|  |
| --- |
| **Vibeflix**  **Group 4 - Vibe** |

**Bianca Cristina Badeu, 315270**

**Dominika Ingrid Ignatowicz, 315277**

**Maria-Elisabeta Mihai, 315231**

**Luis Daniel Guzman, 315194**

**Henrik Kronborg Pedersen**

**Steffen Vissing Andersen**

**characters**

**Software Technology Engineering**

**Semester 2**

**29.05.2021**

**Table of content**

[1 Abstract 4](#_Toc104833523)

[2 Introduction 5](#_Toc104833524)

[3 Analysis 8](#_Toc104833525)

[3.1 Requirements 8](#_Toc104833526)

[3.2 Functional Requirements 10](#_Toc104833527)

[3.3 Non-Functional Requirements 13](#_Toc104833528)

[4 Design 14](#_Toc104833529)

[5 Implementation 15](#_Toc104833530)

[6 Test 20](#_Toc104833531)

[6.1 Test Specifications 20](#_Toc104833532)

[6.1.1 Log-in 20](#_Toc104833533)

[6.1.2 Manage Movies 21](#_Toc104833534)

[6.1.3 Manage rentals 23](#_Toc104833535)

[6.1.4 Give feedback 23](#_Toc104833536)

[6.2 TEST CASES 26](#_Toc104833537)

[6.2.1 ADD MOVIE 26](#_Toc104833538)

[6.2.2 EDIT MOVIE 29](#_Toc104833539)

[6.2.3 REMOVE MOVIE 33](#_Toc104833540)

[6.2.4 RENT 34](#_Toc104833541)

[6.2.5 CANCEL 36](#_Toc104833542)

[6.2.6 Display 49](#_Toc104833543)

[6.2.7 RATE 50](#_Toc104833544)

[6.2.8 REVIEW 52](#_Toc104833545)

[7 Results and Discussion 81](#_Toc104833546)

[8 Conclusions 82](#_Toc104833547)

[9 Project future 83](#_Toc104833548)

[10 Sources of information 84](#_Toc104833549)

[11 Appendices 85](#_Toc104833550)

**List of figures and tables**

Optional

# Abstract

This semester the topic was chosen by the group members, based on the Problem Based Learning guidelines and the assistance of the supervisors to approve the said topic. For this theme, the problem of the availability of movie rental systems was presented and finding a way to deal with the customer’s requirements was a need.

This problem was solved by creating a system called “VibeFlix” which as intended, is a movie rental service with the possibility of maintenance and account creation and verification. The maintenance side allows the administrator to add, remove and edit movies present in the system. However, not only does the service update the movie listing in real-time but the database where the information is stored is also updated. In contrast, the user side allows for the creation of an account and logging in, afterwards a selection of movies is presented in the way of the trending movies and top-rated movies by the system users. The users can rent these movies and see the rentals in the profile section, allowing them to keep track of them in an easier manner.

“VibeFlix” takes into account all of the requirements identified by the group and implements them functionally. Therefore, it can be said that the product meets the initial and final expectations as well as satisfying the customer’s needs.

# Introduction

Currently, the world is moving at a faster pace that requires society to adapt to a different lifestyle. Therefore, personal time is limited, requiring people to reduce the time spent on trivial matters, such as: grabbing the TV remote and looking for that one channel that has the desired movie or researching if that one movie fits the current mood.

Movie rentals were a popular business in the past. With the invention of VHS, companies like Blockbuster and RedBox had a big impact on the entertainment industry. This industry brought a rental system for movies, providing the opportunity to watch a variety of movies for a lower price during the rental period. Nowadays, with the world becoming digitalized, streaming services have replaced the need for physical movie rentals. Movies can now be watched from any place at any time with an ever-growing selection.

The next step in improving the way movies or shows are watched lies in the current technologies available to humankind, for example, the internet. The internet is one of the most innovative technologies of the XX century. Being introduced in the year 1983 as a way for systems to communicate with each other, it has evolved into much more than that. It now offers a range of possibilities, from the use of social media to video calls with people from around the world. (A Brief History of the Internet, n.d.)

Although the internet is fascinating, a particular focus may be brought upon the possibility of streaming content. There are services available such as Netflix or HBO, where movies and TV shows are available at any time and anywhere if there is an internet connection. Most of them showcase the top 100 movies trending at a particular time based on views and ratings.

