Lê Võ Thanh Duy ITITIU15030

Lưu Minh Quân ITITIU15033

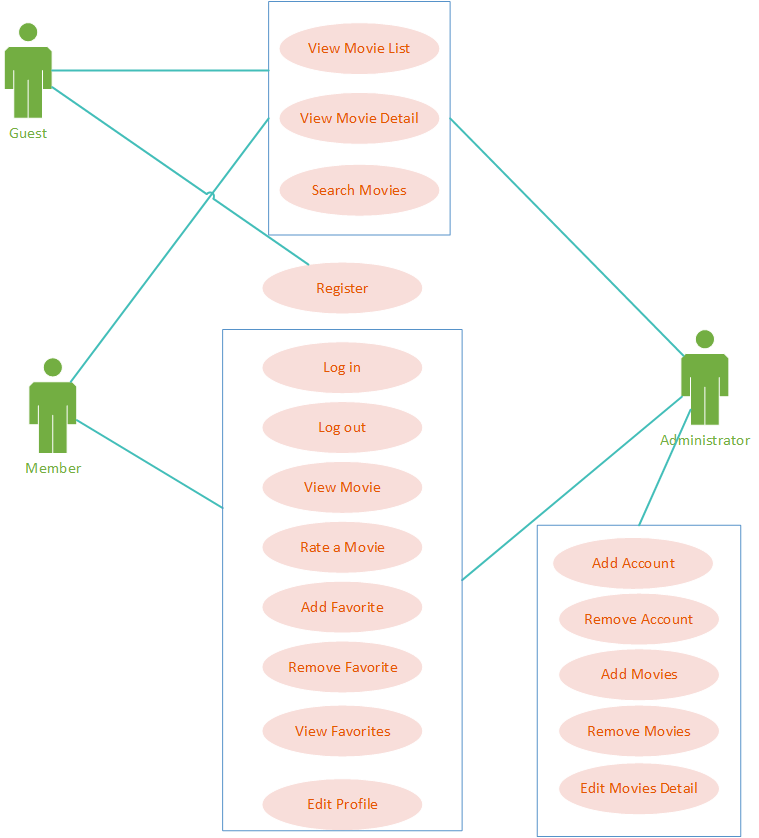
Đặng Quốc Cường ITITIU15005

**Web Application Development Document 2**

1. **Introduction**

Nowadays, movie and TV series are essential and irreplaceable parts of the people’s everyday entertainment. With the rapid development of the entertainment industry, there are various ways of watching movie and TV series. One of the fastest developing methods is video streaming with plenty of services such as Netflix, Amazon Prime, phimmoi and so on. But the problem is that some charge a very high subscription fee, some don’t but are embedded with annoying ads. With this in mind, we have decided to develop our own movie streaming service which is free, fast and convenient for users.

1. **Use case diagram and its description**
2. Use Case Diagram

****

1. **Use Case Description**

**Name: Search Movies**

Identifier: UC1

Inputs:

* Movie name

Outputs:

* A page with the movies with name most similar to the searches
* A page that shows fail search (no result)

Precondition:

* None

Basic course:

|  |  |
| --- | --- |
| Actors: Registered user, admin, guest | System |
| 1. Enter name and chooses category |  |
| 1. Submit | * 1. Check database for user’s searches   2. If found, return all results for the result page   3. If not, return a page with fail result |

Post condition:

* None

**Name: Register**

Identifier: UC2

Inputs:

* Username
* Password
* Email
* Date of birth

Outputs:

* A page with that shows successful user registration
* The register page again to ask for correct input if fail

Precondition:

* None

Basic course

|  |  |
| --- | --- |
| Actors: guest | System |
| 1. Request register page | 1.1 Show page |
| 1. Fill information |  |
| 1. Submit | * 1. Check if the information is suitable   2. If success return success page   3. If fail return register page again with added information |

Post condition:

* Database has been uploaded.

