario:

1) Customer creates account
2) Customer logs in
3) Verifies the log in details
4) Customers is provided with different subscription plans
5) Customer chooses a new plan and makes the payment
6) Or customer renews his/her subscription plan

Alternatives:

- 2a: System displays login error
    - ➢ Customer is given option to re-enter the login credentials

3.5.2 Use case 2: Streaming movies

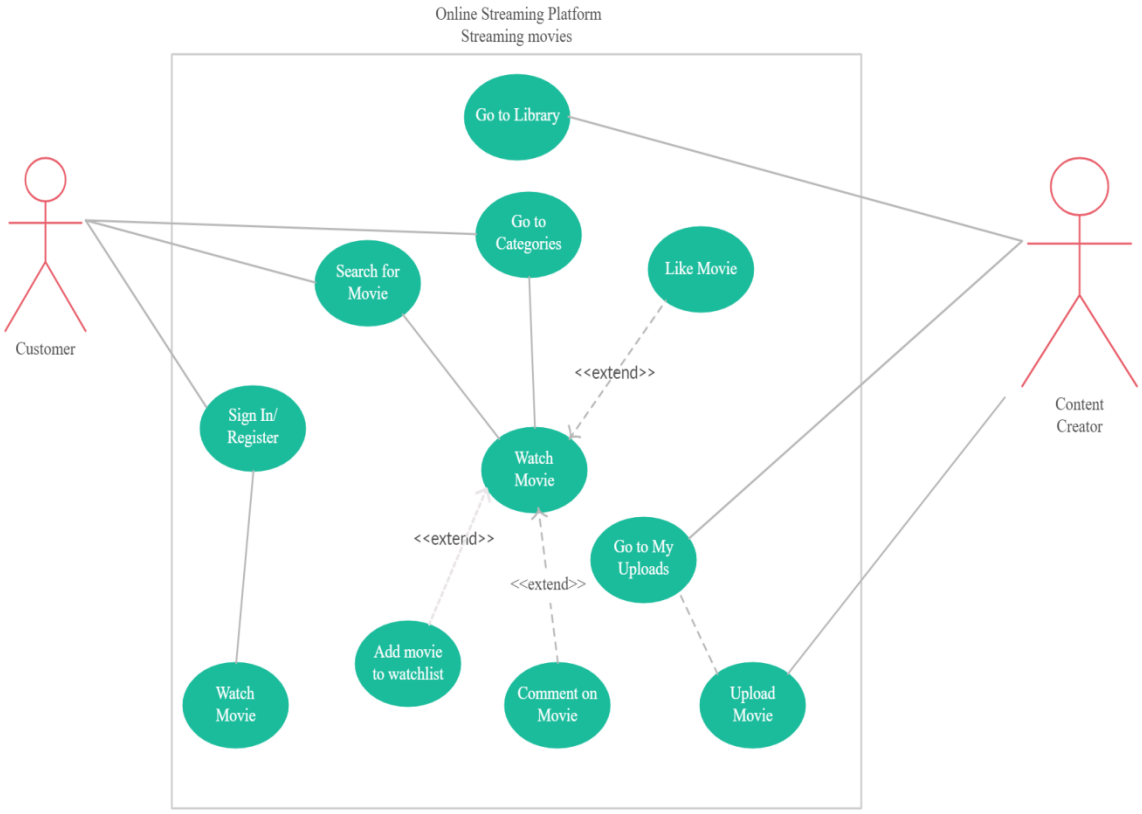


Figure 3.14: Use case diagram for streaming module

Use case description:
Use Case 2: Streaming Movies
Primary actor: Customer
Goal:

- Search movies
- Watch movies
- Add movie to watch list
- Comment and like movies

Precondition: User has logged in

Main Success Scenario:

1) Customer searches for a movie to watch
2) Customer goes to categories to look for movies
3) Customer adds the movie to watch list
4) Customer watches the movie
5) Customer comments on the movie

6) Customer rates the movie
7) Content creator uploads new movies

Alternatives:

- 3a: Website does not let the user add the movie
- 4a: Internet speed is low, movie starts buffering
  - ➢ Customer connects to high speed WiFi to continue watching the movie
- 5a: Comments are disabled for the movie
- 6a: Customer has not been provided the option to rate the movie