Here is where the future for movies and entertainment lies, the possibility of watching the movie from your phone, selecting a specific movie, and finally watching it wherever it is convenient brings entertainment to a whole new level. However, no system is perfect, for instance, Netflix does not have a scored based review system for a movie to determine its quality, nor does it offer the possibility of renting a movie.

This project's topic has been selected by the group members and approved by the supervisors. The scene being the current market for movie rentals, as of today there is not a vast amount of options for online movie rentals, which makes this market inaccessible for customers of this sector. Once again, an approach to deal with this issue had to be found through Problem Based Learning to determine the needs of the customers.

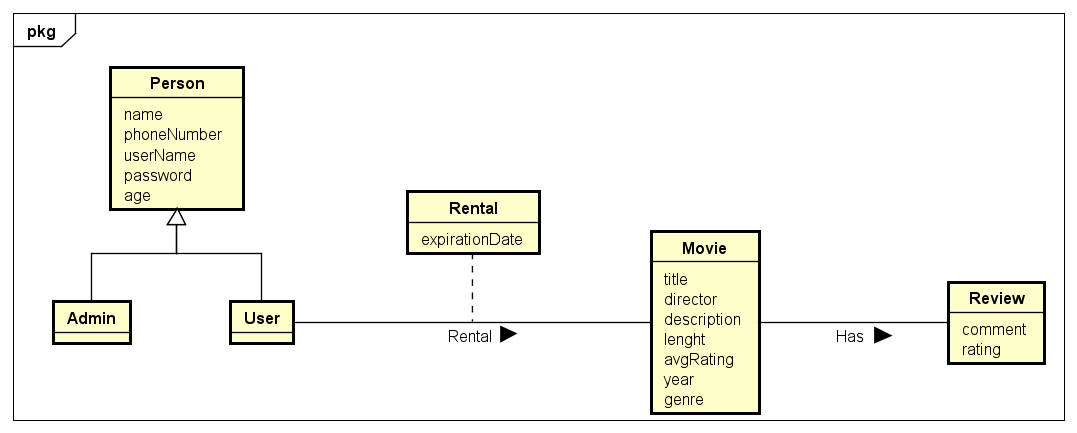
For the purpose of this project, the group was able to determine two relevant customers. The administrator, who would like to manage the system, make any necessary updates and be able to log in with the appropriate credentials, and the users, which are the customers who would like to be able to sign up, log in, rent movies, cancel their rentals, and view their profiles. The only important limitations for this project were the number of available movies, which was determined to be 100.

The challenges presented this semester are working with the database as a way to store data, design patterns, and connections between the client and the server. Furthermore, new methodologies and frameworks such as SCRUM and Unified Process are presented, and learning to work with them as well as doing it in an efficient manner is an important obstacle.

As previously stated, the methodology for this semester is Unified Process (UP), which consists of disciplines acting as business modelling, requirements, analysis and design, implementation, testing and deployment. Furthermore, it is divided into different phases, such as inception, elaboration, construction, and transition. This methodology is accompanied by the SCRUM framework, where team roles are set for the project and the work is divided in an organized, productive, and systematic manner.

# Analysis

The introduction above gives information about the background description of the project and mentions analysis as part of the Unifed Process methodology. Elaborating on the background description, the analysis starts by understanding the problem domain and the objects involved. The project has some obvious objects even from the background description such as: movie, costumer, rental. Based on the information provided, the result of analysis, the Domain Model (Larman 2004, chap.9) was created.



## Requirements

Functional requirements

**Critical**

1. As a client I want to rent a movie for a week so that I can have the movie during that time.
2. As a client I want to see the movie’s information such as title, description, genres, actors, director and the movie’s rating and comments so that I can have an idea about the movie.
3. As an administrator I want to remove or add movies so that I can create diversity among the available movies.
4. As an administrator I want the system to request information such as full name, age, password, username, and phone number, so that customers can create accounts.
5. As a client I want to create an account with my full name, age, password, username, and phone number so that I can login into the system.
6. As an administrator I want to login so that I can get access to special features such as adding, removing, and editing movies.
7. As a client I want to see the top-rated movies so that I can rent a high-quality movie.

