**Online Movie Browser Application**

*Synopsis*

*Submitted by*

**Kota Divya**

**(SAP ID: 51902687)**

***In partial fulfillment for the successful completion of Mode-1 training***

**HCL Technologies**

**Table of contents:**

1. **Introduction**

1.1 Purpose

1.2 Objective

1.3 Technologies Used

**2. Preliminary Design**

2.1 Use Case – Online Movie Browser

2.2 Specification of actors

2.3 Specification of use cases

2.4 E. R Diagram -Online Movie Browser

2.5 Flow chart for online Movie Browser

2.6 DataBase table design

**3. Appendices**

1. **Introduction**

**1.1** **Purpose:**

Online Movie Browser is a website where we can browse movies based on our wish and also view the details of the selected movie and admin can do modifications to database.

**1.2 Objective:**

The basic objective of developing this application is:

* Provides a home page where unauthorized/authorized users can search new movies based on wish.
* Provides a detail of the movie, if the user clicks on the movie picture.
* Provides a view of all movies present in the application to the authorized/unauthorized users.
* Provides a login page where admin can login in order to do operations like adding a new movie, delete a movie detail, update/edit a movie detail into database.
* Provides a registration page also if user wants to do modifications to database.
* Provides a logout facility where admin can be logged out.

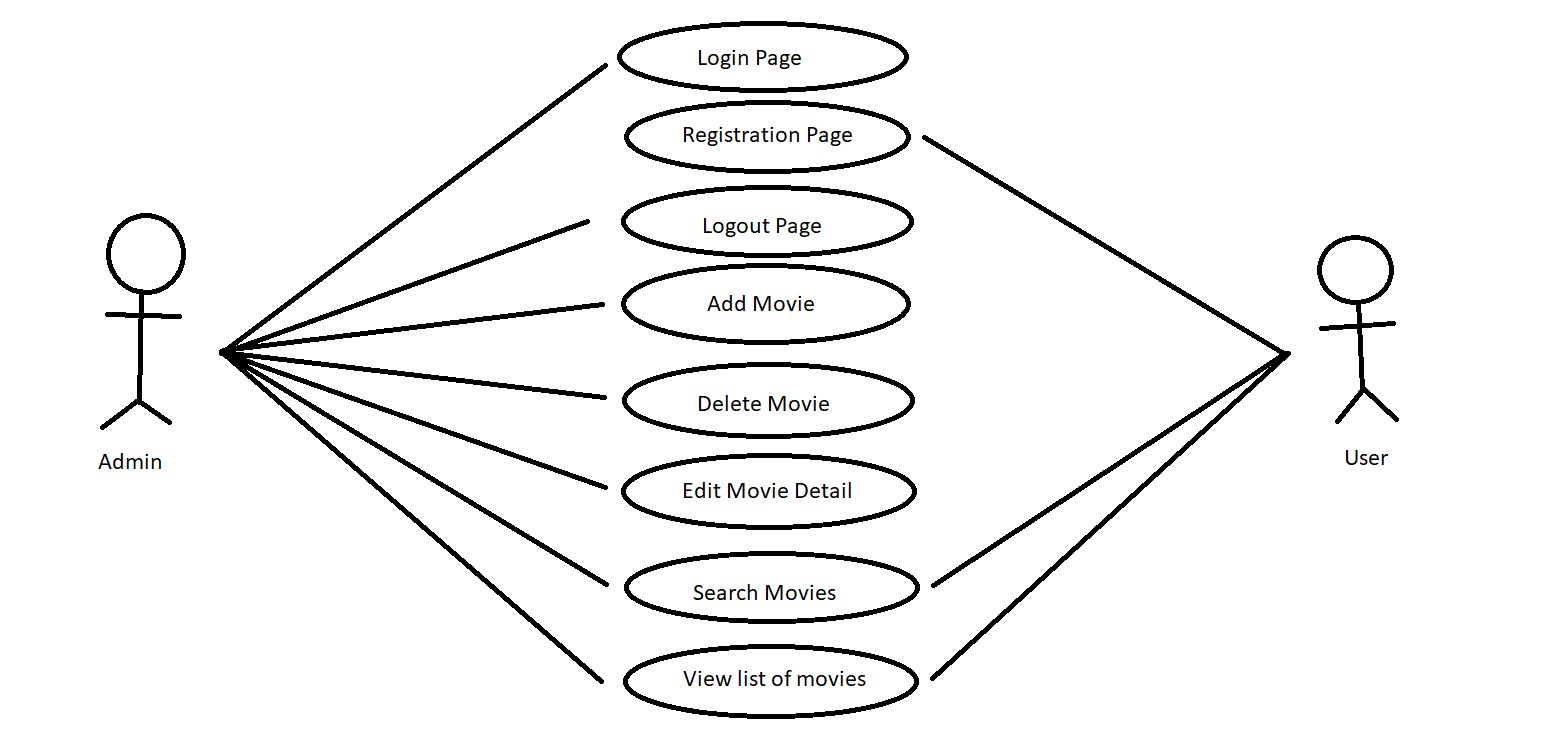
**1.3 Technologies Used:**

* The online movie browser application shall operate on a server running the latest version of IIS (Internet Information Server).
* Software Requirements: Visual Studio Community, SQL server
* Modules used: ASP .NET, Web API, C#, M.V.C

**2. Preliminary Design**

**2.1 Use Case Diagram - Online Movie Browser:**

* The use case diagram is usually referred to as behavior diagram used to describe the actions of all user in a system.
* All user describe in use case are actors and the functionality as action of system.

**Use Case diagram for Online Movie Browser:**

**2.2 Specification of actors:**

**Admin**

|  |  |
| --- | --- |
| **Admin** | |
| **Element** | **Details** |
| Description | An admin is a user of the online movie browser application. |
| Examples | An admin who has account in online movie browser application can modify the database. |

**User**

|  |  |
| --- | --- |
| **User** | |
| **Element** | **Details** |
| **Description** | A user normally wants to search movies in online movie browser application. |
| **Examples** | A user who does not have an account in online movie browser application in order to modify the database. |

**2.3 Specification of use cases:**

**Use case Login page**

|  |  |
| --- | --- |
| **Login page** | |
| **Element** | **Details** |
| Actor | Admin |
| Trigger | Admin has to be login before going to done modifications to database. |
| Pre-Conditions | Admin credentials already present in the database. |
| Post-Conditions | Admin will get access to do modifications in database after successful login. |
| Normal Course | 1. Admin clicks the add/delete/edit movie button he will be redirects to login page. 2. An empty login form will be displayed. 3. Admin will fill the username and password fields in order to get access to modify the database. 4. Validations of all the fields are to be done for both client and server side. 5. Admin credentials has to be submitted to the server. 6. Messages will be displayed after successful login of admin |
|  |  |

**Use case Registration page**

|  |  |
| --- | --- |
| **Registration page** | |
| **Element** | **Details** |
| Actor | User |
| Trigger | User has to be register before going to login page in order to do modifications to database. |
| Pre-Conditions | New user credentials were not present in the database. |
| Post-Conditions | New user will get access to do modifications in database after successful registration. |
| Normal Course | 1. New user clicks the add/delete/edit movie button he will be redirects to login page. 2. An empty login form will be displayed for login. 3. If new user will try to login without completing registration respective messages will be displayed to do registration. 4. New user clicks the registration button 5. An empty registration form will be displayed. 6. New user will fill all the details of the registration form. 7. Validations of all the fields are to be done for both client and server side. 8. New user credentials have to be submitted to the server. 9. Messages will be displayed after successful registration of new user. |
|  |  |

**Use case Logout page**

|  |  |
| --- | --- |
| **Logout page** | |
| **Element** | **Details** |
| Actor | Admin |
| Trigger | Admin can logout after doing modifications to database. |
| Pre-Conditions | Admin credentials were present in the database to login and can also logout. |
| Post-Conditions | After logout of admin, Messages will be displayed for successful logout of admin. |
| Normal Course | 1. Admin clicks the logout button after login. 2. Messages will be displayed after successful logout of admin. 3. After logout of admin, he will be redirect to login page for further modifications to database. 4. If he does not want to do any further modifications to database, he can click on back button which redirects to home page. |
|  |  |