3.5.3 Use case 3: Purchase movie

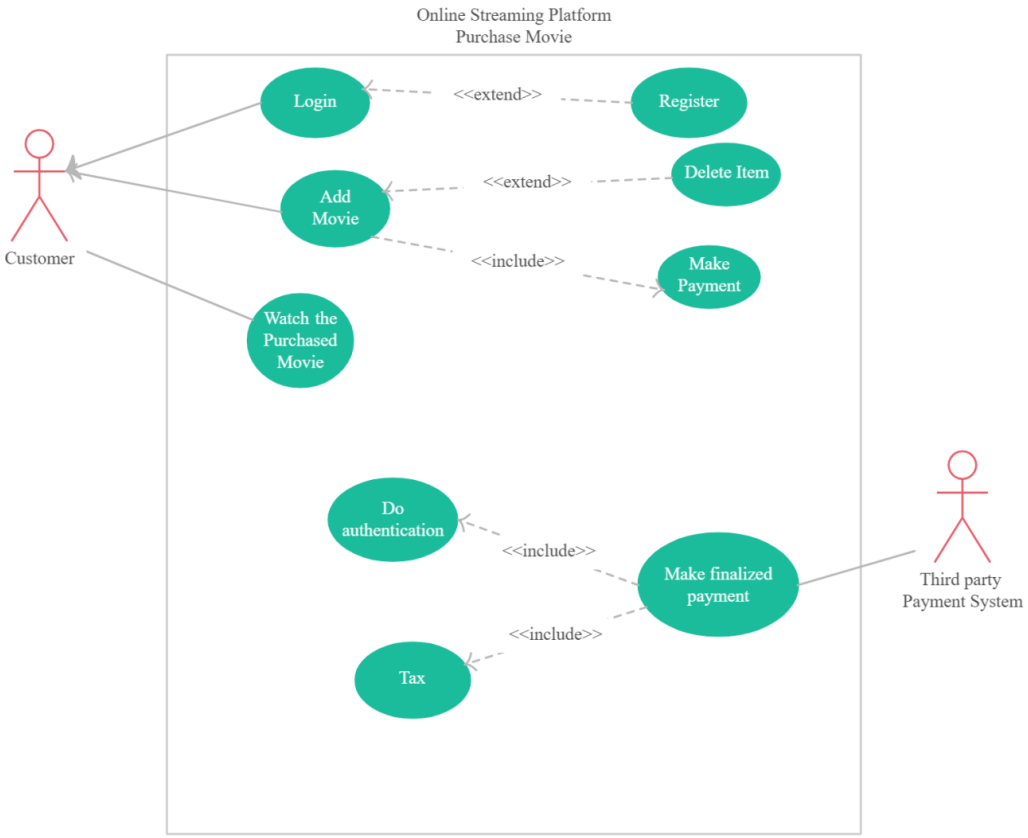


Figure 3.15: Use case diagram for purchase module

Use case description:
Use Case 3: Purchase Movies
Primary actor: Customer
Goal: Purchase a movie which was not available on the website and make payment
Precondition: User has logged in

Main Success Scenario:

1) Customer adds the movie in the cart to be purchased
2) Customer deleted the movie item in case of a change of mind
3) Customer makes payment for the movie to be purchased
4) Third party system does authentication for a safe transaction
5) Tax is added to the final payment
6) After payment is done, customer watches the purchased movie

Alternatives:

- 1a: Unable to add movie
- 2a: Unable to delete the movie
- 4a: Credit Card authorization fails
    - Allows customer to re-enter information
- 6a: Movie has not been made available

Chapter 4
Implementation Plan

4.1 Management Overview

4.1.1 Description of Implementation

The team applied an Agile SDLC Methodology, which is a combination of iterative and incremental process models, with a focus on process adaptability and improvement by rapid delivery of working software application. An incremental approach was adopted, and the working software was built one feature at a time. The implementation process obeyed the following Agile Manifesto principles:

- **Individuals and Interactions -** In Agile development, self-organization and motivation are important, as are interactions for pair programming.
- **Working Software -** Demo working software was developed to analyze and understand the requirements better instead of just depending on documentations.
- **Customer Collaboration -** As the requirements cannot be gathered completely in the beginning of the project due to various factors, continuous self-validation and team interaction is very important to get proper product requirements.
- **Responding to change -** Agile development is focused on quick responses to change and continuous development/improvement.

4.1.2 Major Tasks

This section provides a description of the major system implementation tasks:

- Setup database configuration and connectivity
- Login module: This allows existing users to login and new users to create an account. New users are prompted to purchase a subscription.
- Subscription module: The user is prompted to subscribe to either a monthly or yearly plan. Users are also prompted to renew their subscriptions 2 days before it expires.
- Movie search and streaming: This allows users to search for movies by name, language, genre and actors, and watch the movie. Users are given the option of purchasing unavailable movies from other websites.
- Movie recommendation: Personalized movie recommendations are made to users based on how they rate movies, and the reviews they write. New users are recommended movies with high IMDB ratings.

- Run required tests and security implementation

4.1.3 Security and Privacy

This section provides an overview of the system security and requirements that must be followed during implementation:

- Multiple validations are in place when the user is creating his account and logging in, so it prompts the user to enter a strong password, and the validity of his details is also verified.
- Password storage is done in a one way hashed method, so that sensitive information is not directly stored in the database.

4.2 Implementation Support

This section describes any support hardware, software, facilities, and materials required for the implementation, along with the documentation, necessary personnel and training requirements, outstanding issues, and implementation impacts to the current environment.

4.2.1 Requirement Specification

4.2.1.1 Hardware

This section provides a list of support equipment: This hardware may include computers, servers, peripheral equipment, simulators, emulators, other non-computer equipment as well as any network and data communication requirements.

- Intel i5-4590 processor equivalent or greater
- 8GB+ RAM
- Compatible HDMI 1.3 video output

4.2.1.2 Software

This section provides a list of non-hardware components (software, databases, and compilers, operating systems, utilities, etc.) required to support the implementation.

- Operating system: Windows 7 and above
- MySQL database
- MongoDB
- HTML, CSS, Javascript

- PHP

4.2.1.3 Database Requirements

- MongoDB
- MySQL

4.2.1.4 Data Requirements

- Details of Users registered.
- Details of star ratings and users' reviews.
- Details of movies watched.

4.2.1.5 Functional Requirements

A functional requirement defines the functionality of a system or one of its subsystems. It also depends upon the type of software, expected users and the type of system where the software is used. The functional requirements of the proposed work are as follows:

1. **Account creation, login and authorization**

   The user should be able to create an account by specifying a username, password, email id and mobile number. If any of the details provided are incorrect, or if the username already exists, or the password is too weak, or if there is a password mismatch, the user must be prompted with a suitable error message, and must be prompted to enter the correct details.

2. **Subscription plan**

   A user logging in for the first time should be prompted to purchase a monthly subscription plan. A user whose subscription is about to expire, should be prompted to renew it 2 days before it expires. In both cases, the user must be able to make a payment via Paytm, and a confirmation message should be displayed if the payment is successful. An error message should be displayed if the transaction fails.