**High**

1. As a client I want to see the trending movies so that I can rent the popular movies.
2. As a client I want to search for movies based on their genres so that I can rent a movie from that preferred genre.
3. As a client I want to search for a word in the movie title so that I can find it easier.
4. As a client I want to be able to see my profile information such as full name, age, phone number and rented movies so that I can see what movies I have rented and my information.

**Medium**

1. As a client I want to see the reviews and ratings from other users about the movie so that I can determine if I want to rent it.
2. As a client I want to write a review for a movie so that I can express how I felt about it.
3. As a client I want to leave a rating for a movie so that I can express my opinion on its quality.
4. As a client I want to see the average rating of the movies so that I can determine its quality.

**Low**

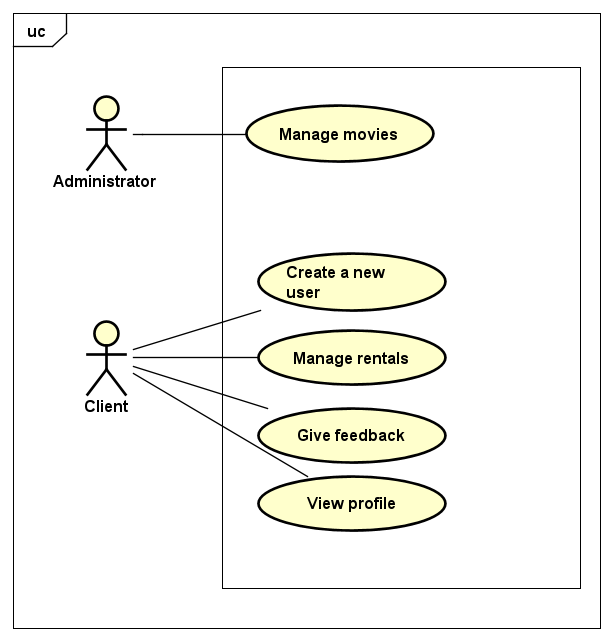
1. As an administrator I want to edit movies information such as descriptions, titles, directors, and lengths so that I can fix any grammatical mistake or error in those fields.
2. As a client I want to cancel a rental so that I do not have that movie in my library.
3. As an administrator I want the system to automatically moderate the comments so that no swear words are allowed.

Non-functional requirements

1. The system will run on Java 8.0.0.
2. The database language will be PostgreSQL.

## Functional Requirements

Functional requirements can be described with Use Cases and Use Case descriptions. Use Case descriptions can be detailed with different types of UML diagrams such as Activity diagrams and System Sequence diagrams.



The use cases presented in the diagram above were created based on the project requirements stated previously.

The Manage movies use case’s purpose is to create the options of adding, removing, or editing a movie that is not rented at the moment.

The Create a new user use case’s purpose is to allow users to create an account, so that they can access the system.

The Manage rentals use case’s purpose is to allow users to rent any movie or cancel any rental from their library.

The Give feedback use case’s purpose is to create the option of reviewing a movie, the user giving a rating and a comment.

The View profile use case’s purpose is to allow the user to view the profile, which presents personal information such as phone number and full name, as well as the library.