**Use case Add Movie**

|  |  |
| --- | --- |
| **Add Movie Details** | |
| **Element** | **Details** |
| Actor | Admin |
| Trigger | New movie details need to be added |
| Pre-Conditions | Movie details was not present in database |
| Post-Conditions | New movie will get added to the database. |
| Normal Course | 1. Admin click the Add Movie button after login 2. Empty Add New Movie form will be displayed 3. Admin will fill the details of the new movie 4. Validation of all the fields is to be done 5. Details has to be submitted to the server 6. Messages will be displayed after successful creation of new movie |
|  |  |

**Use case Delete Movie**

|  |  |
| --- | --- |
| **Delete Movie Details** | |
| **Element** | **Details** |
| Actor | Admin |
| Trigger | Admin wants to delete movie details |
| Pre-Conditions | Movie details already existing in database |
| Post-Conditions | Existing movie details will get deleted in the database |
| Normal Course | 1. Admin click the Delete movie button after login 2. Details of the movie will be deleted in the database 3. Messages will be displayed after successful deletion of movie |
|  |  |

**Use case Edit Movie**

|  |  |
| --- | --- |
| **Edit Movie Details** | |
| **Element** | **Details** |
| Actor | Admin |
| Trigger | Admin wants to modify the movie details |
| Pre-Conditions | Movie details already present in database |
| Post-Conditions | Existing movie details will get updated in the database |
| Normal Course | 1. Admin click the Edit movie button after login 2. Edit form will be displayed to update movie details 3. Admin will modify the details of the movie 4. Validation of all the fields is to be done 5. Updated movie details have to be submitted to the database 6. Messages will be displayed after successful updating of movie details |
|  |  |

**Use case Search Movie**

|  |  |
| --- | --- |
| **Search Movie Details** | |
| **Element** | **Details** |
| Actor | Admin/User |
| Trigger | Admin/user wants to search movie based on their wish |
| Pre-Conditions | Movie details was already present in the database |
| Post-Conditions | Movie details will be displayed to user based on searching movie |
| Normal Course | 1. User/Admin can search the movie based on their wish 2. List of movies will be displayed based on searching 3. If user/admin wants to view a particular movie detail, then can click on the specific movie picture 4. Then, the particular movie detail has to be displayed on the browser to the user/admin |
|  |  |

**Use case List of all movies**

|  |  |
| --- | --- |
| **List of all movie** | |
| **Element** | **Details** |
| Actor | Admin/User |
| Trigger | Admin/user wants to view list of all movies present in the database |
| Pre-Conditions | Movie details was present in database |
| Post-Conditions | New movie will get added to the database |
| Normal Course | 1. Admin/User clicks the List of movies button without login 2. Admin/User can view the list of movies 3. If user/admin wants to view a particular movie detail, then can click on the specific movie picture 4. Then, the particular movie detail has to be displayed on the browser to the user/admin |
|  |  |

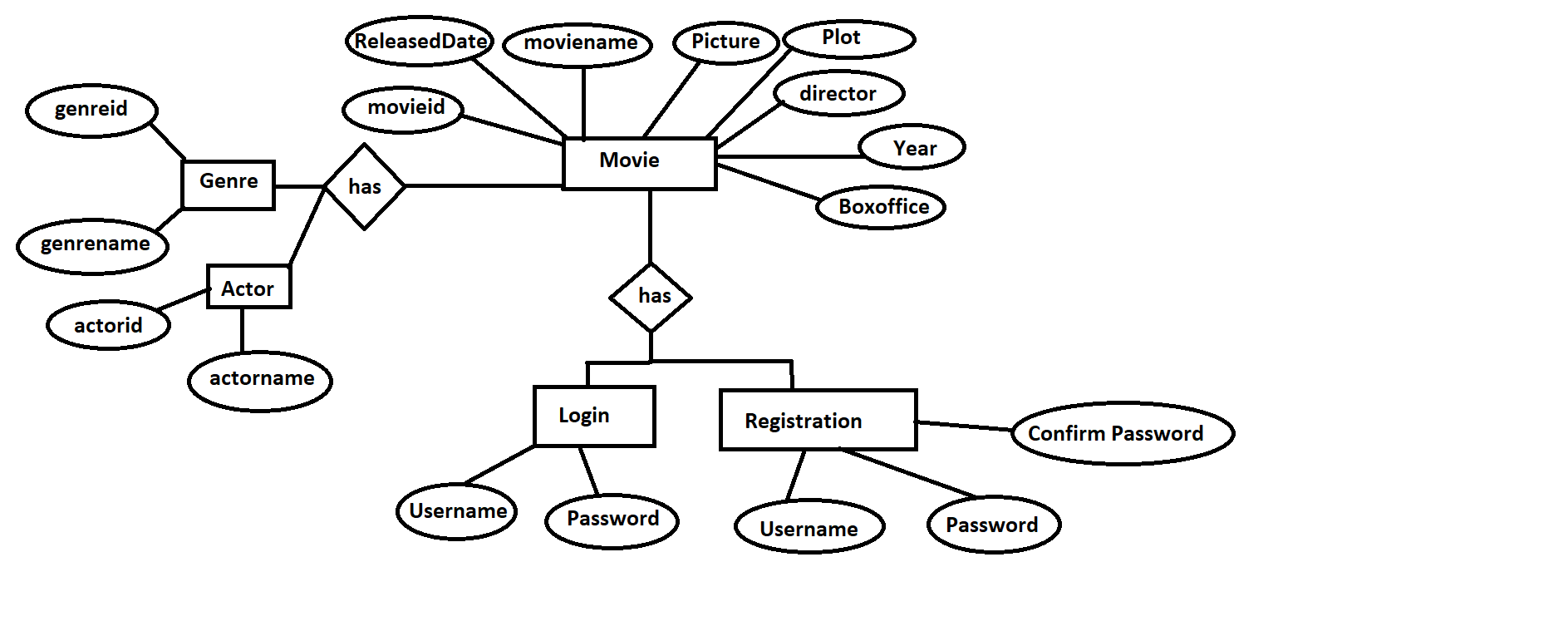
**2.4 E. R Diagram - Online Movie Browser**

This E.R (Entity Relationship) diagram represents the model of online movie browser application entity and shows all the visual instrument of database tables. It used structured data and to define relationships between structured data groups of online movie browser application functionalities. The main entities of the online movie browser application are movie, genre, actor, registration, login.

