



Movie Database  
Analysis & Design



COMP H2027 - Software Engineering & Testing  
Institute of Technology Blanchardstown  
2014

Lecturer: Simon McLoughlin

David Kelly: B00060572  
David Malone: B00060176

**Table of contents:**

1. Project Description.....	3
2. UML Use Case Diagram.....	4
3. Use Case Specifications.....	5
4. Class Description and Class Diagram.....	9
5. Sequence Diagrams.....	10
6. User Interface Design.....	15
6.1 Search	15
6.2 Login	15
6.3 Add Movie	16
6.4 Add Review	16
6.5 Flag Content	17
6.6 Remove Content	17
6.7 Search Member	18
6.8 Create Account	18
6.9 Search Results	19
6.10 Home Item List	19
7. User Iterface.....	20
8. References.....	20

## Project :

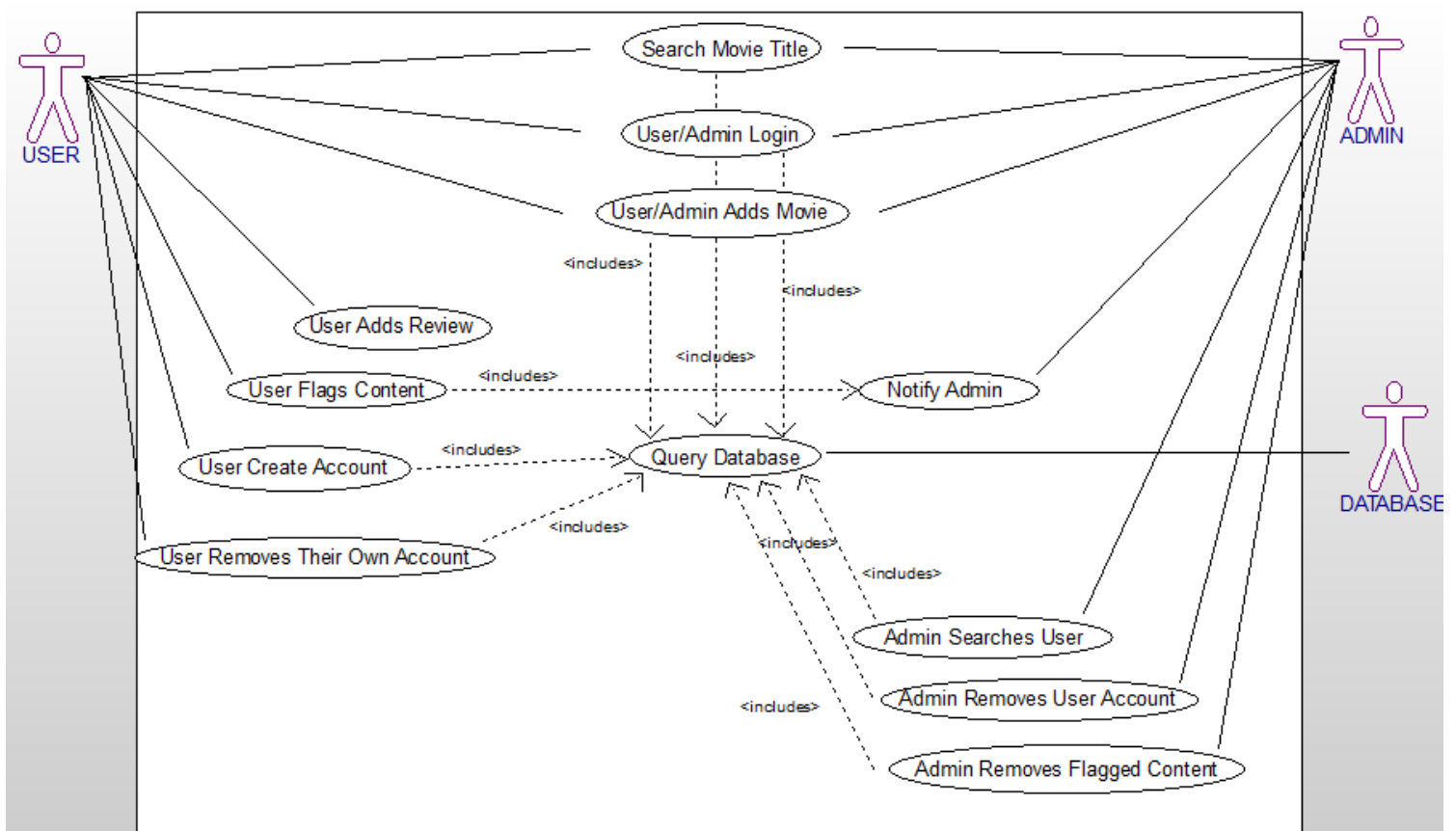
The goal of this project is to develop a movie database that anyone can search through and, if an account is created, add content to. It will be primarily a desktop PC application, with the potential to be ported to mobile systems such as Android or Apples IOS. This application will be presented to the user as a neat graphical user interface with a database back end containing the content. An executable of the application will be available for download with no install required.

## Scope:

The application will be designed for any platform that can run Java applications. It will be tested on PC, Mac OS and Linux with intent to develop for the Android and IOS mobile markets in the future. The application will not be available on the Playstation or Xbox platforms. A database will be stored in the cloud to house all of the content generated by the application users and the Java GUI front end of the application will be downloaded onto the users device to provide a client interface for database access.

## UML:

Unified Modeling Language allowed us to create and plan our program with efficiency and relative ease. Creating a **use case diagram** first gave the team an impression of how the final system will look when completed and implemented. The **use case** helped us as a team understand the classes and methods needed for the system to interact with users, something that is integral to the systems functionality. After creating the **use case diagram** we then decided to create a **class diagram**. The **class diagram** helped us structure our classes with methods and connect the classes that need to interact with each other. In order for the system to operate as a user to database sequence with returning values displayed through the GUI, it was imperative that we were methodical and analytical to ensure there was no unused methods and that the system was as functional and streamlined as possible. In order to confirm the methods we implemented in the **class diagram** served a purpose and were truly functional we needed to do **sequence diagrams**. The **sequence diagrams** were the final act in confirming our system had all the classes and methods needed to be operational.



**Use case specifications:****1. User Searches Movie Title****Primary Path**

1. User input's movie title and clicks search button
2. Program queries the database
3. Program finds a matching result in the database
4. Program returns all the information under that title
5. Program displays the results

**Alternative Path**

- 3a. Program does not find the results in the database
- 3b. Program returns null value
- 3c. Program displays search fail
- 3d. User is returned to search bar page

**2. User Creates Account****Primary Path**

1. User selects create account option
2. User enters their details
3. Program verifies email format and password fields
4. Program queries the database for duplicate account details
5. Program displays successful account creation
6. Program sends an email to the user's email address
7. Program logs the user in to their account
8. Program redirects the user to the home screen

**Alternative Path**

- 3a. User has entered incorrect details
- 3b. Program prompts the user to re-enter correct details
- 3c. Program rechecks the details
- 4a. Duplicate details are found in the database
- 4b. Program prompts the user to change their details
- 4c. Program re-checks the details

**3. Member/Admin Login****Primary Path**

1. Member navigates to login page
2. Member enters their details
3. Program queries the database for correct details
4. Program finds the correct details and retrieves them from the database
5. Program applies the correct permissions relative to the account
6. Member is redirected to the home page

**Alternative Path**

- 3a. Program finds no matching details in the database
- 3b. Program shows a login failed window
- 3c. Program prompts the user to re-enter the correct details
- 3d. Program re-checks the details