**Relation between requirements and use cases**

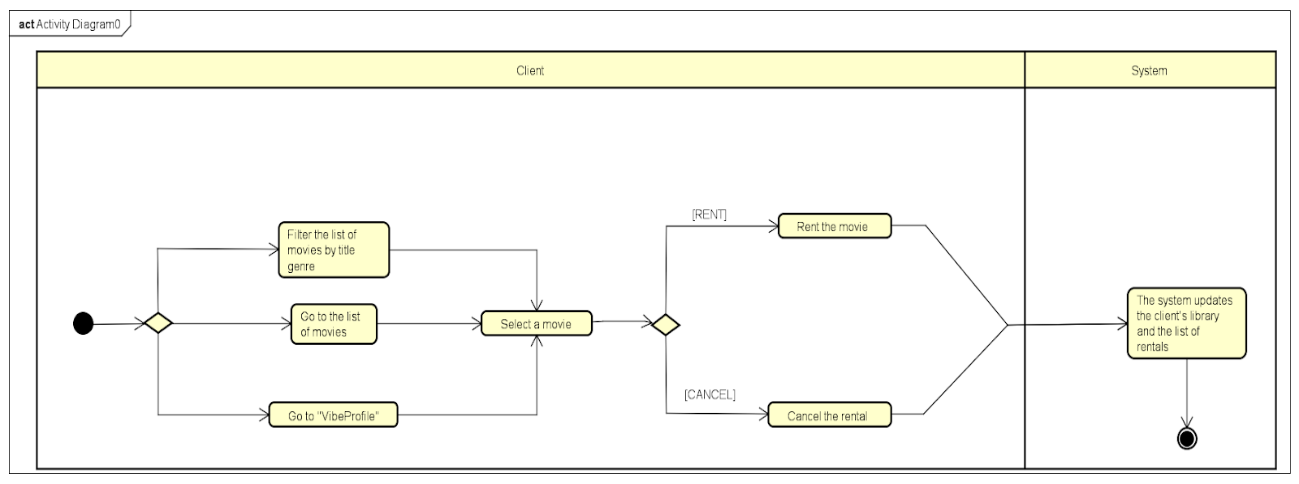
|  |  |
| --- | --- |
| **Use Case** | **Covered requirements** |
| **Manage movies** | 3, 6, 16 |
| **Create new user** | 5 |
| **Manage rentals** | 1, 2, 7, 8, 9, 10, 11, 12, 15, 17 |
| **Give feedback** | 12, 13, 14, 15, 18 |
| **View profile** | 2 |

**Use case description**

|  |  |
| --- | --- |
| **Use case** | Manage rentals |
| **Summary** | Create a rental in the system and display it in the client’s library where the client gets access to the movie they have chosen to rent during the specified period. Cancel a rental from the client’s library and remove it from the system. |
| **Actor** | Client |
| **Precondition** | A list of movies is accessible to the client. |
| **Post condition** | The rental is added or removed from the client’s library and the system. |
| **Base sequence** | RENT, CANCEL  1.      Go to the list of movies.  2.      Select a movie.  If Rent, go to step 3  If Cancel, go to step 5  START RENT  3.      Rent the movie.  4.      The system updates the client’s library and the list of rentals.  END RENT  START CANCEL  5.      Cancel the rental.  6.      The system updates the client’s library and the list of rentals.  END CANCEL |
| **Alternative sequence** | 1.a. Filter the list of movies by title or genre.  1.b. A rental can be canceled from the client’s library, in the client’s profile. |
| **Exception sequence** | 1.a.i. The filter doesn’t return any movies. The use case ends.  1.b.i. The client’s library is empty. The use case ends. |
| **Note** | This Use case covers the requirements: 1, 2, 7, 8, 9, 10, 11, 12, 15, 17 |

The use case above presents the process of renting a movie or removing a rental. Functionality-wise, it is the most important use case, and it is the base of creating a rental system for the project. The main important features are being able to search a movie by title or genre, rent the chosen movie for a week period at a time, and cancel that rental before the period expires and it is automatically removed. For the rest of the use cases descriptions, see the Appendices section.

**Activity diagram**

****

An activity diagram is created based on the corresponding use case description. Therefore, the steps from the user case description should match perfectly with the steps from the activity diagram. However, their visual presentation offers a more approachable way of interpreting the process. The most noticeable features are complementary to the use case description, more specifically: being able to search a movie by title or genre, rent the chosen movie for a week period at a time, and cancel that rental before the period expires and it is automatically removed. For the rest of the activity diagrams, see the Appendices section.

## Non-Functional Requirements

The non-functional requirements for the project target mainly the java and database applications versions. They were thought of using the FURPS+ principle.

# Design

The purpose of the design section is to outline HOW the system is structured; i.e. to transform the artefacts of the analysis into a model that can be implemented. The design section is relevant for the programmer, whereas the analysis is relevant for the stakeholder.

Elements that may be relevant in this section:

·  Architecture: Find architecture patterns here (Leszek Maciaszek 2004, chap.9).

·  Technologies: Describe technologies used, also alternative technologies. Argue for choice of technology according to the project aim.

·  Design Patterns: Describe which design patterns (GoF (Gamma et al. 2002) etc.) you are using and why.