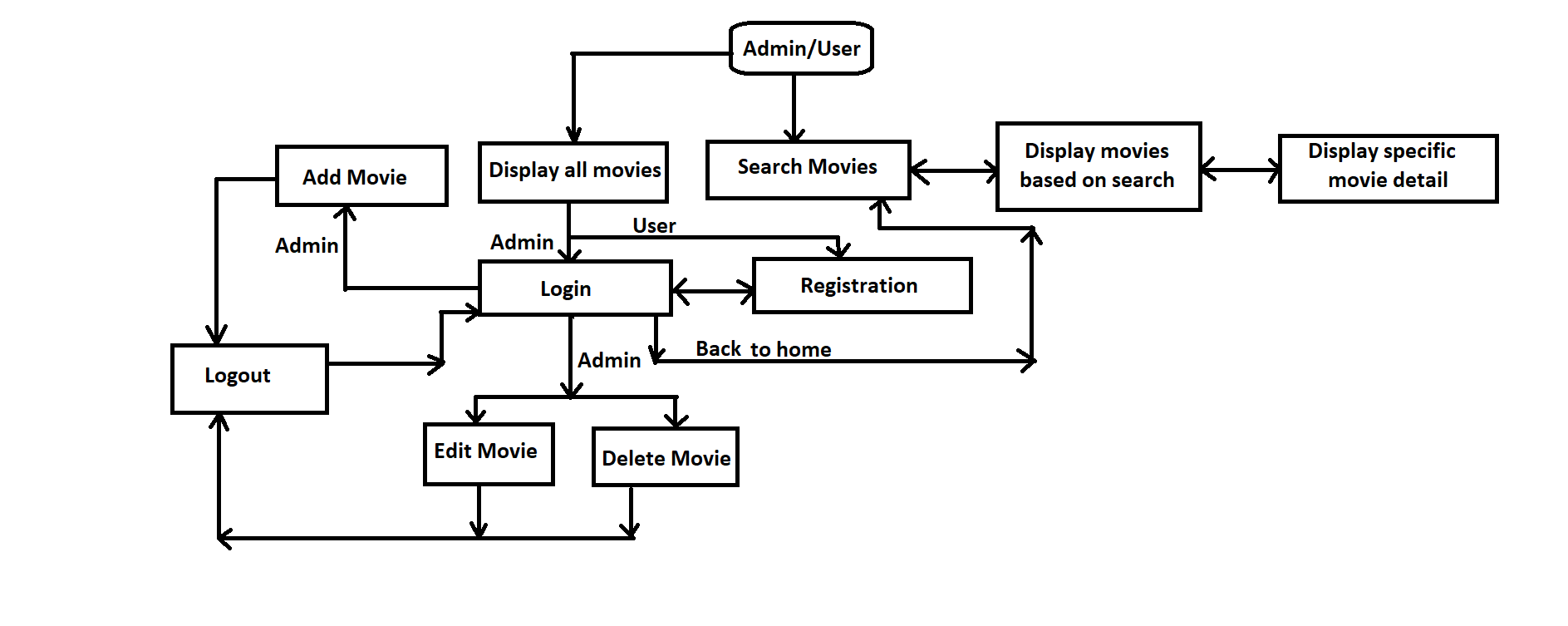
**online movie browser application entities and their attributes:**

* **Movie Entity:** Attributes of movie are MovieId, MovieName, Picture, Director, Plot, Year, Genre, Actor, BoxOffice, ReleasedDate
* **Genre Entity:** Attributes of genre are GenreId and GenreName.
* **Actor Entity:** Attributes of actor are ActorId and ActorName.**Description of Online Movie Browser DataBase**:
* The details of the movie are store into movie table.
* In the movie table, the movie\_id is the primary key.
* The details of the genre are store into genre table.
* In the genre table, the genre\_id is the primary key.
* The details of the actor are store into actor table.
* In the actor table, the actor\_id is the primary key.
* In the movie\_genre table, both the movie\_id and genre\_id are primary keys
* In the movie\_actor table, both the movie\_id and actor\_id are primary keys
* Here, there is a many-to-many relationship between table movie and table genre and table actor.

**ER diagram for online movie browser:**



**2.5 Flow Chart for Online Movie Browser:**

**2.6 DataBase table Design:**

In the online movie browser application, the database has tables like registration, userdetail, movie, actor, genre, moviegenre, movieactor.

**Userdetails (Owner of the online movie browser application):**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Type** | **Note** | **Remark** |
| userid | int | User Id | Primary key |
| username | varchar (100) | User Name |  |
| picture | varchar (100) | User Picture |  |
| city | varchar (100) | User City |  |
| state | varchar (100) | User Country |  |

**Registration:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Type** | **Note** | **Remark** |
| name | Varchar (50) | Admin name | Primary key |
| password | Varchar (30) | Admin password |  |
| Confirm password | Varchar (30) | Admin confirm password |  |

**Actor:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Type** | **Note** | **Remark** |
| actorid | int | Actor Id | Primary key, self-generated |
| actorname | Varchar (100) | Actor Name |  |

**Genre:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Type** | **Note** | **Remark** |
| genreid | int | Genre Id | Primary key, self-generated |
| genrename | Varchar (100) | genre Name |  |

**Movie:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Type** | **Note** | **Remark** |
| movieid | int | Movie Id | Primary key |
| moviename | Varchar (100) | Movie Name |  |
| picture | Varchar (100) | Movie Image |  |
| director | Varchar (100) | Movie Director |  |
| plot | Varchar (500) | Movie Plot |  |
| year | int | Movie Released Year |  |
| boxoffice | Varchar (100) | Box-Office collections |  |
| released | date | Released Date of movie |  |

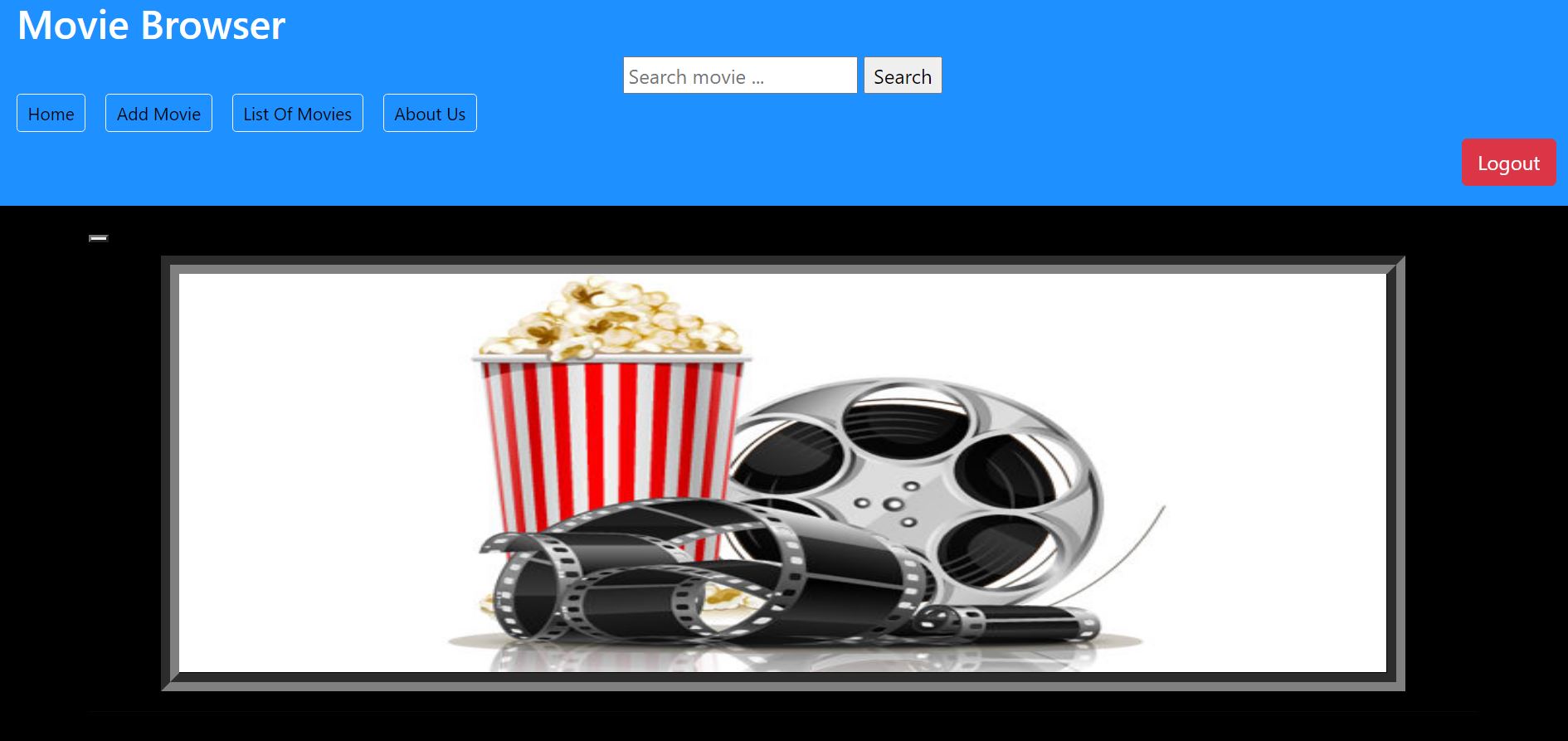
**moviegenre: (The whole table is primary key)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Type** | **Note** | **Remark** |
| movieid | int | Movie Id |  |
| genreid | int | Genre Id |  |