## 4. Member/Admin Adds Movie

### Primary Path

1. Member logs in
2. Member navigates to the create movie profile page
3. Member enters all the required fields on the form and clicks submit button
4. Program checks that all the required fields are entered
5. Program queries the database for a duplicate movie title
6. Program finds no duplicate movie title
7. Program adds the form information to the database
8. Member is redirected to the new movie profile page

### Alternative Path

- 4a. Program finds the required fields have not been entered
- 4b. Program prompts the user to enter the required fields
- 4c. Program rechecks the fields
- 5a. Program finds a duplicate title existing in the database
- 5b. Program notifies the user that the title exists and shows movie information from database
- 5c. Program prompts the user to change their title or cancel operation
- 5di. Program rechecks the title      5dii. User is redirected to the create movie profile page

## 5. Member Adds Review

### Primary Path

1. Member logs in
2. Member searches for movie
3. Member enters page of movie they wish to review
4. Member clicks on the add review button
5. Member enters their review in the text area provided
6. Member clicks the add review button when finished
7. Program adds and saves the review to the database
8. Program redirects the user to the movie title page

### Alternative Path

- 2a. Database does not have the movie title to be reviewed
- 2b. Program prompts the user to add the title or cancel the search
- 2ci. Member adds the movie      2cii. Member is redirected to the home page
- 5a. Member clicks the cancel search button
- 5b. Program redirects the user to the movie title page

## 6.Member Flags Content

### Primary Path

- 1.Member logs in
- 2.Member searches for content
- 3.Member navigates to the page containing the content they wish to flag
- 4.Member clicks on the flag icon related to the content
- 5.Program notifies user with option to confirm or cancel flagging the content
- 6.Member clicks confirm
- 7.Program sends a notification email to the admin with the content details
- 8.Program returns the user to the content page

### Alternative Path

- 5a.Member cancels the flagging option
- 5b.Program returns the user to the content page

## 7.Admin Removes Flagged Content

### Primary Path

- 1.Admin logs in
- 2.Admin navigates to the flagged content
- 3.Admin selects flagged content
- 4.Admin clicks the remove content button on flagged content
- 5.Program shows JOptionPane to confirm content removal
- 6.Admin clicks remove
- 7.Program permanently removes the content from the database
- 8.Program returns the admin to content page

### Alternative Path

- 6a.Admin selects cancel option
- 6b.Program redirects the admin back to content page

## 8.Admin Searches Member

### PrimaryPath

- 1.Admin logs in
- 2.Admin navigates to the search member option
- 3.Admin enters username in search bar and clicks search
- 4.Program searches the username in the database
- 5.Program finds the username
- 6.Program returns the account page for that user

### Alternative Path

- 5a.Program does not find that username in the database
- 5b.Program notifies the admin that no username is found
- 5c.Program prompts the admin to research the username or cancel
- 5di.Program rechecks the database for username
- 5dii.Program redirects the admin back to the search username page

## 9. Admin Removes User Account

### Primary Path

1. Admin logs in
2. Admin searches members
3. Admin clicks the remove member account button
4. Program shows a JOptionPane to confirm member removal
5. Admin clicks remove button
6. Program permanently removes the members account information from the database
7. Admin is redirected to the search member page

### Alternative Path

- 5a. Admin selects cancel option on JOptionPane
- 5b. Program returns the admin to the members account page

## 10. Member Removes Their Own Account

### Primary Path

1. Member logs in
2. Member navigates to member account page JMenuItem
3. Member clicks the remove account button
4. Member is prompted to enter their password or cancel the operation
5. Member enters their password
6. Program queries the database for matching the password
7. Program finds the matching password in the database
8. Program permanently removes the member account information from the database
9. Member is redirected to the home page search window

### Alternative Path

- 5a. Member clicks the cancel button
- 5b. Program returns the member to their account page
- 6a. Program finds no matching password in the database
- 6b. Program prompts the member to re-enter their password
- 6c. Program rechecks the database for the password