3. **Movie search and streaming**

   A search bar present on the home screen shall allow the user to search for movies or TV shows by entering the movie name. When an available movie is clicked on, the

user should be able to watch the movie. In case a movie is unavailable, a message must be displayed, informing the user of which website the movie is available on, and giving him an option to purchase it. The user should also be able to download a movie and watch it when he has no internet connection.

4. **Movie recommendation**

   The user should be able to see a recommended movies list on his home page. In case the user is new, and hasn't seen any movies, he can be recommended movies which have a high IMDB rating, and his recommendations can be tuned further to his preferences as he keeps watching more movies. The user must have the option of rating movies and writing a review about it.

4.2.1.6 Non-Functional Requirements

The non-functional requirements of this project are:

- **Performance Requirements**

  The product shall be based on web and has to be run from a web server.The product shall take initial load time depending on internet connection strength, and the quality of the video being streamed.
  The performance shall, to a large extent, depend upon hardware components, internet connection and RAM of the user. The system must be interactive and the delays involved must be minimal. The delays in case of opening windows forms, of popping error messages and saving the settings or sessions must be minimized. Even operations like opening databases, sorting questions and evaluation must not have significant delays.

- **Safety Requirements**

  Consent must be taken from every user creating an account that he is over 18 years of age, or has a parent or guardian's consent. Also, movies must be provided age ratings, so that children don't end up watching violent movies. These are required to ensure the safety of the user.

- **Security Requirements**

The username and password of every user must be stored in a database. However, the password must be stored in a hashed form to ensure security. Other information such as the email id and mobile number of the user must also be kept confidential. Also, when purchasing a movie or subscription, it must be ensured that the user's transaction is secure.

- **Software Quality Attributes**

  The software must be reliable and available. While it is not a safety critical application, it must still be available throughout the day. Both experienced and inexperienced computer users must be able to use the application easily, and its user interface must be simple enough for anyone to understand.

- **Business Rules**

  Any application that stores confidential information of the users will have some rules regarding who can access what. Only authorized users (the administrators of the application) must be able to access the database. Most of the processes, however, must be automated.

4.2.2 Documentation

Following documentation is required to be delivered.

- Software requirements specification
- Use case Diagram
- ER Model
- Domain class Diagram
- Sequence diagram
- Data flow diagram
- Test coverage
- Test suite
- Design document

4.2.3 Societal Concern

In the current pandemic situation around the world, it is not advised to go to the movie theatres and our system allows the customers to watch the movies and TV shows that they

like without stepping out of their houses. It provides a wide variety of movie choices and also allows the users to buy a movie that might not be available on the website. In order to provide a more personalized experience to the customers, their ratings and reviews of the movies are taken into consideration and analyzed through NLP to provide more accurate recommendations based on their likes and dislikes.

4.3 Implementation Requirements by Site

4.3.1 Site Implementation Details

4.3.1.1 Implementation details:

- A NoSQL database (MongoDB) has also been made use of for this project. It is being used to store the star ratings and reviews provided by the user to movies or TV shows. A collection named reviews is created inside the database review, and is linked to the review form in the main website. Once the customer enters the review details and clicks submit, the reviews get stored in the reviews collection in the database.
- Both PHP and Python 3 have been used to interface to MongoDB. The PHP interface to MongoDB is for storing the review data in the database. The Python interface to MongoDB is for the purpose of retrieving the reviews and ratings one at a time, analyzing them, and recommending movies to users based on this analysis. The Python script executes inside PHP.
- The system has a friendly UI for the users to interact easily with the frontend of the website easily and to interact with the webpage.

4.3.1.2 Risks and Contingencies

Following risks are identified in the project:

- Breach of confidential data from the database (Except password, which is protected by a one way hashing mechanism)
- Server overload due to high traffic
- Server breakdown on DoS attack

4.3.1.3 Implementation Verification and Validation

Following functionality must be present for validation:

Authentication

- Registration system for users
- Authentication system for users.

User

- User must be able to update his/her contact details.
- User must be able to watch movies by clicking on them.
- User must be able to write reviews for movies.
- User must be able to view his recommended movies list after writing a positive review.
- User must be able to receive a confirmation message upon successful payment.

4.3.2 Acceptance Criteria

For transitioning the system into production, many of the requirements have to be filled which will lead to the smooth functioning of the system. Some of the criteria what we have identified in our project are described below:

- Server must be able to handle multiple requests within short periods of time
- Security provided to the customers should of very high quality.
- The website should be active continuously, with very little down time, to cater to the customers' needs.

APPENDIX A: Project Implementation Plan Approval

The undersigned acknowledge that they have reviewed the **Online Streaming Management System (Movieholic) - Implementation Plan** and agree with the information presented within this document. Changes to this **Project Implementation Plan** will be coordinated with, and approved by, the undersigned, or their designated representatives.

Signature: _____ Date: _____

Print Name: _____

_____

Title: _____

Role: Project Manager

Chapter 5

Testing

5.1 Introduction

The process or method of finding errors in a software application or program so that the application functions according to the end user's requirement is called software testing. Testing is evaluation of the software against requirements gathered from users and system specifications. Testing is conducted at the phase level in software development life cycle or at module level in program code.
A Test Plan is a detailed document that describes the test strategy, objectives, schedule, estimation, deliverables, and resources required to perform testing for a software product. Test Plan helps us determine the effort needed to validate the quality of the application under test. The test plan serves as a blueprint to conduct software testing activities as a defined process. Test Plan guides our thinking. It is like a rule book, which needs to be followed.