**Name: Log in**

Identifier: UC3

Inputs:

* User Name
* Password

Outputs:

* The home page with user’s authentication [if success]
* The login page [if fail]

Basic Course

|  |  |
| --- | --- |
| Actor: User(User/ Administrator) | System |
| 1. Open the login page | * 1. Display the login page |
| 1. Enter user name and password |  |
| 1. Submit | * 1. Check user’s info.   2. If success, return the home page   3. If fail, the login page returns |

Precondition

* User has an account which was registered before

Post condition: None

**Name: Log out**

Identifier: UC4

Inputs:

* The home page with authentication

Outputs:

* The home page without authentication

Basic Course

|  |  |
| --- | --- |
| Actor: User(User/ Administrator) | System |
| * + - 1. Click logout | * 1. Return home page |

Precondition

* User is in home page with their authentication

Post condition: None

**Name: Rate a movie**

Identifier: UC5

Input:

* Number of star from 1(worst) to 5(best) by clicking star in vote section

Output:

* New voting instance based on (movie, user) in database
* New order of trendy movie in home page

Basic course:

|  |  |
| --- | --- |
| Actor: User(User) | System |
| 1. Choose the movie | * 1. Display the page with movie details |
| 1. Click the play button | * 1. Count the time user is watching the movie |
| 1. Choose number of star | * 1. If the time is greater than a third of movie duration, enable vote section   2. Otherwise, disable vote section |
|  | 4.1. Update the voting instance in database |

Precondition:

* User has an account which was registered before

Post condition: None

**Name: View Movie List**

Identifier: UC6

Input: Movie category

Output: The movie list

Basic Course:

|  |  |
| --- | --- |
| Actor: Registered and Unregistered User | System: |
| 1. Click one specific Category | 1.1 Display movies (title and image) in the same Category |

Preconditions:

* User are viewing movie information

Post conditions: None

**Name: View Movie Detail**

Identifier: UC7

Inputs:

* Specific movie

Outputs:

* A page with movie’s introduction, length, actor, director, etc

Precondition:

* There must be some movie at the screen, maybe from search method or view movie list

Basic course:

|  |  |
| --- | --- |
| Actors: Registered user, admin, guest | System |
| * + - 1. Click on the movie on the page | 1.1 Display all buttons |
| * + - 1. Click on the “Detail” button | * 1. Display all the information of the movie |

Post condition: None

**Name: Add the movie to the favorite**

Identifier: UC8

Inputs:

* Specific movie

Outputs:

* The notification of successful adding the movie to favorite

Precondition:

* There must be some movie at the screen, maybe from search method or view movie list
* The user must be login with their username, password

Basic course:

|  |  |
| --- | --- |
| Actors: Registered user, admin | System |
| * + - 1. Click on the movie on the page | 1.1 Display all buttons |
| * + - 1. Click on “Add favorite” button | * 1. Add the movie into the database of favorite   2. The icon of favorite movie will be display at the right-side of the movie   3. Show the success notification |

Post condition:

* Add the movie into database

**Name: View the favorite**

Identifier: UC9

Inputs: Home page with authentication

Outputs:

* A page with your favorite movie

Precondition:

* The user must be login with their username, password

Basic course:

|  |  |
| --- | --- |
| Actors: Registered user, admin | System |
| Click on the “Favorite” in the menu | * 1. Display all the favorite movie |

Post condition:

* If there is no movie in the Favorite. There is a message “You should add films to the favorite”

**Name: Delete movies in the favorite**

Identifier: UC10

Inputs:

* Home page with authentication
* Favorite session

Outputs:

* The notification of successful removing the movies from Favorite

Precondition:

* The user must be login with their username, password
* There must be some films in the Favorite
* Already go to the Favorite page