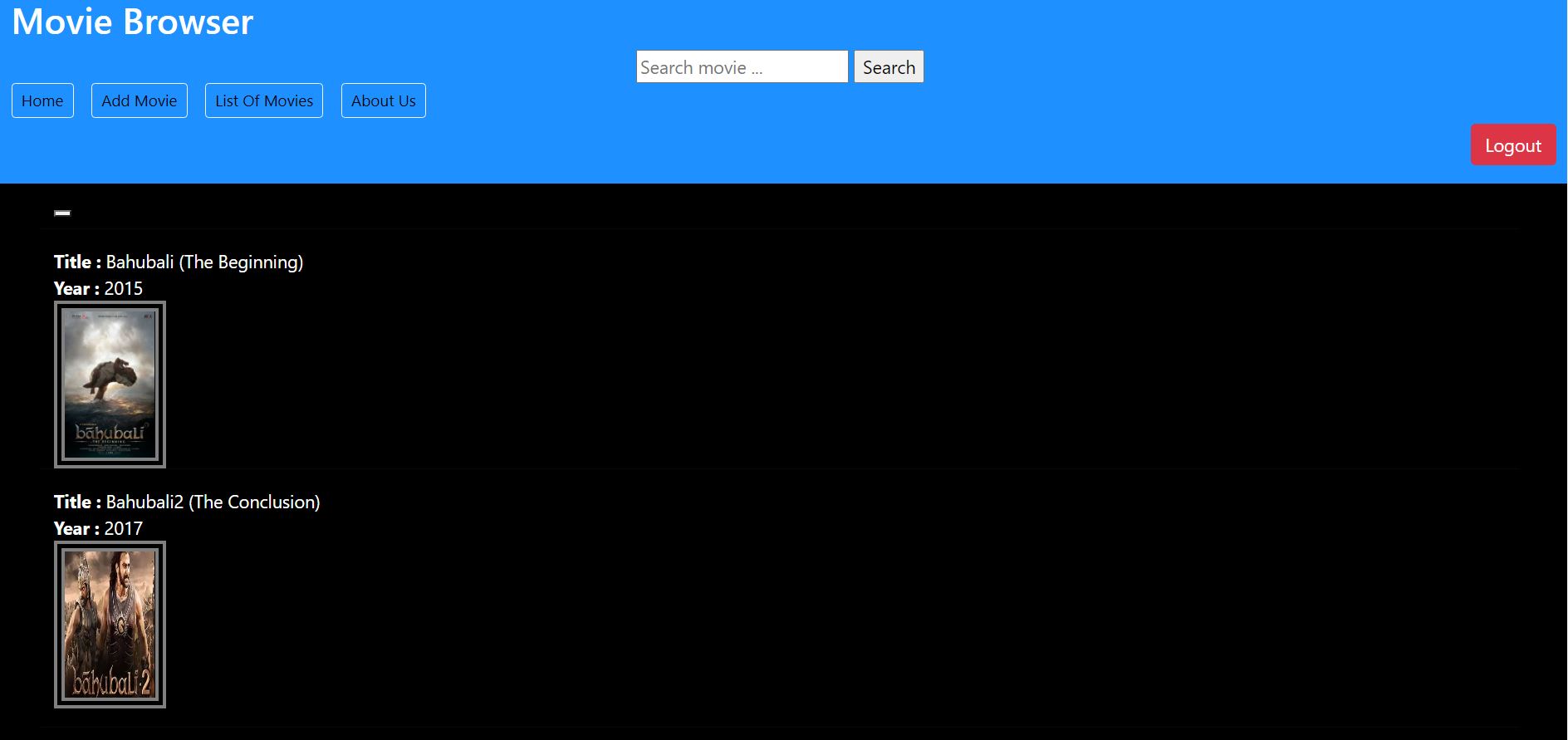
**movieactor: (The whole table is primary key)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field Name** | **Type** | **Note** | **Remark** |
| movieid | int | Movie Id |  |
| actorid | int | Actor Id |  |