## Classes & Attributes:

**Account:** AccountID, Username, Password, Email

**GUI:** Input, SearchString

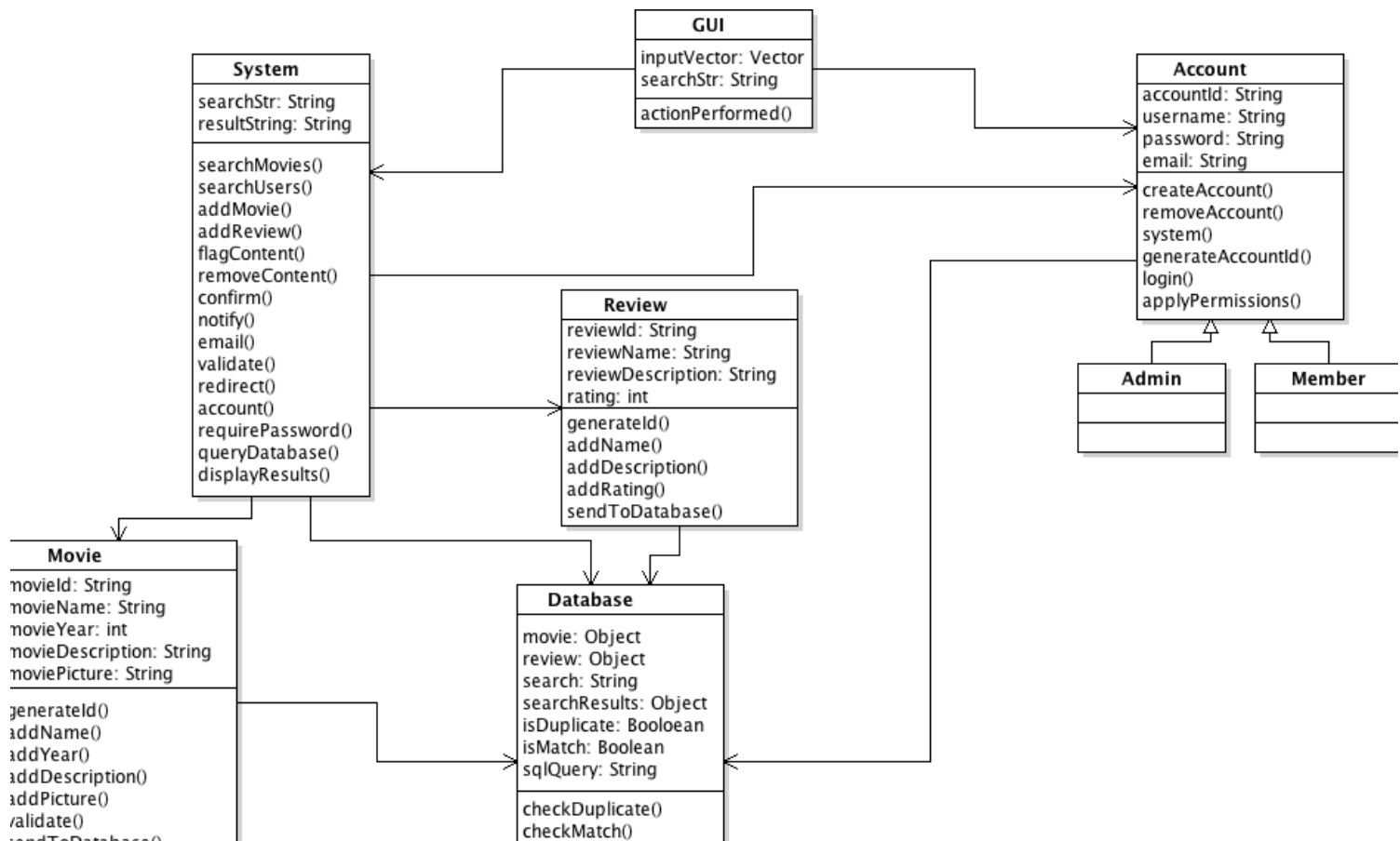
**System:** SearchString, ResultsString

**Movie:** MovieID, MovieName, MovieYear, MovieDescription, MovieImage

**Review:** ReviewID, ReviewName, ReviewDescription, Rating

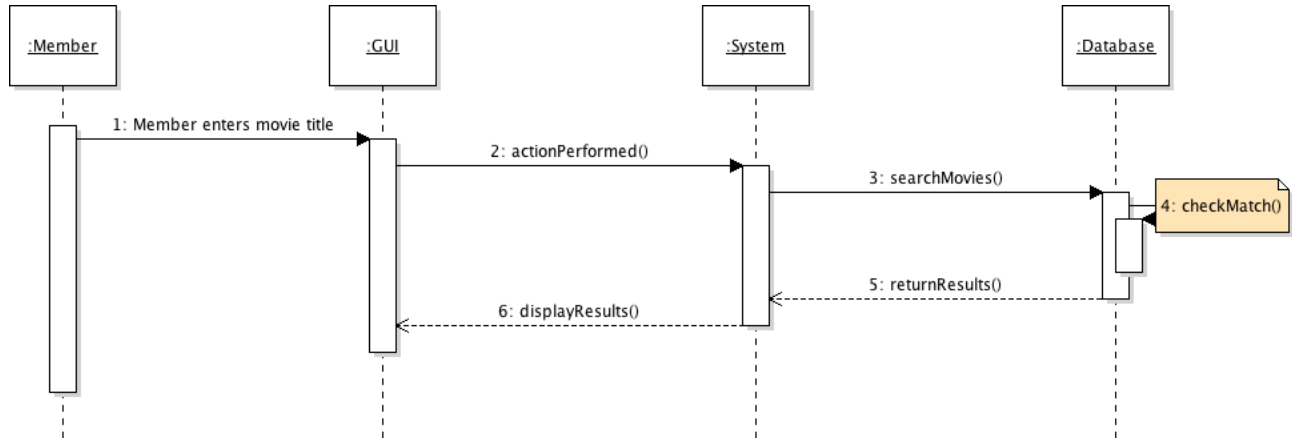
**Database:** Movie, Review, Search, SearchResults, IsDuplicate, IsMatch, SqlQuery

## Class Diagram:

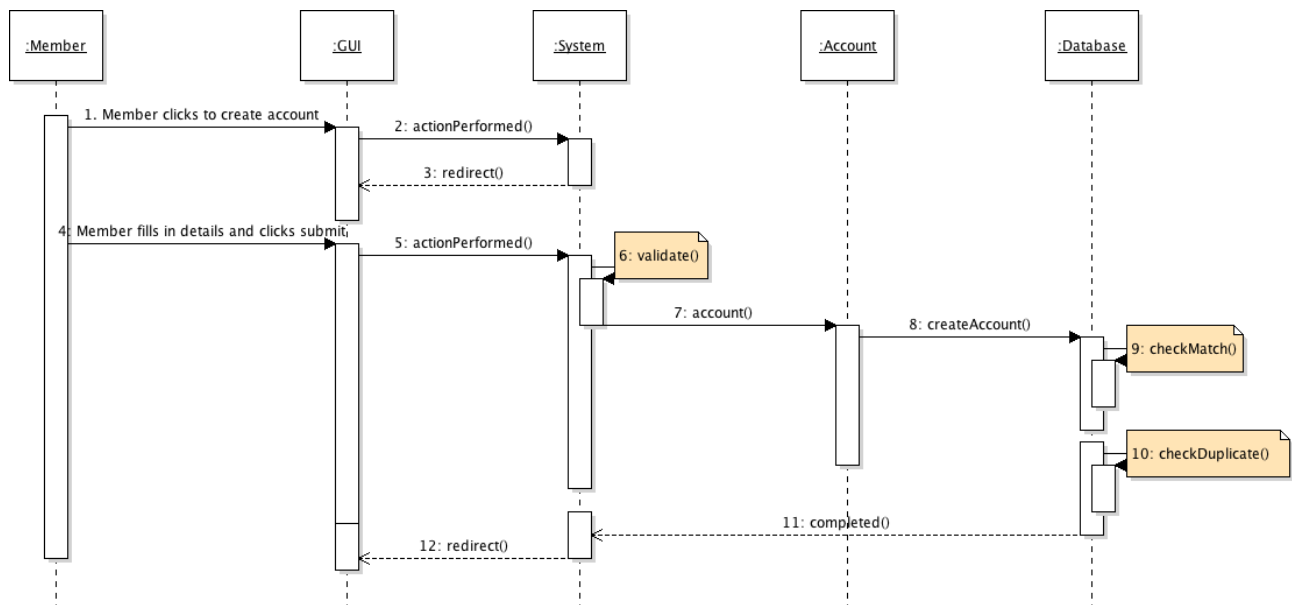


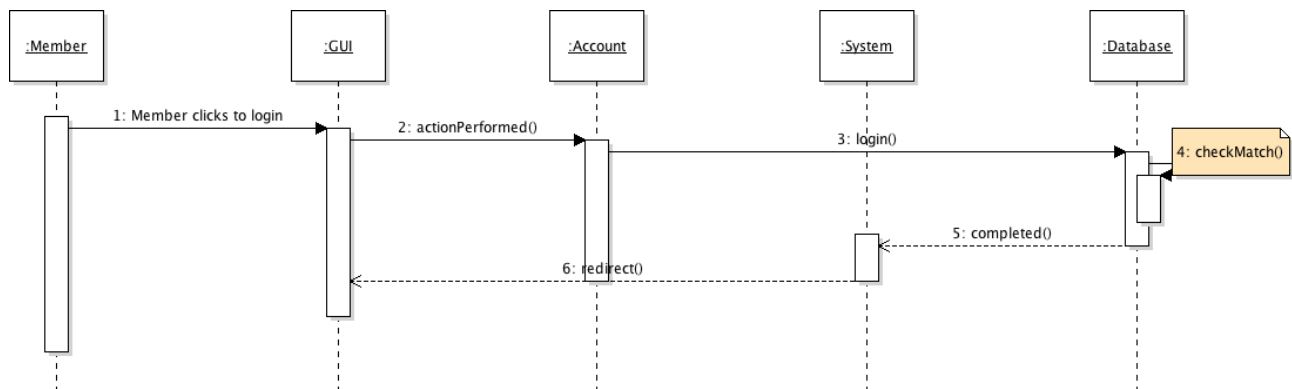
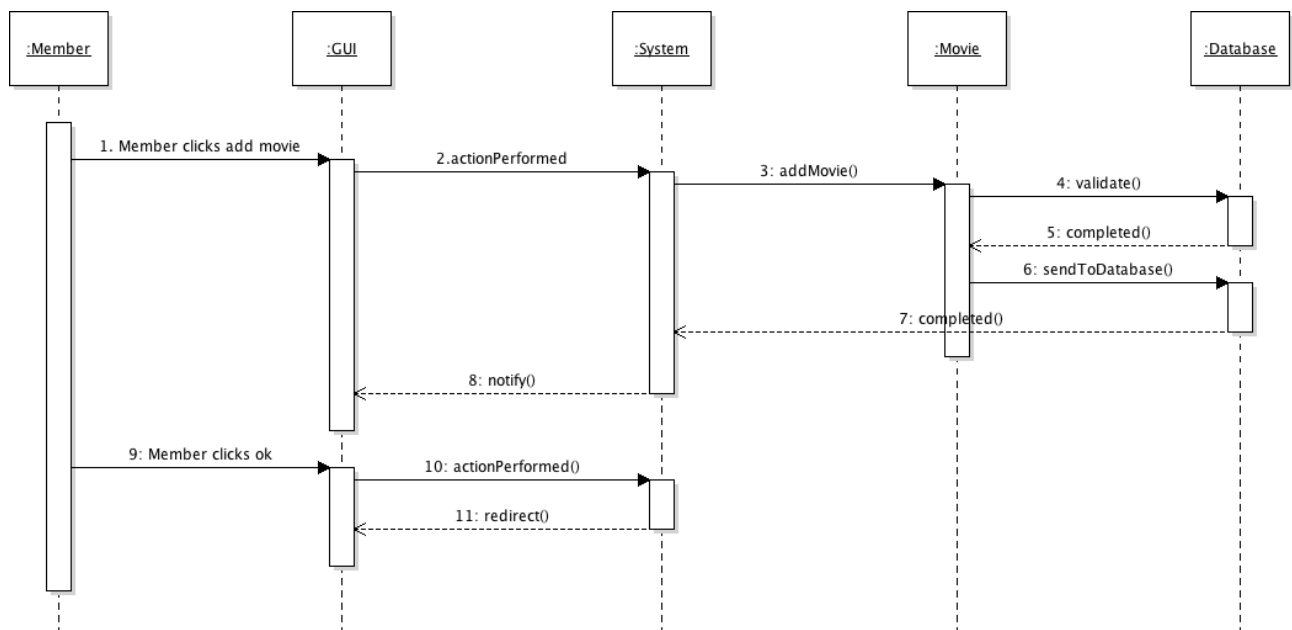
## Sequence Diagrams:

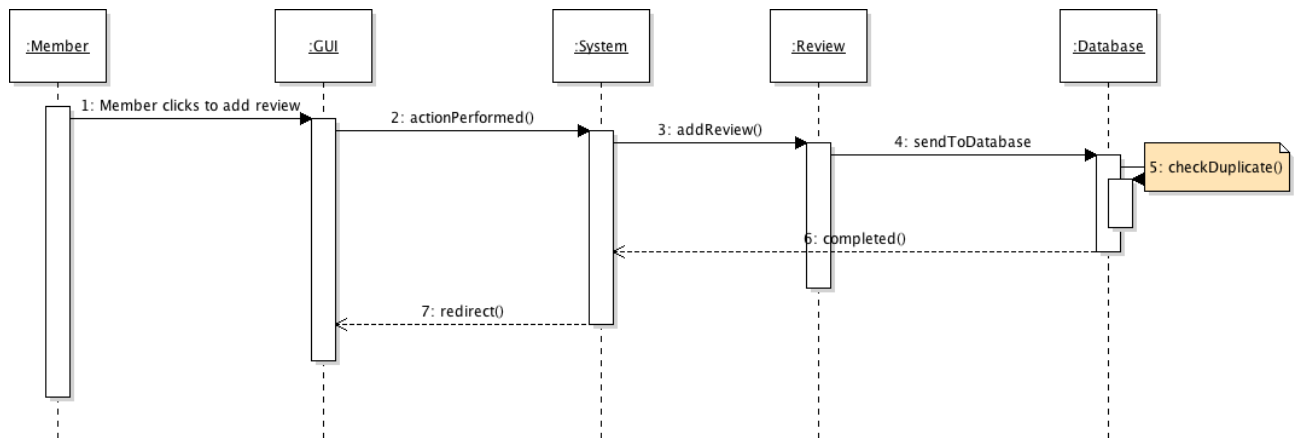
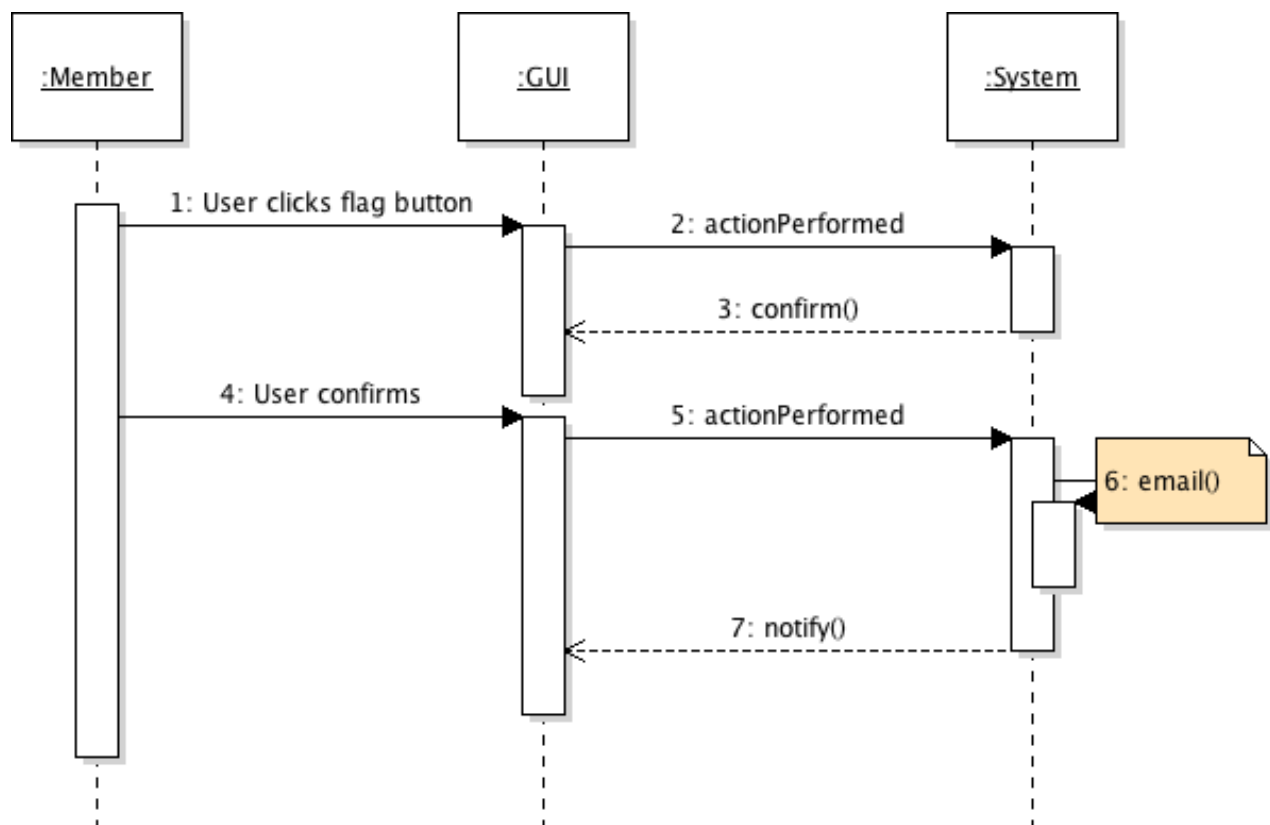
### Search Movie:

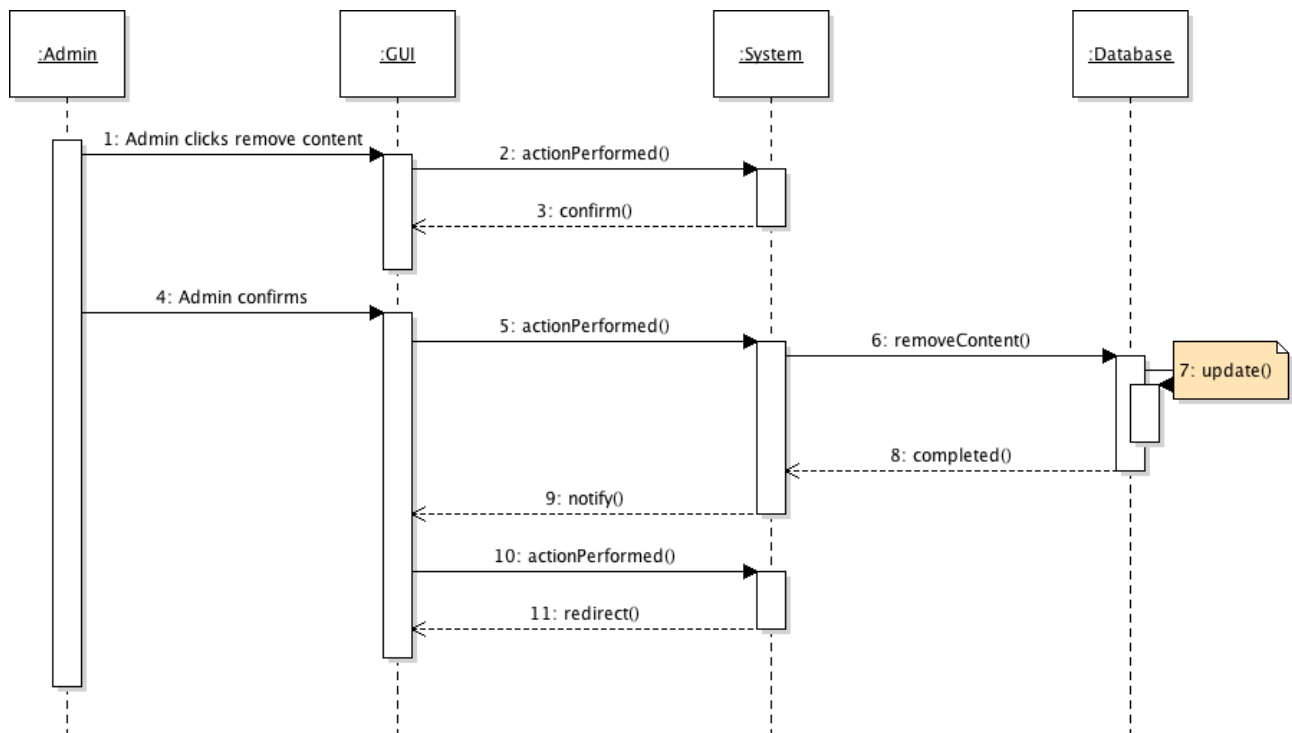
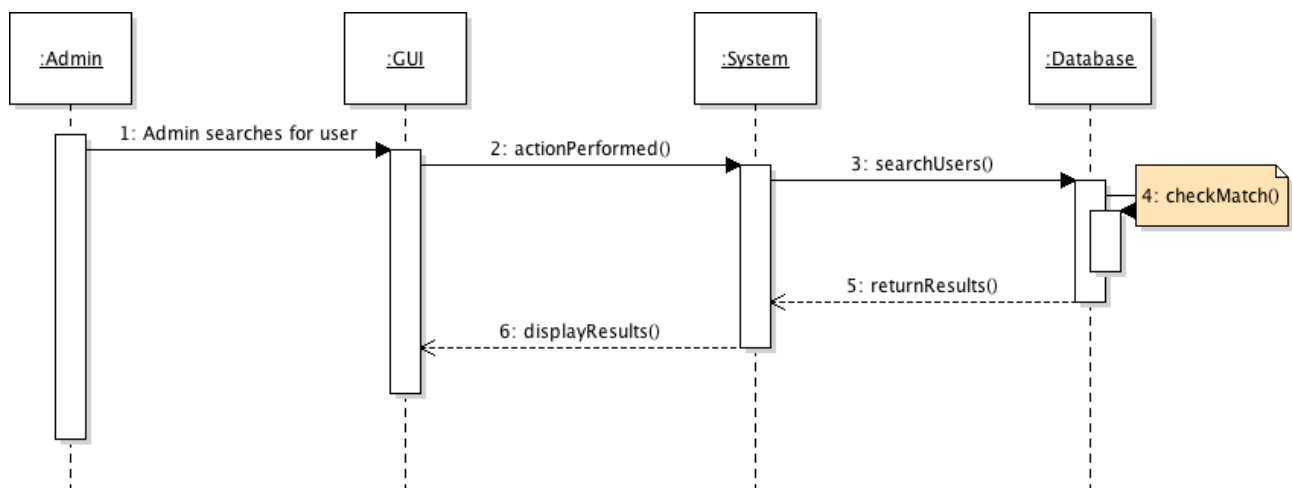