Basic course:

|  |  |
| --- | --- |
| Actors: Registered user, admin | System |
| 1. Click on the movie | * 1. Display all buttons |
| 2. Click on “Remove from favorite” button | * 1. The Confirm message is display   2. There is button “Confirm” “Cancel” |
| 3. Click on “Confirm” button | * 1. Remove the movie from the database of favorite   2. The icon of favorite movie will be disappear at the right-side of the movie   3. Show the success notification |

Post condition:

* The movie is remove from the database

**Name: Add accounts**

Identifier: UC11

Inputs:

* Request that contain the information of users

Outputs:

* The notification of successfully adding the account

Precondition:

* The users must be login as administrator

Basic course:

|  |  |
| --- | --- |
| Actors: Admin | System |
| 1. Go to the “Request” tab | * 1. Display all the request   2. Display the button “Accept” and “Ignore” next to the request |
| 2. Click on “Accept” button | * 1. The account is add to the database   2. Show the success notification |

Post condition:

* If the admin clicks to the “Ignore” button the request is removed and the registration of the user is unsuccessful

**Name: Remove accounts**

Identifier: UC12

Inputs:

* Specific account

Outputs:

* The notification of successfully removing the account

Precondition:

* The users must be login as administrator
* There must be some accounts in the page

Basic course:

|  |  |
| --- | --- |
| Actors: Admin | System |
| 1. Click on the “Accounts” tab | * 1. Display the “Account” page   2. Display all the accounts   3. Display some buttons which is next to the accounts |
| 2. Click on “Remove” button | * 1. The Confirm message is display   2. There is button “Confirm” “Cancel” |
| 3. Click on “Confirm” button | * 1. The account is remove from the database   2. Show the success notification |

Post condition:

* The account is remove from the database

**Name: Add movies**

Identifier: UC13

Inputs:

* All the information of the movie
* The links to the movies

Outputs:

* The notification of successfully adding movies

Precondition:

* The users must be login as administrator

Basic course:

|  |  |
| --- | --- |
| Actors: Admin | System |
| 1. Click on the “Movie” tab | * 1. Display the “Movie” page   2. Display all movies   3. Display some buttons “Add new” “Remove”… |
| 2. Click on “Add new” button | * 1. Open the “Input information” page |
| 3. Input all of information |  |
| 4. Click “Submit” | 4.1 Update all information of the movie into database  4.2 The notification of successfully adding the movies |

Post condition:

* The information must be checked by the system, if the information is invalid, there is the message to show the error
* The information of the movie is added to the database

**Name: Remove movies**

Identifier: UC14

Inputs:

* Specific movies

Outputs:

* The notification of successful remove the movie

Precondition:

* The users must be login as administrator
* There must be some movies in the page

Basic course:

|  |  |
| --- | --- |
| Actors: Admin | System |
| 1. Click to the “Movies” tab | * 1. Display the “Movie” page   2. Display all the accounts   1.3 Display some buttons “Add new” “Remove”… |
| 2. Click on “Remove” button | * 1. The Confirm message is display   2. There is button “Confirm” “Cancel” |
| 3. Click on “Confirm” button | * 1. The movies is remove from the database   2. Show the success notification |

Post condition:

* The movie is remove from the database

**Name: Edit Profile**

Identifier: UC15

Inputs:

* Home page with authentication
* Profile page

Outputs:

* The notification of successfully edit profile

Precondition:

* The users must be login

Basic course:

|  |  |
| --- | --- |
| Actors: Registered user, admin | System |
| 1. Click to the “Profile” tab | * 1. Display the information of the user |
| 2. Edit the information | * 1. The Confirm message is display   2. There is button “Confirm” “Cancel” |
| 3. Click “Submit” button | * 1. The Confirm message is display   2. There is button “Confirm” “Cancel” |
| 4. Click on “Confirm” button | * 1. The information is edited   2. Show the success notification |

Post condition:

* The information must be checked by the system, if the information is invalid, there is the message to show the error instead of the “Confirm” message
* The accounts’ information is update to the database