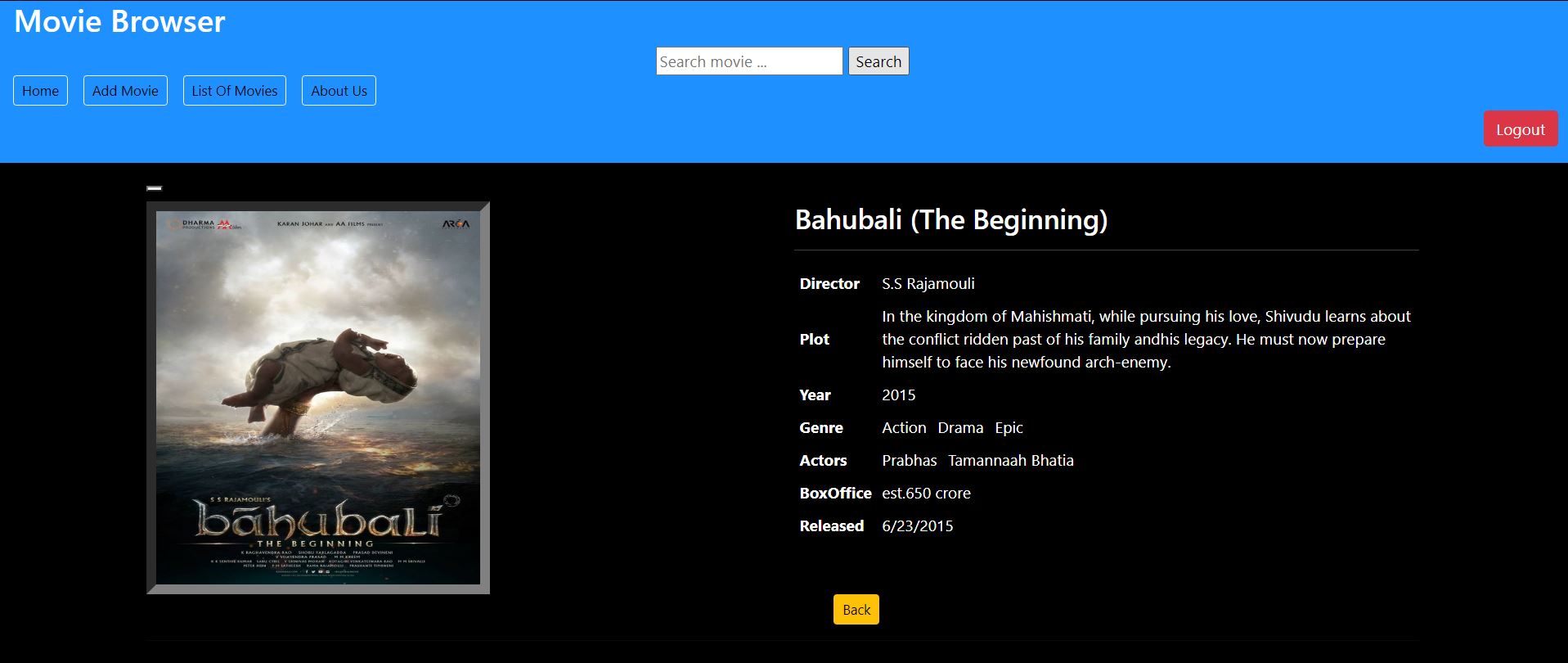
**Appendices:**

**i) Home Page:** 

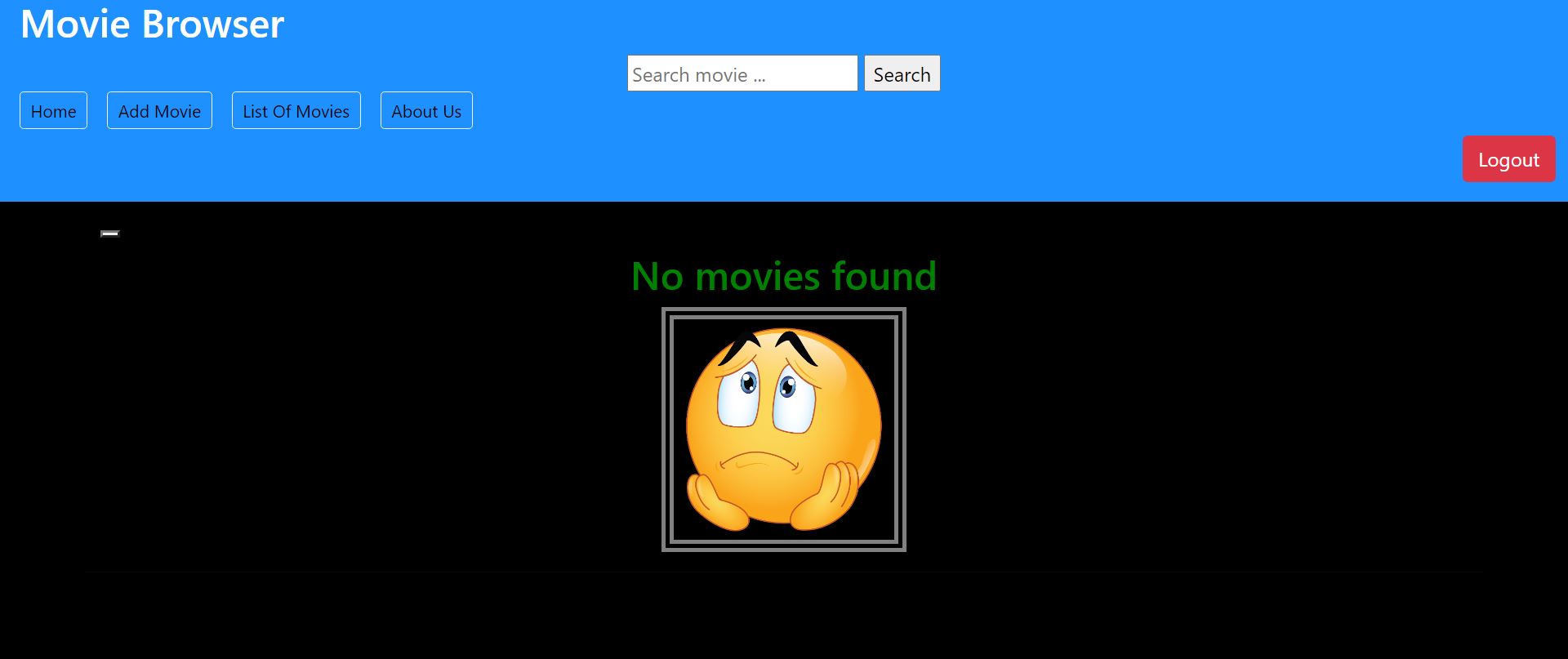
**ii) After Searching for movies:**



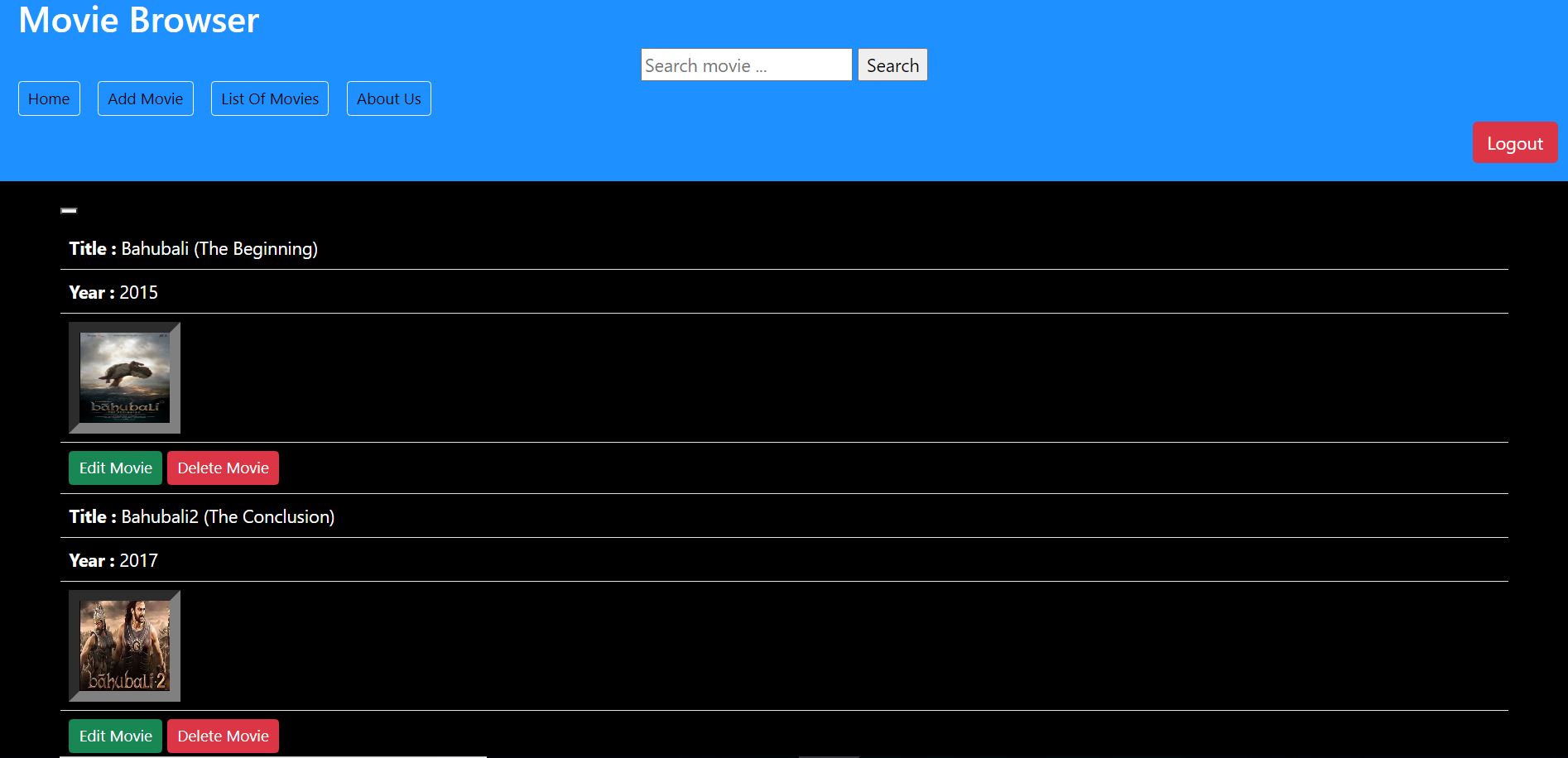
**iii) Displaying specific movie detail:**