### Create Account:

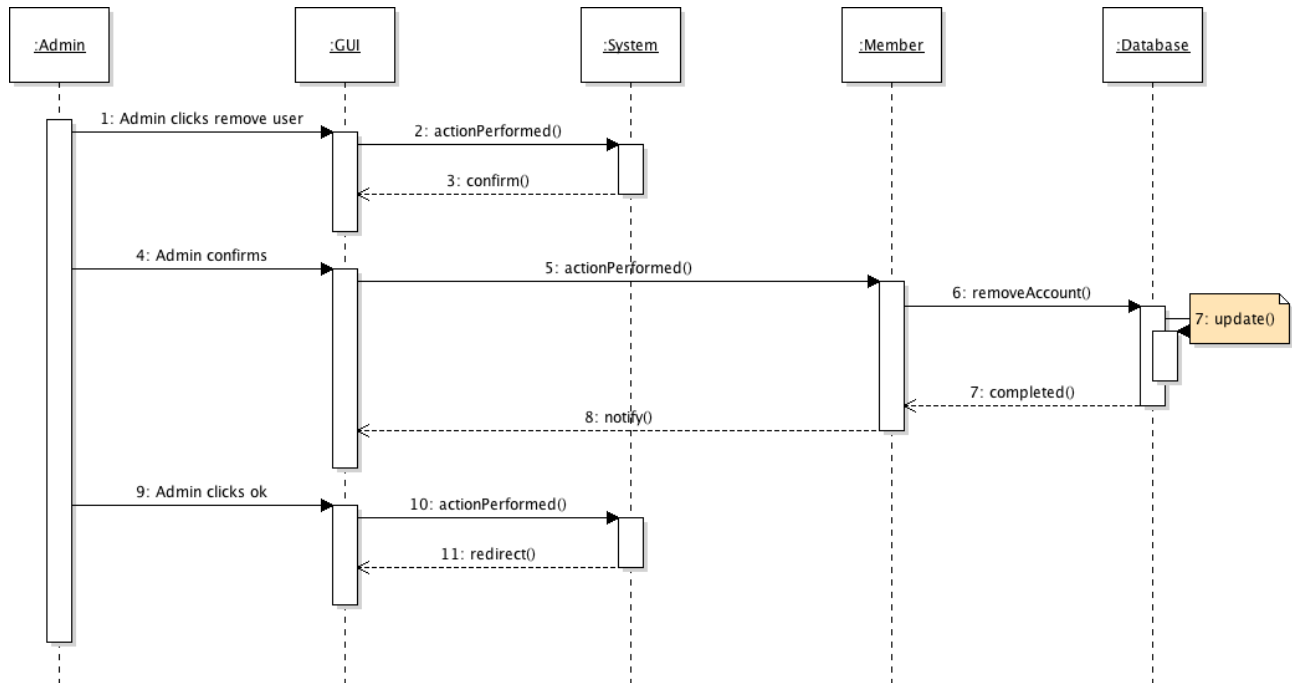


**Login:****Add Movie:**

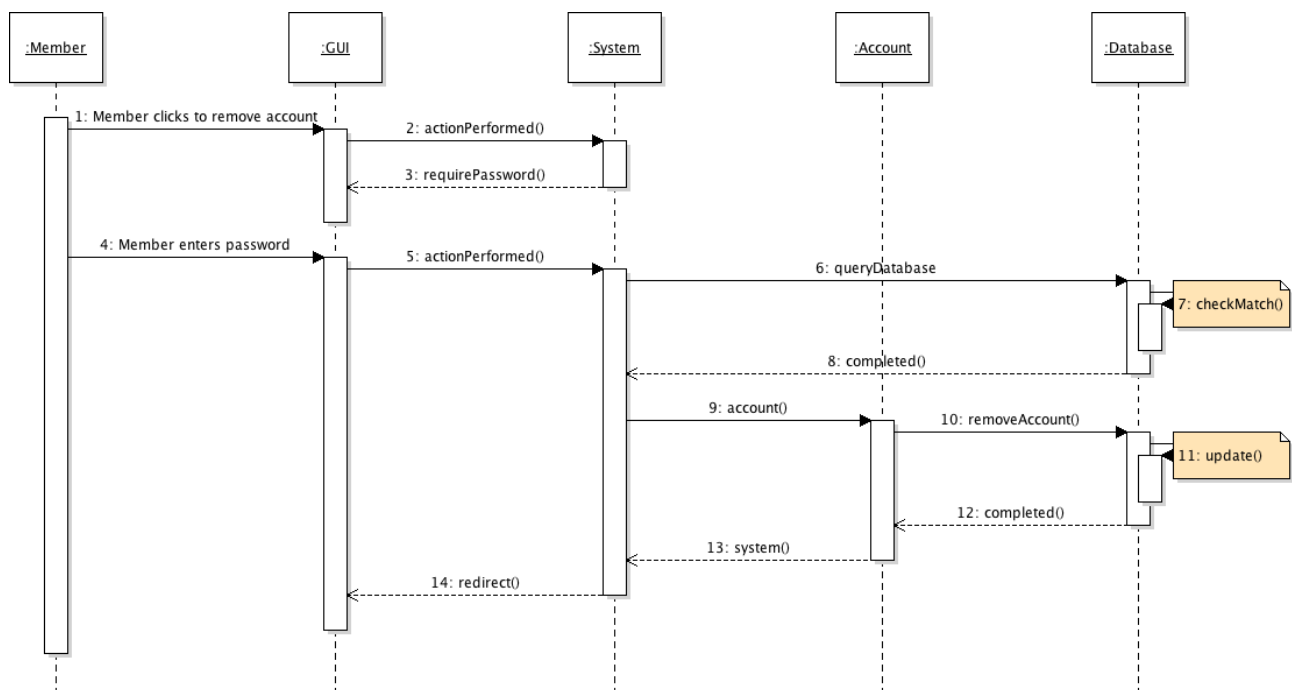
**Add Review:****Flag Content:**

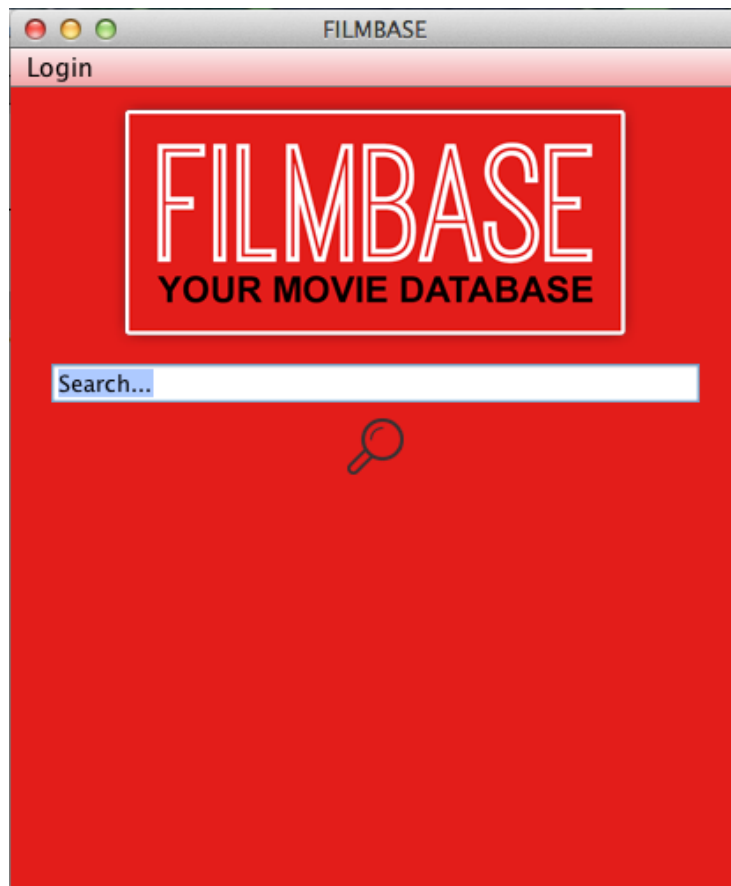
**Remove Content:****Admin Searches Member:**