·  Class Diagrams

·  Interaction Diagrams

·  UI design choices

·  Data models, persistence, etc.

You must explain all diagrams in the report. These diagrams including descriptions are the blueprints for the implementation.

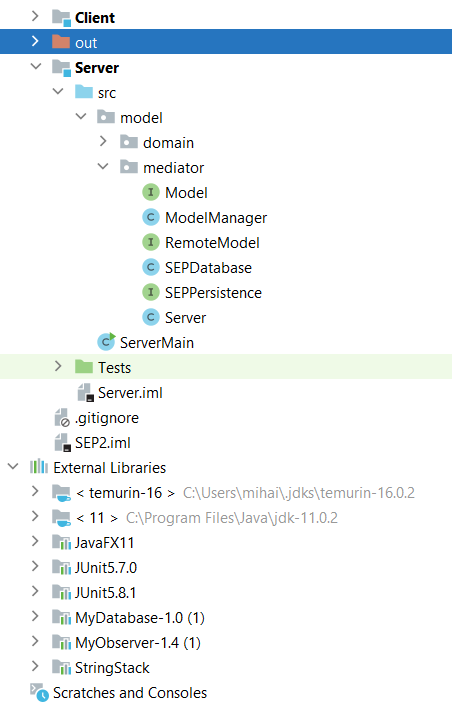
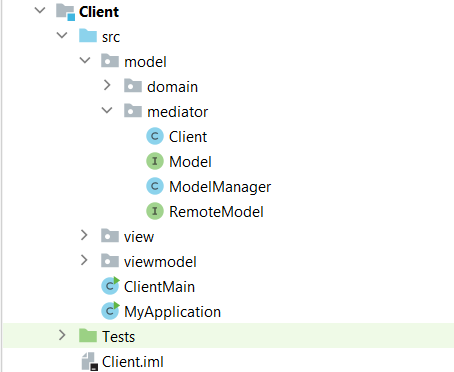
Hint: One way to figure out which objects/classes are needed in the design is to apply the General Responsibility Assignment Software Patterns/principles (GRASP) (Larman 2004, chap.17).

Hint: Consider how to design your system to make it testable.

# Implementation

The domain model from the analysis led to the UML class diagram from the design stage and enabled the possibility to start the implementation phase.

In our project there are two modules, one for the client and one for the server. The server has access to a database.

The libraries used are JavaFX11 for the GUI, Junit5.8.1 and StringStack for the testing and MyDatabase-1.0 in an adapter pattern to connect the database. Other design patterns used are MVVM, proxy and state patterns on the Client side. In the picture above it can be seen the MVVM pattern, as the project has bot view and viewmodel packages. This allows us to integrate Single Responsibility because the view is no longer functional; the functionality of the window being now in the viewmodel, which calls methods in the model. The methods will travel all the way until the server’s ModelManager, where the information is stored, leaving the other classes clueless to the process behind a called method. The project uses RMI to connect the client and the server, meaning the RemoteModel class, as well as any classes that are passed to one another are identical on both sides.

O imagine care conține text

Descriere generată automat

O imagine care conține text

Descriere generată automat

The method above is called every time we run the server. It gets the information from the database and store it in the model, so that we can receive information easier and faster from the server. All the methods that return a search for the client will be done inside the model, and all methods that change in any way the data, will be stored in the model as well as in the database. Below are shown two different methods inside the ModelManager in the server side as example.

O imagine care conține text

Descriere generată automat

Mentioned before is the proxy pattern on the client side. The ModelManager will delegate all work to the Client class, both implementing the Model interface. The Client class will then delegate the work to the RemoteModel class which will connect to the server. Below are the classes ModelManager and Client from the client side.

O imagine care conține text

Descriere generată automat

O imagine care conține text

Descriere generată automat

  The last, but not least, design pattern is the state. The project uses these classes to temporarily store an object and pass it through viewmodels. The classes are to be found in the viewmodel package and are name ViewModelState and ViewModelStateUser, first one storing a movie object and the later one, a user object.