**iv) No searching movies found page:**



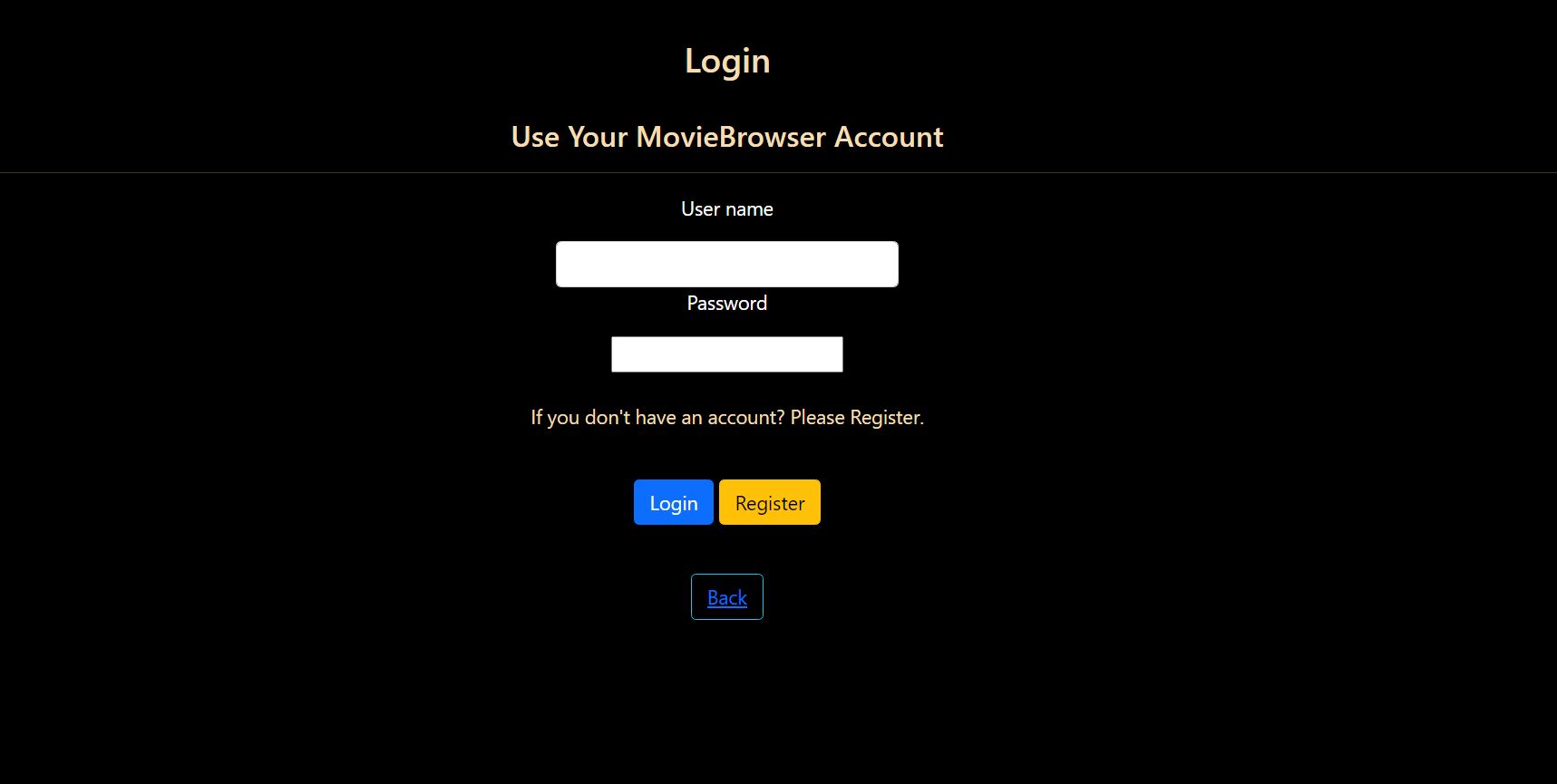
**v) Display all movies page:**



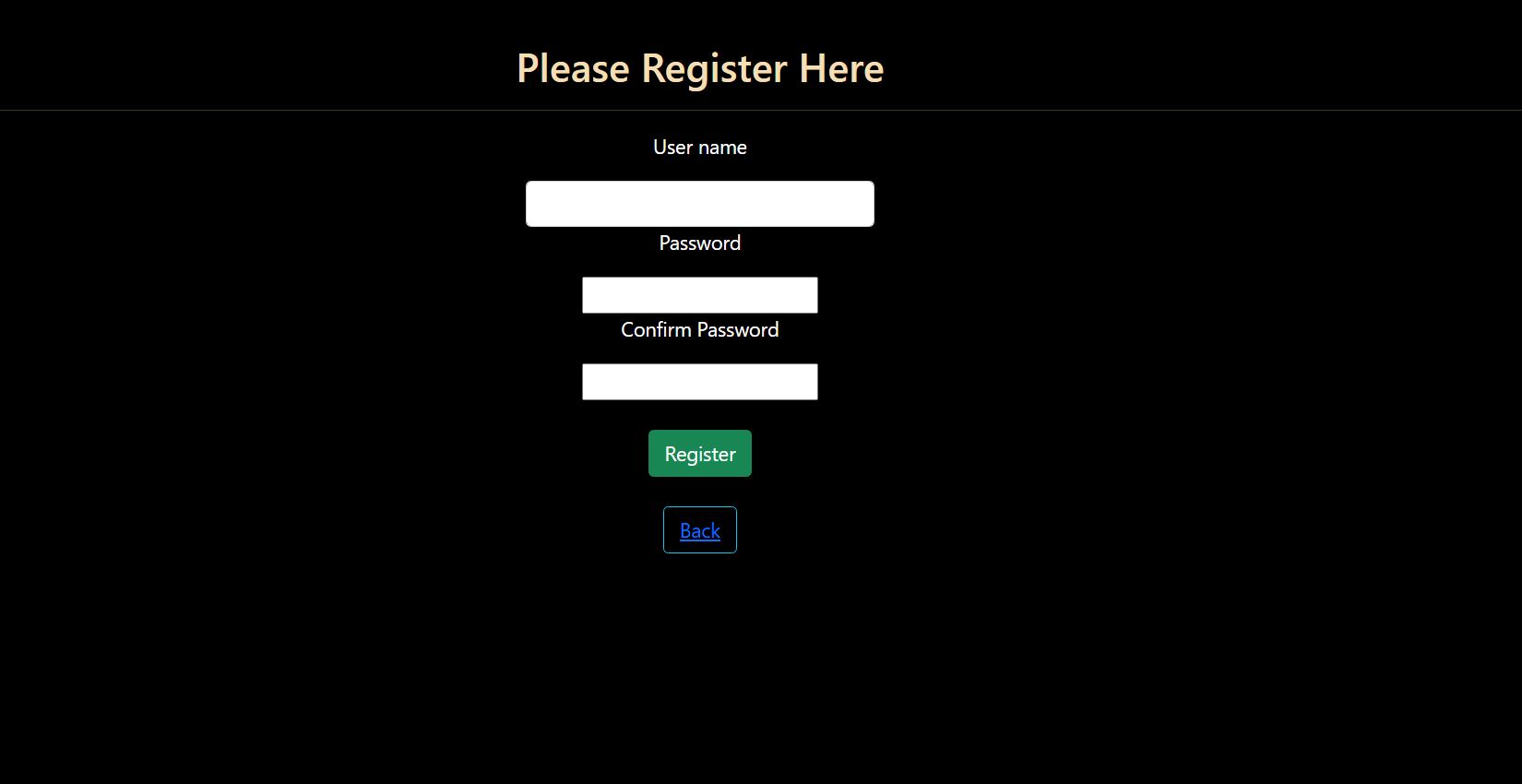




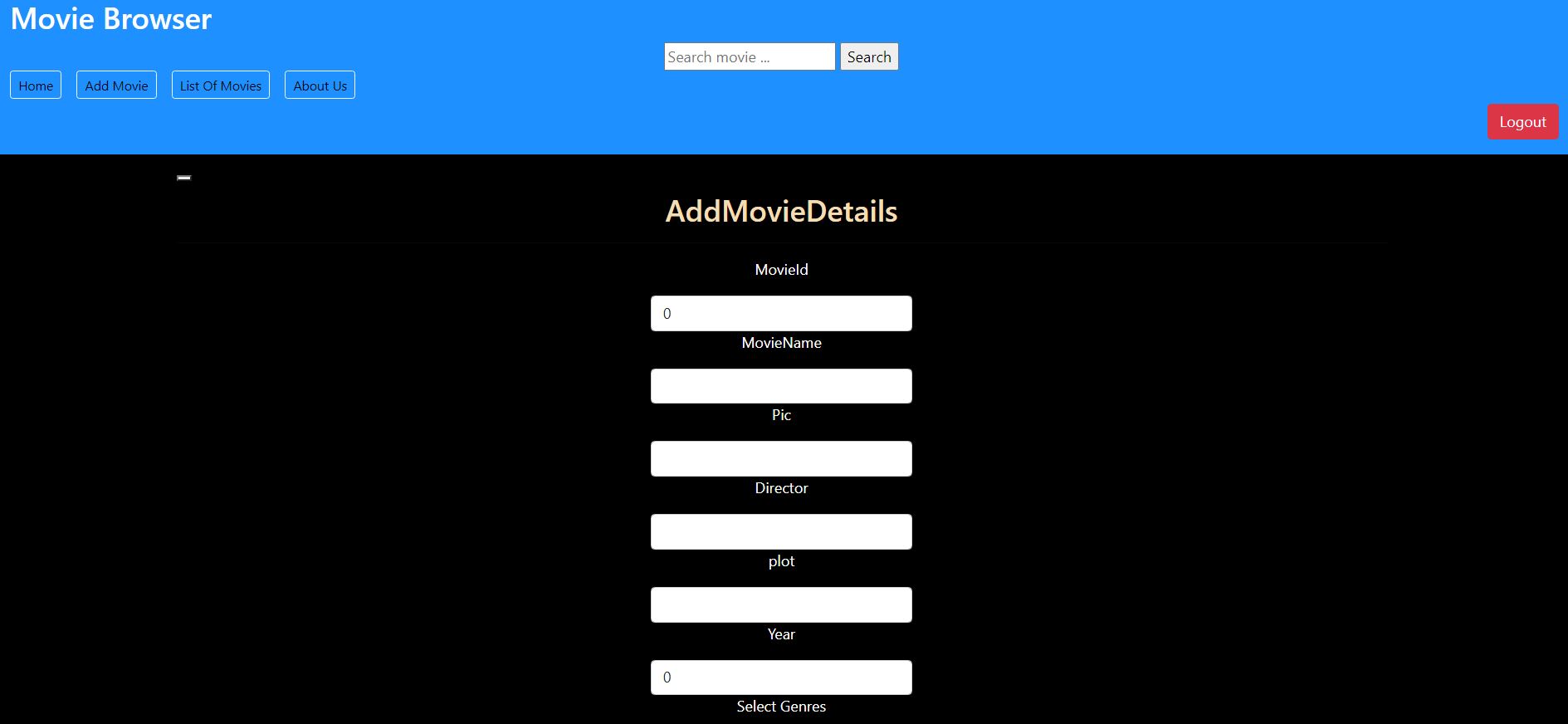