## Admin Removes User:

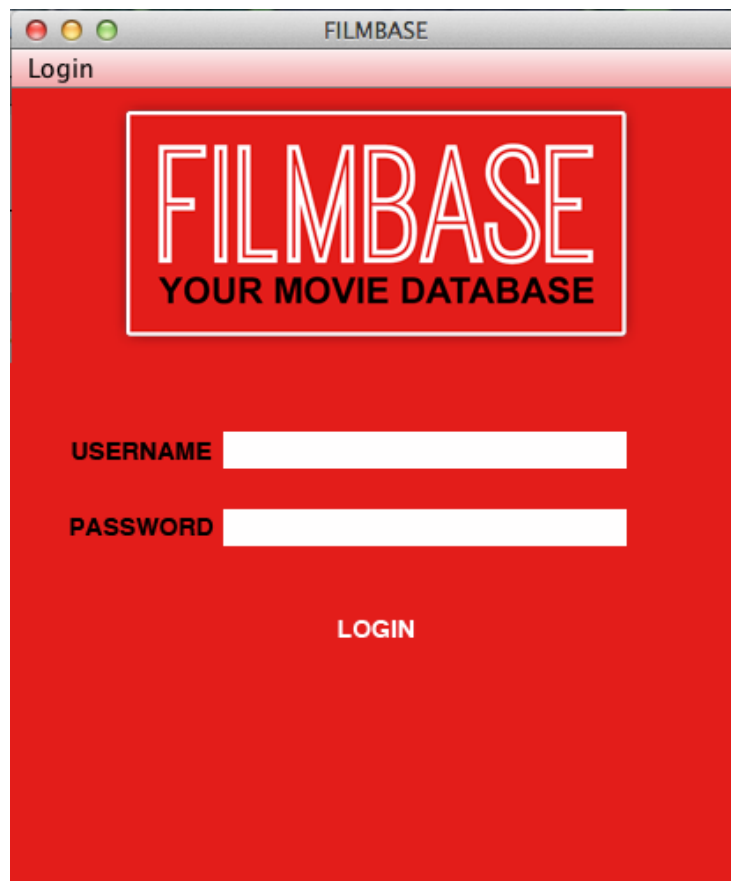


## User Removes Account:



**User Interface Design****Home/Search:**

A screenshot of a web browser window titled "FILMBASE". The browser's address bar shows "Login". The page has a solid red background. At the top center, there is a white rectangular box containing the word "FILMBASE" in a large, white, outlined, sans-serif font, with the text "YOUR MOVIE DATABASE" in a smaller, black, sans-serif font directly below it. Below this box is a white search input field with the placeholder text "Search...". Centered below the search field is a magnifying glass icon.

**Login:**

A screenshot of a web browser window titled "FILMBASE". The browser's address bar shows "Login". The page has a solid red background. At the top center, there is a white rectangular box containing the word "FILMBASE" in a large, white, outlined, sans-serif font, with the text "YOUR MOVIE DATABASE" in a smaller, black, sans-serif font directly below it. Below this box, the text "USERNAME" is followed by a white input field. Below that, the text "PASSWORD" is followed by a white input field. At the bottom center of the page, the word "LOGIN" is displayed in a white, sans-serif font.

**Add Movie:**

The screenshot shows a web browser window titled 'FILMBASE'. The page has a red background and a 'Home' header. It contains a form with the following fields: 'TITLE' (a single-line text input), 'YEAR' (a single-line text input), 'DIRECTOR' (a single-line text input), 'PLOT' (a multi-line text area), and 'COVER' (a button labeled 'choose file....'). At the bottom right of the form is an 'Add Movie' button.

**Add Review:**


The screenshot shows a web browser window titled 'FILMBASE'. The page has a red background and a 'Review' header. At the top, there is a logo that reads 'FILMBASE YOUR MOVIE DATABASE'. Below the logo, there is a movie poster for 'FARGO' by Joel & Ethan Cohen. To the right of the poster, the text reads: 'Fargo (1996)', 'Joel & Ethan Cohen', and a paragraph: 'Jerry Lundegaard's inept crime falls apart due to his and his henchmen's bungling and the persistent police work of the quite pregnant Marge Gunderson.' Below this text is a large white rectangular area for the review. At the bottom right of the page is an 'Add Review' button.



**Flag Content:**

FILMBASE

Login




**Argo (2013)**  
**Ben Affleck**

---

Acting under the cover of a Hollywood producer scouting a location for a science fiction film, a CIA agent launches a dangerous operation to rescue six Americans in Tehran.


Reviews

I really enjoyed this movie. I was surprised to find the Ben Affleck both directed and starred in Argo. I was a gripping movie and well executed



-username

Not Good!..... Do not watch this movie! I don't know how this won awards.....



-username2









**Remove Flagged Content:**

FILMBASE

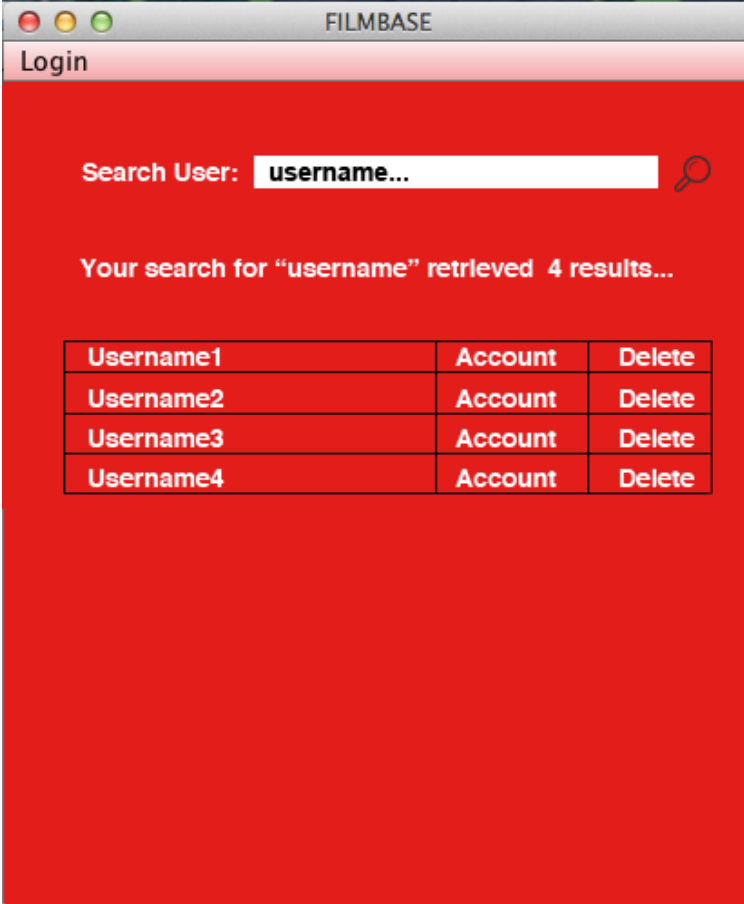
Login

FILMBASE

YOUR MOVIE DATABASE

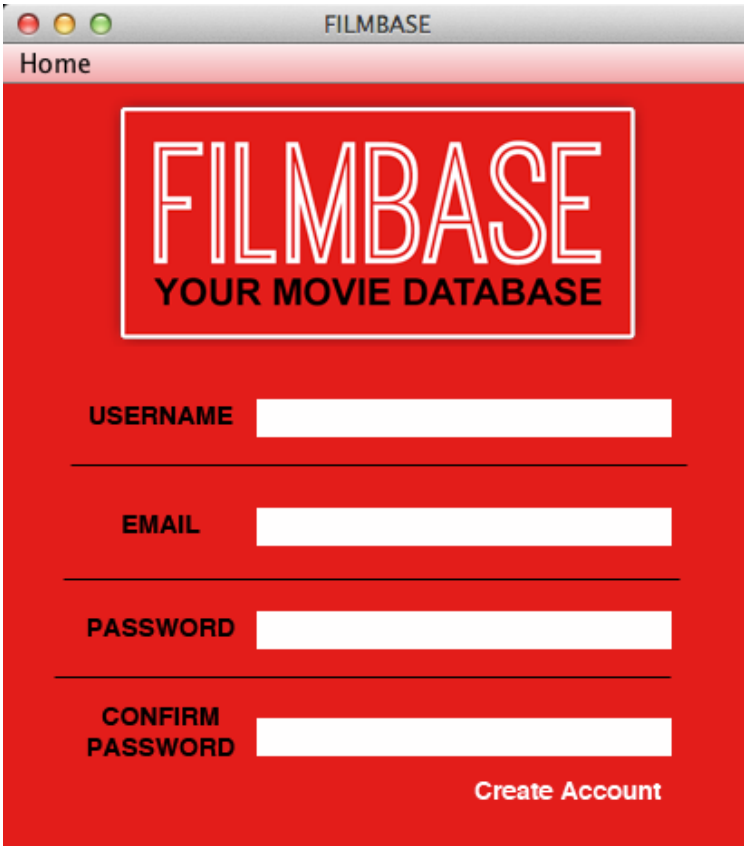
username		Review	Remove
username2		Review	Remove
username3		Review	Remove
username4		Review	Remove
username5		Review	Remove
username6		Review	Remove
username7		Review	Remove
username8		Review	Remove

Confirm

**Search Member:**

A screenshot of a web browser window titled "FILMBASE". The page has a red background and a "Login" header. It features a search bar with the text "Search User: username..." and a magnifying glass icon. Below the search bar, it says "Your search for 'username' retrieved 4 results...". A table displays the search results with four rows, each containing a username, the word "Account", and a "Delete" link.