5.2 Purpose of the test plan document:

This test plan keeps track of possible tests that will be run on the system after coding plan describes the testing approach and overall framework that will drive the testing of the **Online Streaming Platform**. Following objectives are achieved through this document.

- **Test Strategy**: It describes the rules on which the test will be based on, including the givens of the project description of the process to set up a valid test (e.g.: entry / exit criteria, creation of test cases, specific tasks to perform, scheduling, data strategy).
- **Execution Strategy**: It describes how the test will be performed and process to identify and report defects, and to fix and implement fixes.
- **Test Management**: It is the process to handle the logistics of the test and all the events that come up during execution (e.g.: communications, escalation procedures, risk and mitigation.

5.3 Test Suites

5.3.1 Authentication

Test Scenario ID	Authentication 1	Test Case ID	Auth-1A
Test case description	Testing Authentication feature	Test Priority	High
Pre-Requisites	None	Post-Requisites	None

Sl no	Action	Inputs	Expected output	Actual output	Test browser	Test result
1	Sign up page: Enter first name, last name and username, email, confirm email, password, confirm password, Click Submit	Srishti Moorthy, sm2000, s@example.com, s@example.com, Testing999*, Testing999*	Redirect to home page	Redirect to home page	Chrome	Pass
2	Click logout button		Redirect to sign in page	Redirect to sign in page	Chrome	Pass
3	Sign in page: Enter username, password, Click submit	sm2000, wrongpwd	Error message displayed: username or password is incorrect	Error message displayed: username or password is incorrect	Chrome	Pass
4	Sign in page: Enter username, password, Click submit	sm2000, Testing999*	Redirect to home page	Redirect to home page	Chrome	Pass
5	Sign up page: Enter first name, last name and username, email, confirm email, password, confirm password, Click Submit	Srishti Moorthy, s@example.com, s@example.com, Testing999*, Testing999*	Error message displayed: Please fill in the username field	Error message displayed: Please fill in the username field	Chrome	Pass
6	Sign up page: Enter first name, last name and username, email, confirm email, password, confirm password, Click Submit	Srishti Moorthy, newusername, s@example.com, s@example.com, Testing999*, Testing999*	Error message displayed: This email has been registered before	Error message displayed: This email has been registered before	Chrome	Pass
6	Sign up page: Enter first name, last name and username, email, confirm email, password, confirm password, Click Submit	Mahek Jain, newusername, m@example.com, m@example.com, Testing999*, Testing99922*	Error message displayed: Passwords do not match	Error message displayed: Passwords do not match	Chrome	Pass
7	Sign up page: Enter first name, last name and username, email, confirm email, password, confirm password, Click Submit	Mahek Jain, newusername, m@example.com, mj@example.com, Testing999*, Testing999*	Error message displayed: emails do not match	Error message displayed: emails do not match	Chrome	Pass
8	Sign up page: Enter first name, last name and username, email, confirm email, password, confirm password, Click Submit	Mahek Jain, newusername, m@example.com, mj@example.com, aa, aa	Warning message should be displayed for weak password, only redirect to home page if user confirms	Redirect to home page	Chrome	Fail

5.3.2 Updating profile information

Test Scenario ID	Updating profile information	Test Case ID	Profile-1A

Test case description	Testing updating profile functionalities	Test Priority	Moderate
Pre-Requisites	User should have an account	Post-Requisites	None

Sl no	Action	Inputs	Expected output	Actual output	Test Browser	Test Result
1	Change first name	S	Error message: Your first name must be in between 2 and 25 characters	Error message: Your first name must be in between 2 and 25 characters	Chrome	Pass
2	Change last name	M	Error message: Your last name must be in between 2 and 25 characters	Error message: Your last name must be in between 2 and 25 characters	Chrome	Pass
3	Change email id	xy	Error message: Please include @ in the email id	Error message: Please include @ in the email id	Chrome	Pass
4	Change email id	xyz@example.com	Details saved successfully	Details saved successfully	Chrome	Pass
5	Change first name	Sr	Details saved successfully	Details saved successfully	Chrome	Pass
6	Change last name	Mm	Details saved successfully	Details saved successfully	Chrome	Pass
7	Update password: Enter old password, new password, confirm new password	Testing999*, Testing99*, Testing99*	Password updated successfully	Password updated successfully	Chrome	Pass
8	Update password: Enter old password, new password, confirm new password	Testing999*, Testing99*, Testing3399*	Passwords do not match, password not updated	Passwords do not match, password not updated	Chrome	Pass

5.3.3 Search and watch movie

Test Scenario ID	Search and watch movie	Test Case ID	Movie-1A
Test case description	Testing search and watch movie functionalities	Test Priority	High
Pre-Requisites	None	Post-Requisites	None

Sl no	Action	Inputs	Expected output	Actual output	Test Browser	Test Result
1	Search Movie	Enter name of movie in search bar	Filtered results are displayed	Filtered results are displayed	Chrome	Pass
2	Click on a movie to watch		Movie starts playing	Movie starts playing	Chrome	Pass

5.3.4 Subscription and purchase movie

Test Scenario ID	Subscription and purchase	Test Case ID	Transaction-1A
Test case description	Testing subscription and purchase functionalities	Test Priority	High
Pre-Requisites	None	Post-Requisites	None

Sl no	Action	Inputs	Expected output	Actual output	Test Browser	Test Result
1	Purchase subscription: Order id and amount to pay appear automatically, Click checkout		Redirected to third party app where transaction occurs, payment fails because the business account is not registered	Redirected to third party app where transaction occurs, payment fails because the business account is not registered	Chrome	Pass
2	Purchase movie: Order id and amount to pay appear automatically, select the movie to buy from the dropdown list, Click checkout	Lone star (1996) selected from dropdown	Redirected to third party app where transaction occurs, payment fails because the business account is not registered	Redirected to third party app where transaction occurs, payment fails because the business account is not registered	Chrome	Pass