**vi) Login Page:**

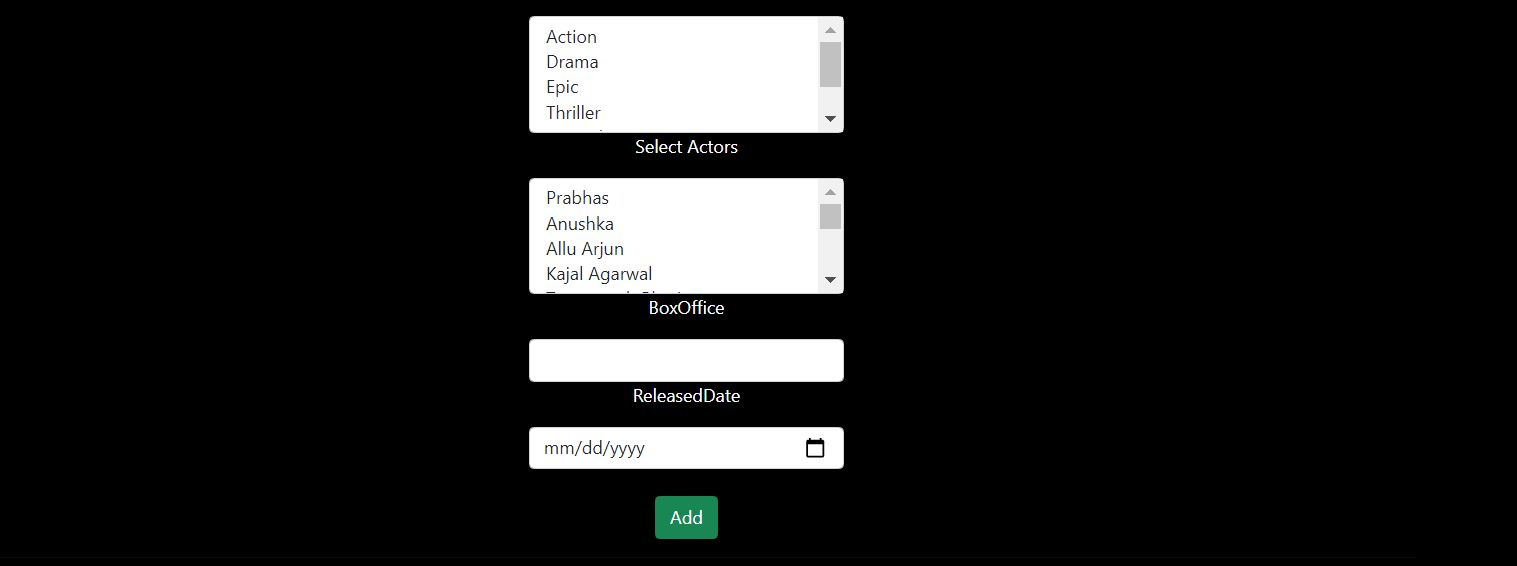


**vii) Registration Page:**

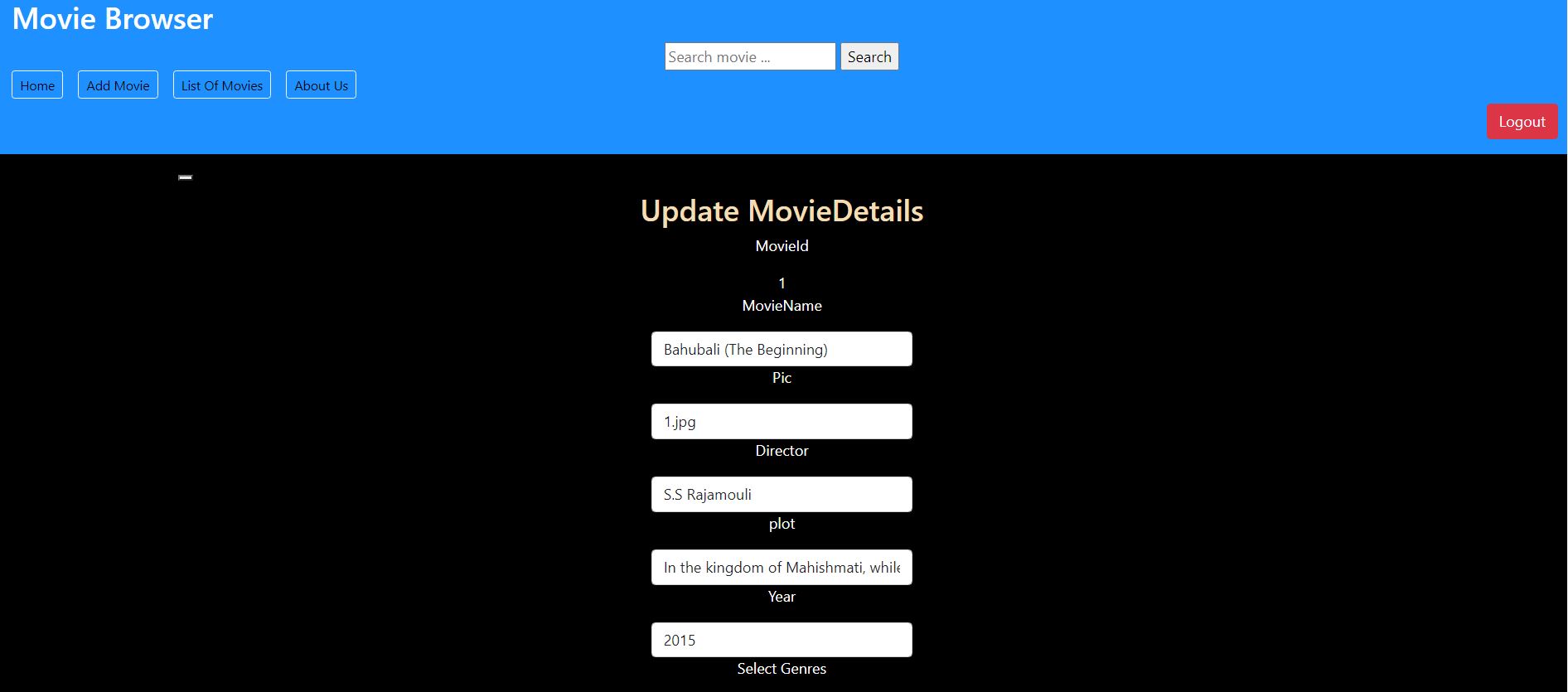


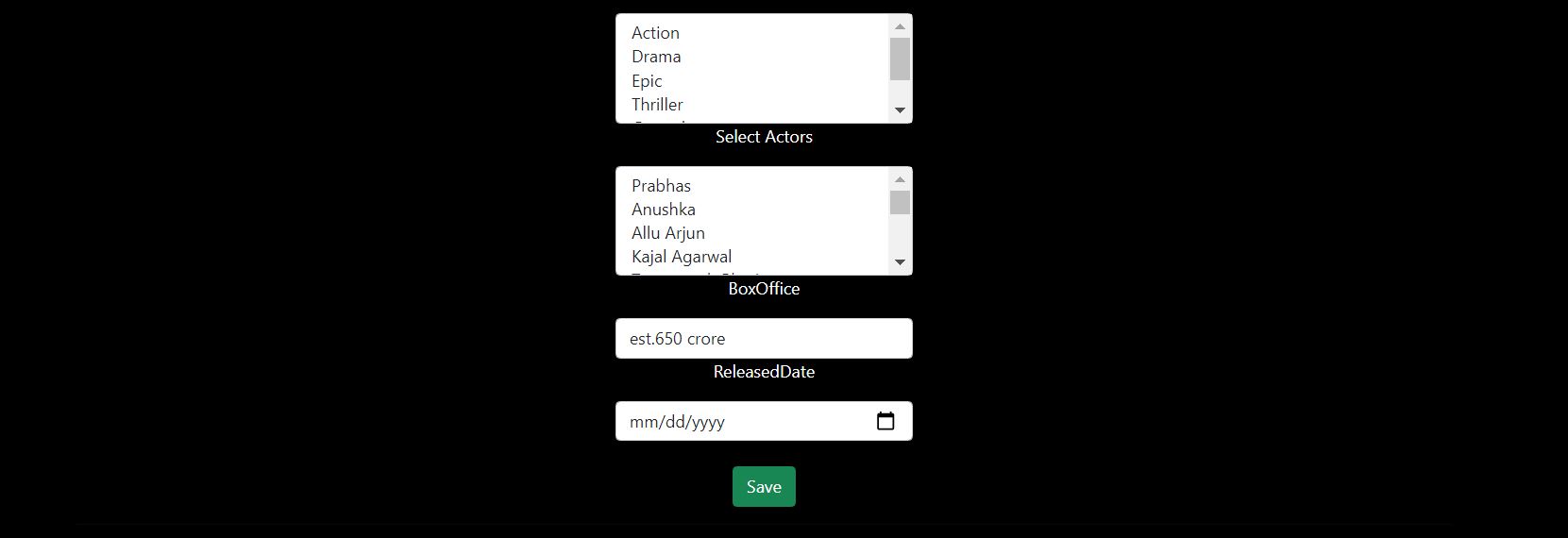