Username1	Account	Delete
Username2	Account	Delete
Username3	Account	Delete
Username4	Account	Delete

**Create Account:**

A screenshot of a web browser window titled "FILMBASE". The page has a red background and a "Home" header. It features a large logo that says "FILMBASE YOUR MOVIE DATABASE". Below the logo, there are four input fields for "USERNAME", "EMAIL", "PASSWORD", and "CONFIRM PASSWORD". A "Create Account" button is located at the bottom right of the form.

**USERNAME**

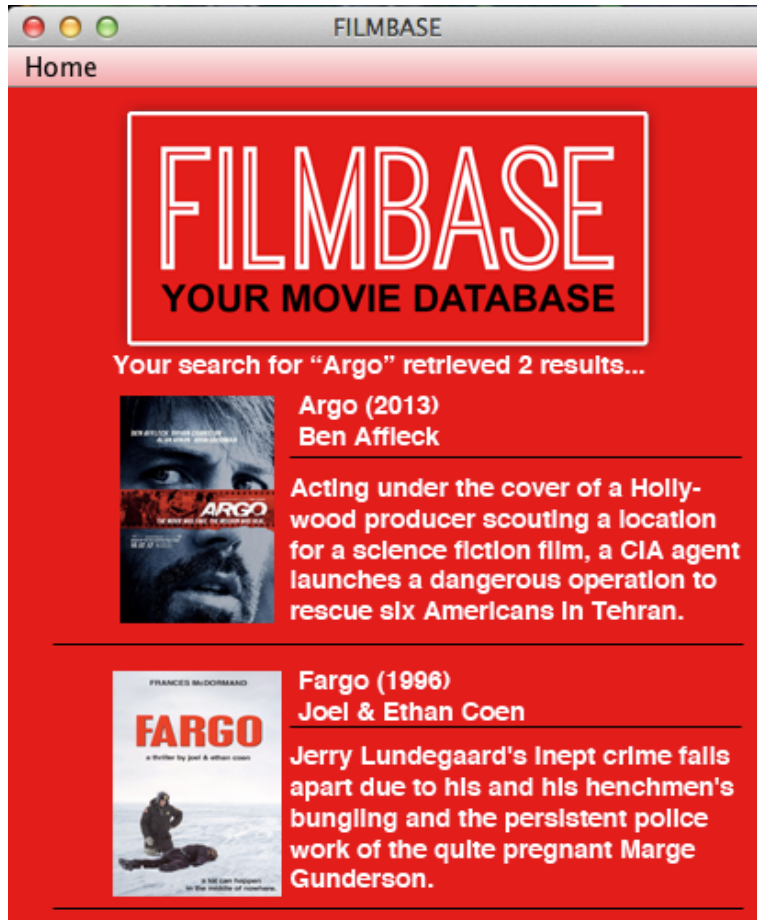
**EMAIL**

**PASSWORD**

**CONFIRM PASSWORD**

Create Account

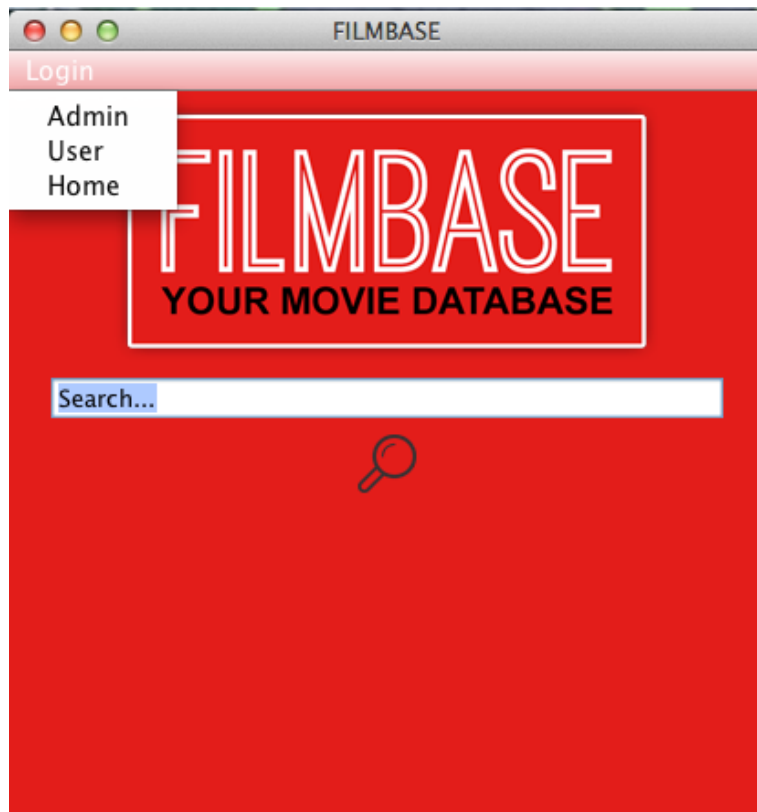
## Search Results:



The screenshot shows a web browser window titled 'FILMBASE'. The page has a red background. At the top, there's a 'Home' link. Below it, the 'FILMBASE YOUR MOVIE DATABASE' logo is displayed. A message states: 'Your search for "Argo" retrieved 2 results...'. Two movie results are shown:

- Argo (2013)**  
Ben Affleck  
Acting under the cover of a Hollywood producer scouting a location for a science fiction film, a CIA agent launches a dangerous operation to rescue six Americans in Tehran.
- Fargo (1996)**  
Joel & Ethan Coen  
Jerry Lundegaard's inept crime falls apart due to his and his henchmen's bungling and the persistent police work of the quite pregnant Marge Gunderson.

## Home Item List:



The screenshot shows the FILMBASE home page. It features a red background with the 'FILMBASE YOUR MOVIE DATABASE' logo. A 'Login' link is visible in the top left. A dropdown menu is open, showing 'Admin', 'User', and 'Home' options. Below the logo, there is a search bar with the placeholder text 'Search...' and a magnifying glass icon.

The user interface is an integral part of our system. The interface has to be consistent on all levels for the user and the administrator alike . The GUI is the link between the user and the system and must be accessible and easily navigated at all times.

We have taken the approach of separating the GUI to it's own class and calling methods from the system class. This modularises the code and allows methods to be used by many different classes.

The GUI we plan on implementing is very clean with only a few options for the user to interact with on any given screen. The menu option is consistent across every screen and is the users anchor for navigation. Allowing the user to access many different options from wherever they are in the system(dependent on their permissions).

**References:**

Fowler, M., 2004. UML Distilled: A Brief Guide To The Standard Object Modeling Language.

Pearson Education, Inc. ISBN 0-321-19368-7.

Larman, C., 2004. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development. 3rd ed.

Prentice Hall, ISBN 978-0131489066.