5.3.5 Reviews and recommendations

Test Scenario ID	Reviews and recommendations	Test Case ID	Recommendation-1A

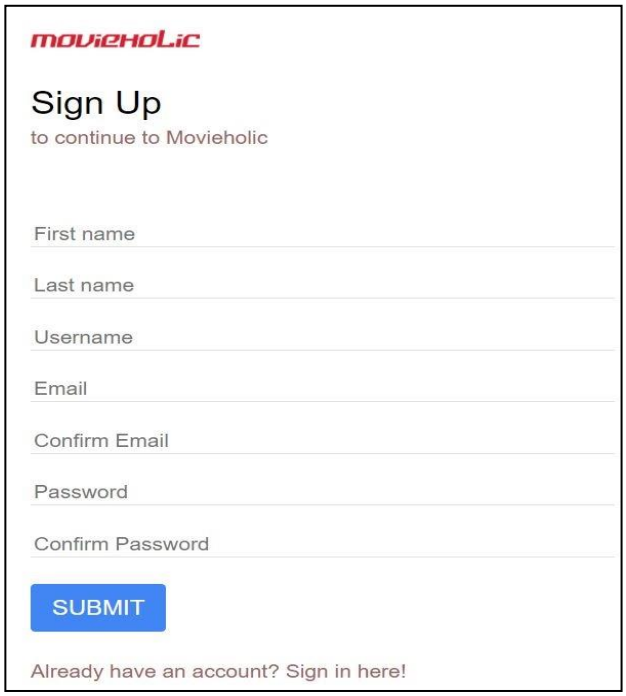
Test case description	Testing reviewing and movie recommendation functionalities	Test Priority	Moderate
Pre-Requisites	User should have written a positive review for a movie	Post-Requisites	None

Sl no	Action	Inputs	Expected output	Actual output	Test Browser	Test Result
1	Review page: Enter username, moviename, select star rating from the dropdown, enter review	sm2000, Lone Star (1996), 5, Good movie!	Review submitted succesfully	Review submitted succesfully	Chrome	Pass
2	Review page: Enter username, moviename, select star rating from the dropdown, enter review	sm2000, Lone S (1996), 5, Good movie!	Error message: Invalid movie	Review submitted successfully	Chrome	Fail
3	Recommendations page: Visiting the recommendations page after writing a positive review for a moviw		List of recommended movies is displayed	List of recommended movies is displayed	Chrome	Pass

References:

1. http://data.conferenceworld.in/IFUNA18DEC16/P513-528.pdf
2. https://www.journals.elsevier.com/telecommunications-policy/call-for-papers/special-issue-ott-and-live-streaming-services
3. Fundamentals_of_Database_Systems,_6th_Edition
4. NoSQL-Distilled_Book
5. https://www.w3schools.com
6. https://www.dreamcast.in/blog/what-are-the-features-of-good-video-platform/
7. https://www.tutorialspoint.com/php/index.htm
8. https://www.visual-paradigm.com/guide/data-flow-diagram/what-is-data-flow-diagram/#:~:text=Also%20known%20as%20DFD%2C%20Data,divided%20into%20logical%20and%20physical.
9. https://beginnersbook.com/2015/04/e-r-model-in-dbms/
10. https://byte-notes.com/relational-schema/#:~:text=Relational%20schema%20refers%20to%20the,structure%20organizes%20data%20into%20tables.
11. https://www.import.io/post/what-is-data-normalization-and-why-is-it-important/#:~:text=The%20Importance%20of%20Data%20Normalization&text=Data%20normalization%20gets%20rid%20of,information%2C%20or%20updating%20existing%20information.
12. https://stackoverflow.com/questions/39122710/php-and-mongodb