**Name: Edit Movie details**

Identifier: UC16

Inputs:

* Home page with authentication
* Movie edit page

Outputs:

* The notification of successfully edit profile

Precondition:

* The users must be login as administrator
* There must be some movies in the page

Basic course:

|  |  |
| --- | --- |
| Actors: Admin | System |
| 1. Click to the “Movies” tab | * 1. Display the “Movie” page   2. Display all the accounts   1.3 Display some buttons “Add new” “Remove”… |
| 2. Click on “Edit” button | * 1. Display the information of the movies |
| 3. Edit the information |  |
| 4. Click “Submit” button | 4.1 The Confirm message is display  4.2 There is button “Confirm” “Cancel” |
| 4. Click on “Confirm” button | * 1. The movies’ information is updated to the database   2. Show the success notification |

Post condition:

* The information must be checked by the system, if the information is invalid, there is the message to show the error instead of the “Confirm” message, and return back to the page to edit the information
* The accounts’ information is update to the database

**Name: View Movie**

Identifier: UC17

Inputs:

* Home page with authentication

Outputs:

* The movie

Precondition:

* The users must be login as administrator

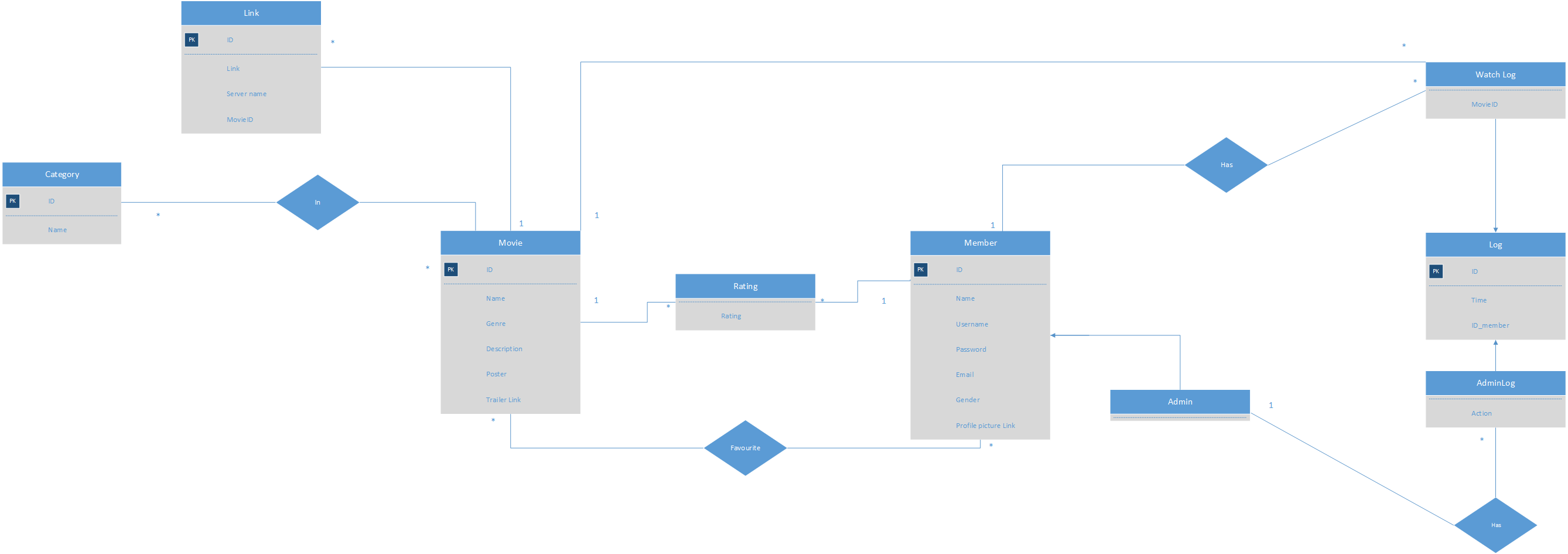
Basic Course:

|  |  |
| --- | --- |
| Actor: User | System |
| Click the play button | * 1. Play the movie in default server |
| 1. Choose the server | * 1. Play the movie in requested server |