**viii) Add movie page:**



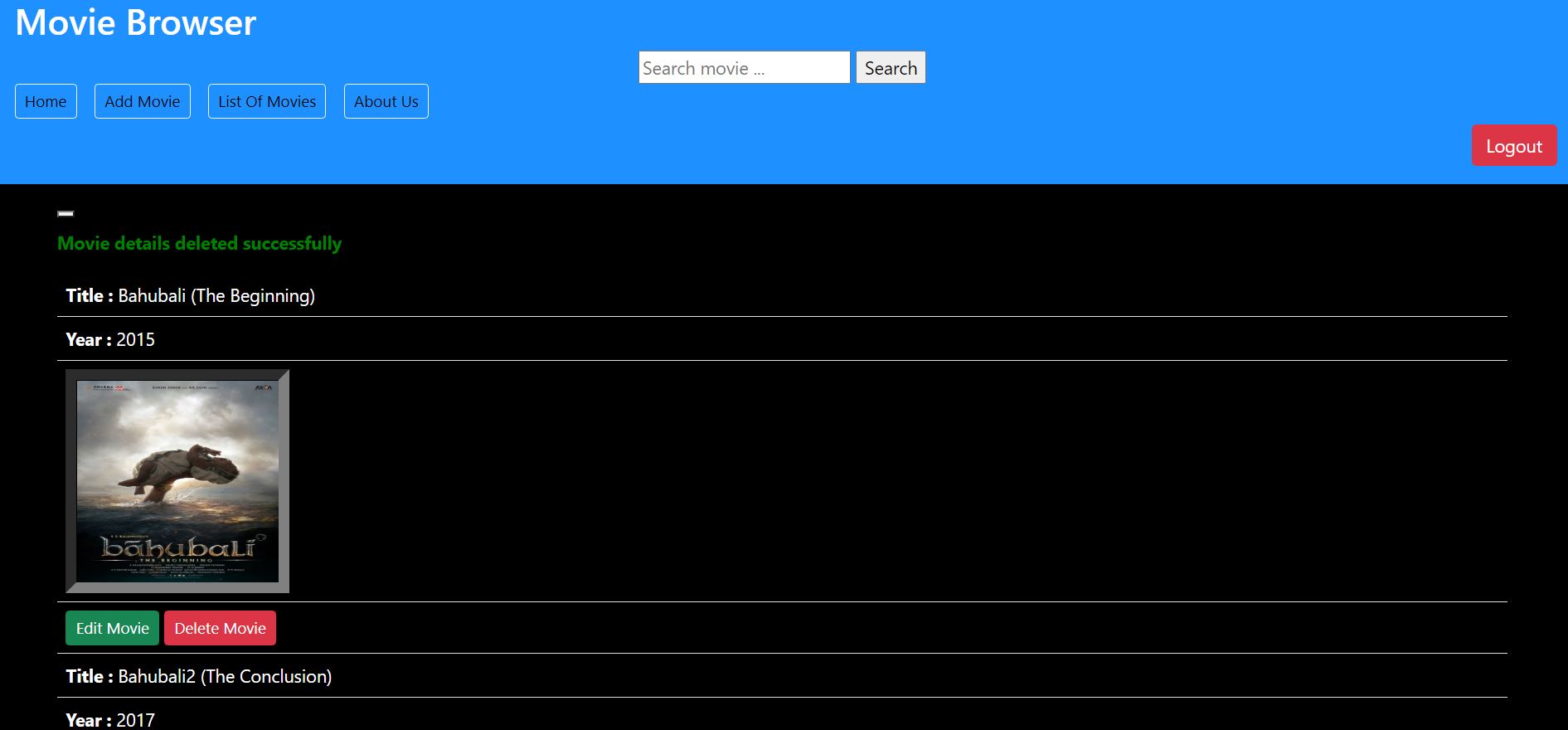


**iX) Edit movie page:**





**X) After deleted a movie:**



**Xi) About Us page (Owner of the application):**