APPENDIX A: Snapshots of Output



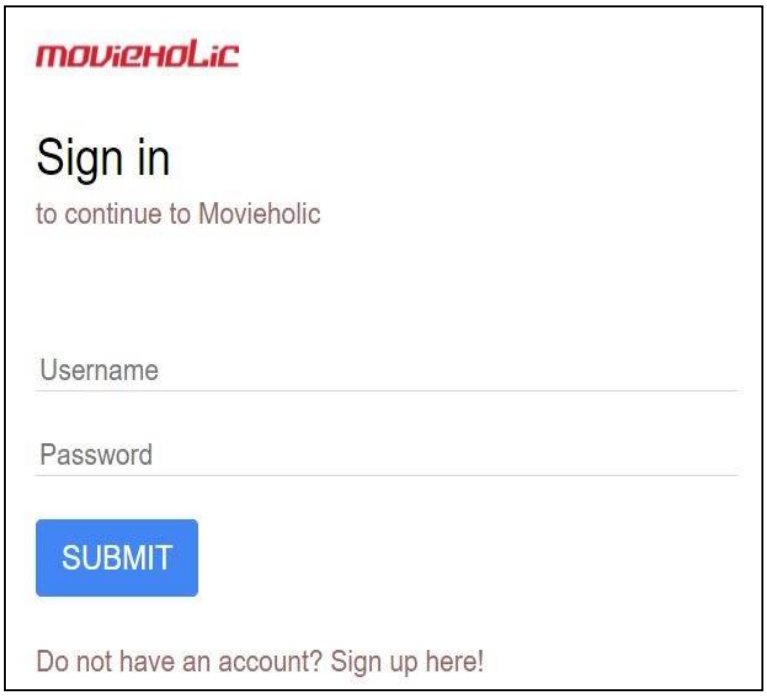


Figure B.1: Signup and login module

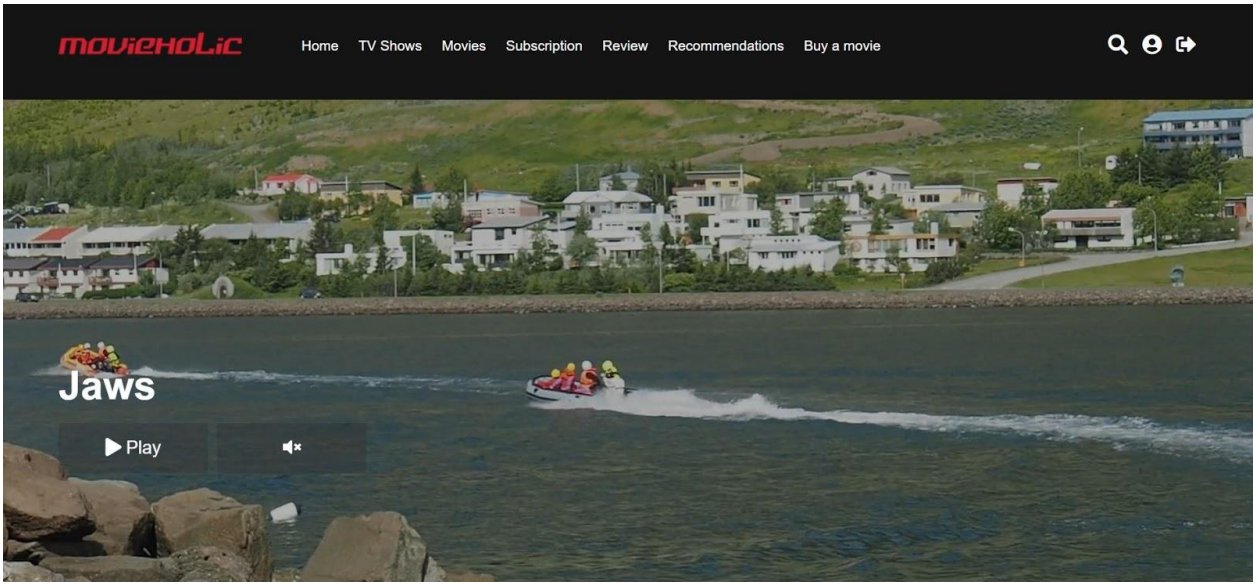


Figure B.2: Home page

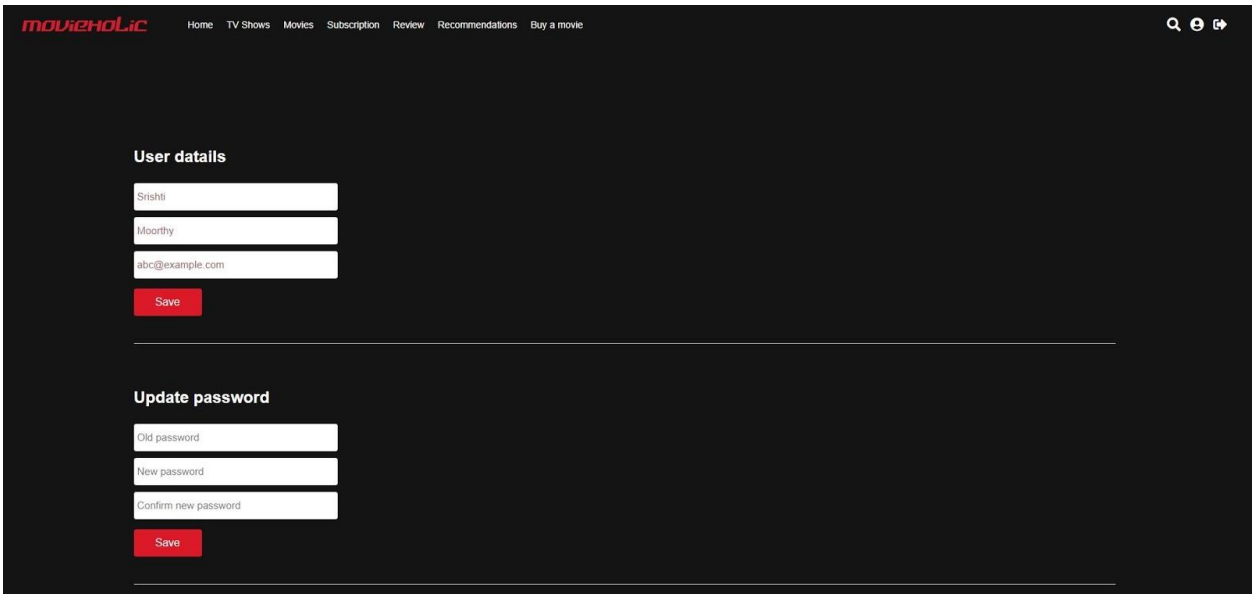