Post condition: None

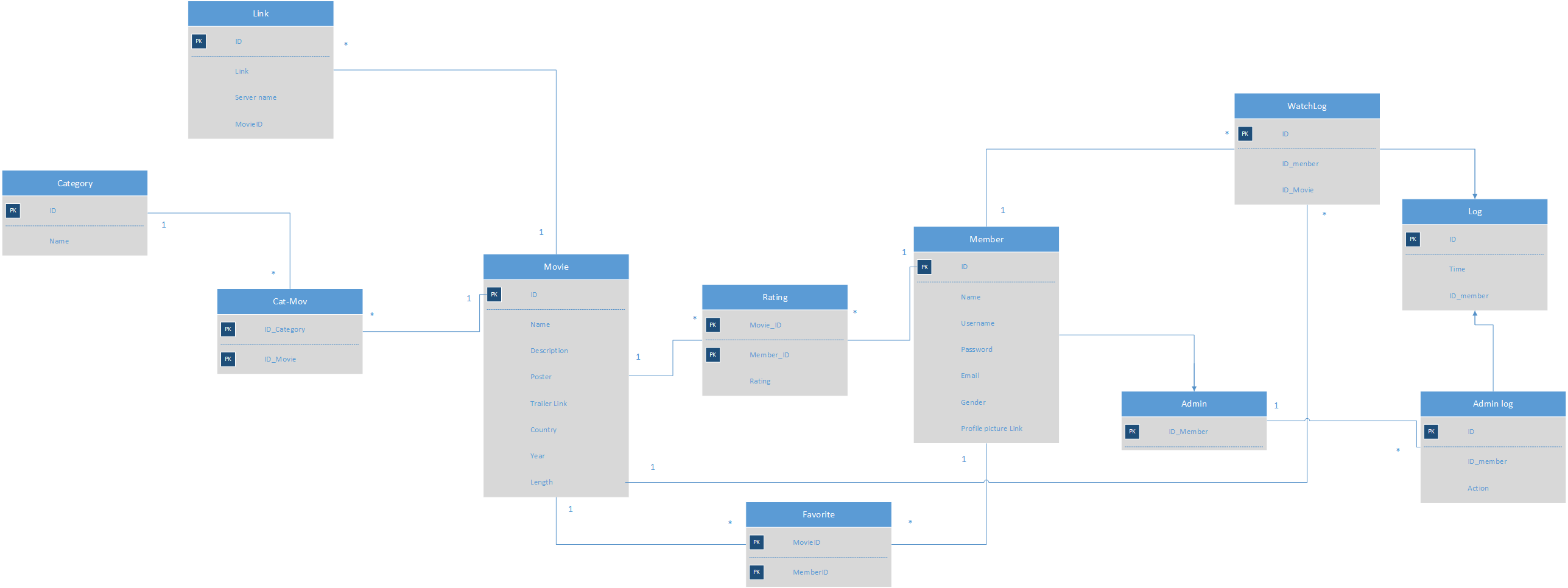
1. **Database design**

Entity Relation Diagram



*(Link to original file:* [*https://drive.google.com/open?id=123dH38S7K-90IN9Y4q0krVDEnIfv9RiG*](https://drive.google.com/open?id=123dH38S7K-90IN9Y4q0krVDEnIfv9RiG)*)*