Figure B.3: User profile page

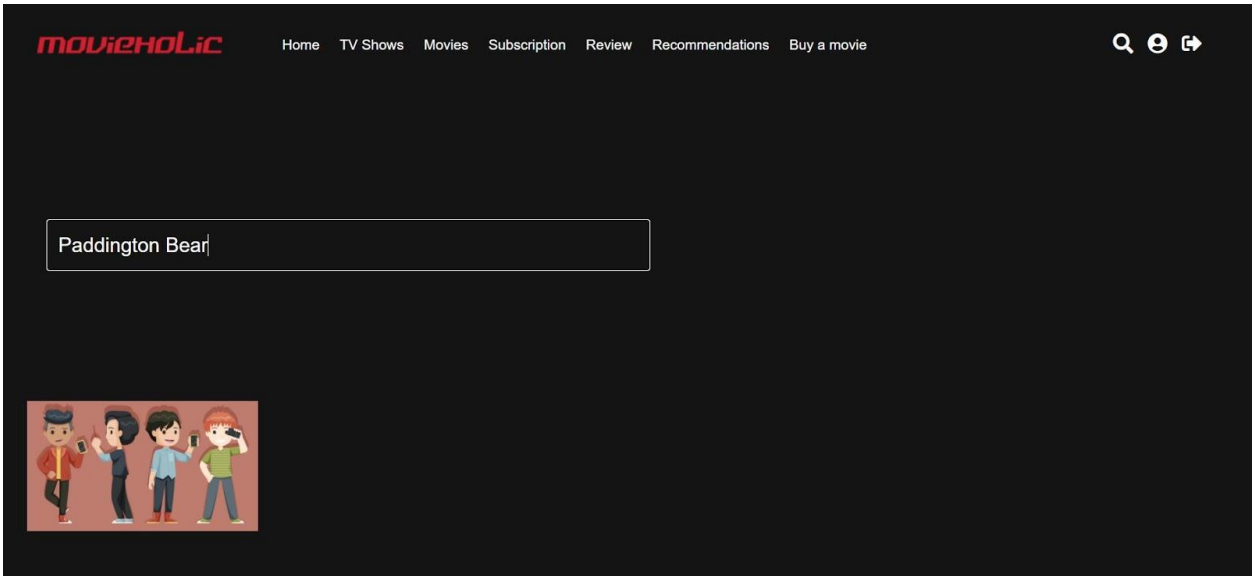


Figure B.4: Search page

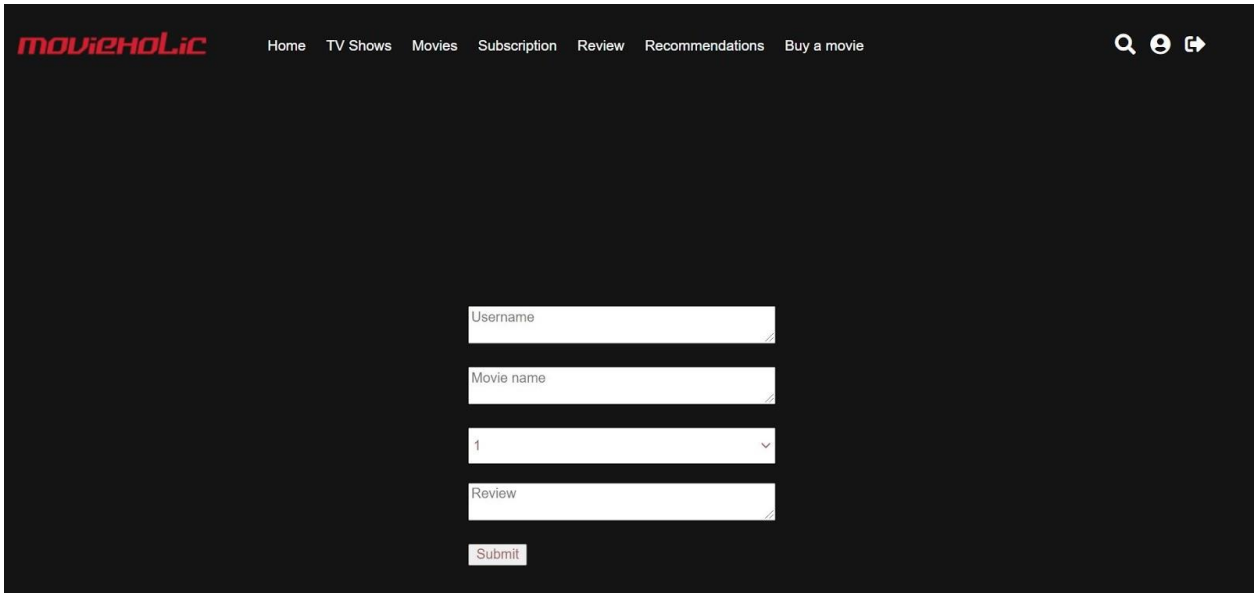


Figure B.5: Review page

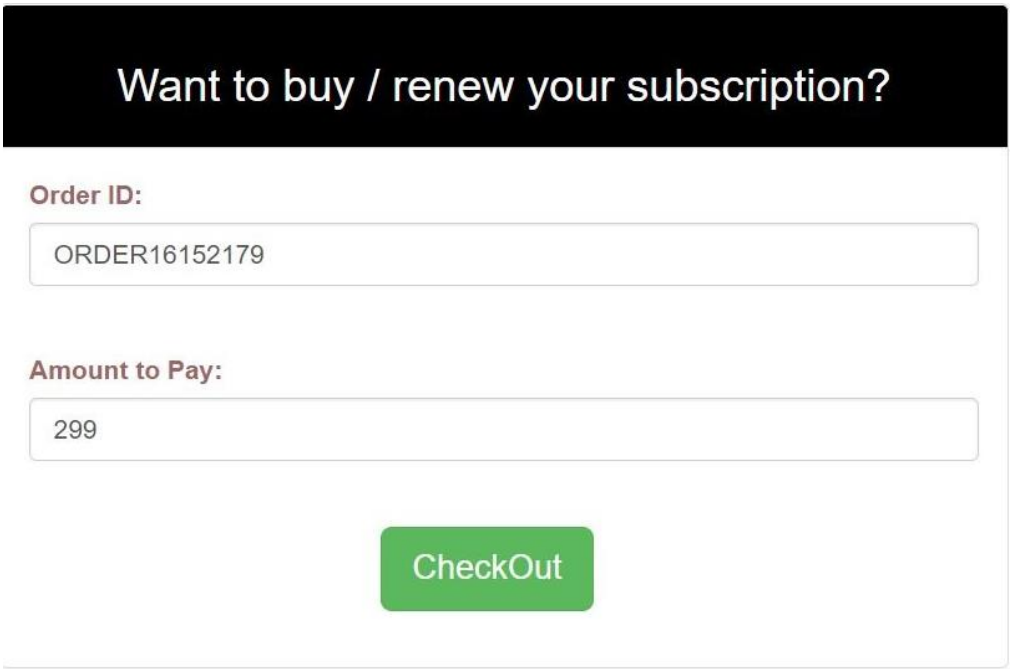


Figure B.6: Subscription module

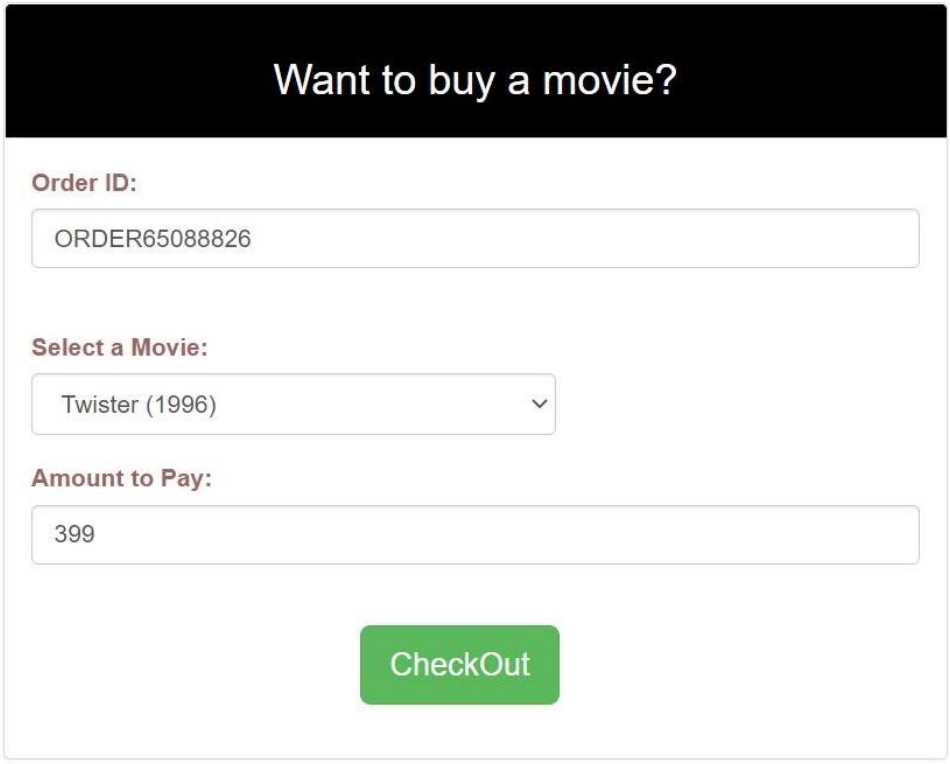


Figure B.7: Purchase movie module

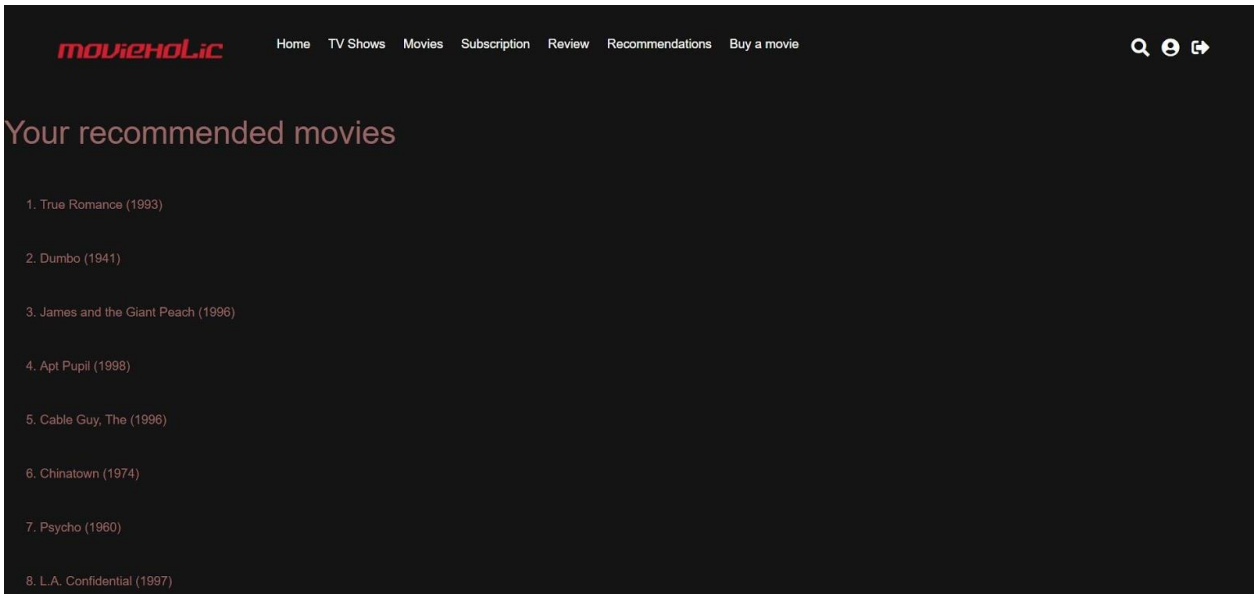


Figure B.8: Recommendations page