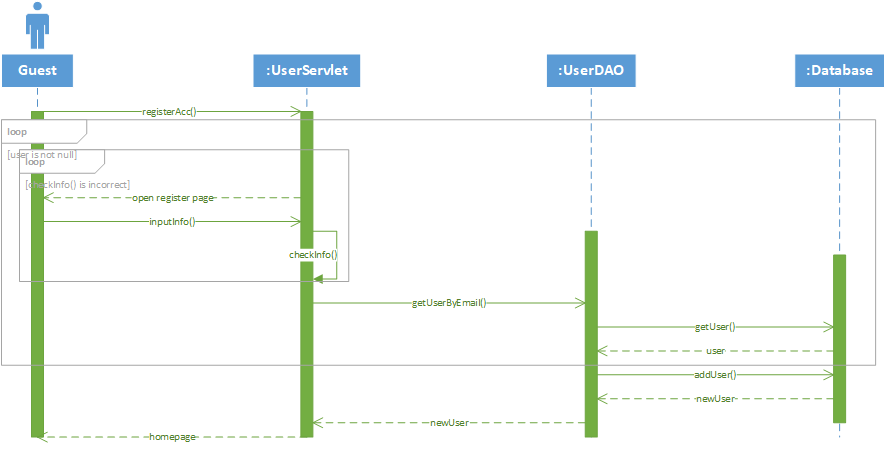
* + - 1. Database Schema



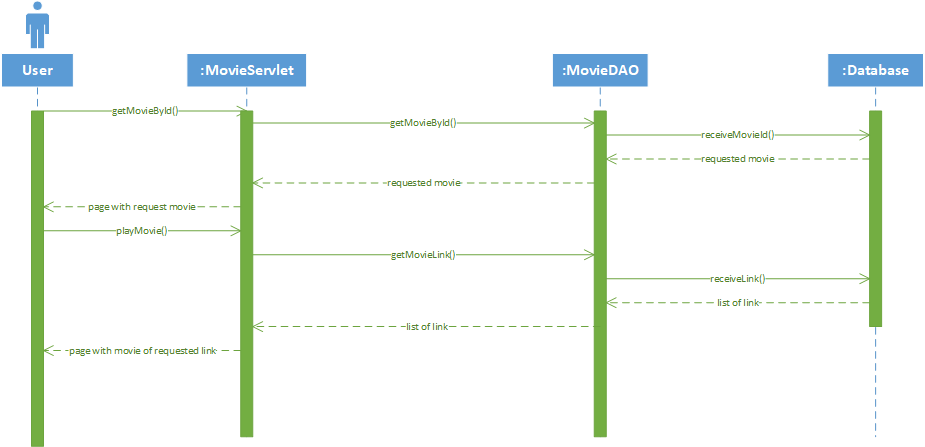
*(Link to original file:* [*https://drive.google.com/open?id=1sHqrJ6Uu99Rm6fTfq5skporuhvvmk6aF*](https://drive.google.com/open?id=1sHqrJ6Uu99Rm6fTfq5skporuhvvmk6aF)*)*

1. **Sequence Diagram**

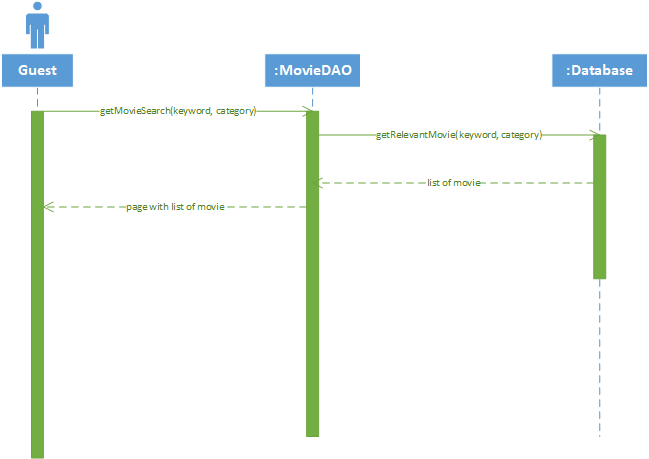
Register:

****

View Movie:



Search Movie:



1. **Class Diagram**