O imagine care conține text

Descriere generată automatO imagine care conține text

Descriere generată automat

They are used in the ViewModelFactory when creating viewmodels. Both the viewmodel setting the object and the one getting the object need to have access to it. We also have to make a small change in the ViewHandler, which is resetting the viewmodel that gets the object before opening the GUI window:

O imagine care conține text

Descriere generată automat

O imagine care conține text

Descriere generată automat

# Test

The testing starts early in the UP-system development method. At first, Alpha and Beta tests were done to progress with the implementation. More bugs and errors were fixed during the Black Box testing as well as setting up the communication protocol (RMI). White Box testing was used to ensure better coverage of all the possible scenarios and outcomes regarding the implementation while Unit Testing used JUnit and focused on low-level pieces of the system, making sure the units perform as expected. In the later stages of development, Integration tests ensured that all the units worked together correctly, and Acceptance tests check that the requirements from the Analysis have been met.

## Test Specifications

Unit Testing

### Log-in

**You can:**

* Create a new user.
* Log in using your username and password. This will take you to the home window for a user or to the admin window for an admin.

**When creating a new user, the following requirements must be met:**

* The username cannot be an empty String and it must not already exist in the system.
* The name cannot be an empty String.
* The phone number cannot be an empty String or null.
* The age must be an integer between 0 and 150.
* The password cannot be an empty String or null.

Name, username and password are case sensitive.

**Behaviour:**

When creating a new user with the correct credentials, the home window will open with the new user logged in.

When creating a new user with incorrect credentials:

* Empty String name: Error message “Invalid name”
* Empty String username: Error message “Invalid username”
* The username already exists in the system: Error message “Username already exists!”
* Empty or null Phone number: Error message “Invalid phone number”
* Age is not an integer between 0 and 150: Error message “Invalid age”
* Empty or null Password: Error message “Invalid password”

**When logging in, the following requirements must be met:**

* The username cannot be an empty String and it must exist in the system.
* The password cannot be an empty String or null and it must match the password of the user in the system.

Username and password are case sensitive.

**Behaviour:**

When logging in with the correct credentials, the home window will open with the user logged in.

When logging in with incorrect credentials:

* Empty String username: Error message “Invalid username”
* The username already exists in the system: Error message “Username already exists!”
* Empty or null Password: Error message “Invalid password”
* Password doesn’t match the password of the user in the system: Error message “password is incorrect”

### Manage Movies

**You can:**

* Add a new movie.
* Edit a movie.
* Remove a movie(there are no error messeges).

**When adding a new movie, the following requirements must be met:**

* The title cannot be an empty String or null.
* The length cannot be an empty String.
* The director cannot be an empty String or null.
* The average rating cannot be an empty String.
* The genre cannot be an empty String or null.
* The release year cannot be an empty String.
* The description cannot be an empty String or null.

**Behaviour:**

When adding a new movie with the correct credentials, the list is updated with the added movie.

When adding a new movie with incorrect credentials:

* Empty or null Title: Error message “Please fill in all fields”
* Empty String Length: Error message “Please fill in all fields”
* Empty or null Director: Error message “Please fill in all fields”
* Empty String Average rating: Error message “Please fill in all fields”
* Empty or null Genre: Error message “Please fill in all fields”
* Empty String Release year: Error message “Please fill in all fields”
* Empty or null Description: Error message “Please fill in all fields”

**When editing a new movie, the following requirements must be met:**

* The title cannot be an empty String or null.
* The length cannot be an empty String.
* The director cannot be an empty String or null.
* The average rating cannot be an empty String.
* The genre cannot be an empty String or null.
* The release year cannot be an empty String.
* The description cannot be an empty String or null.

**Behaviour:**

When editing a movie with the correct credentials, the list is updated with the edited movie.

When editing a movie with incorrect credentials:

* Empty or null Title: Error message “Please fill in all fields”
* Empty String Length: Error message “Please fill in all fields”
* Empty or null Director: Error message “Please fill in all fields”
* Empty String Average rating: Error message “Please fill in all fields”
* Empty or null Genre: Error message “Please fill in all fields”
* Empty String Release year: Error message “Please fill in all fields”
* Empty or null Description: Error message “Please fill in all fields”

### Manage rentals

**You can:**

* Rent a movie.
* Cancel rental.

**When renting a movie, the following requirements must be met:**

* The movie has to exist in the system

**Behaviour:**

When renting a movie with the correct credentials, the client’s library is updated with the new rental.

When renting a movie with incorrect credentials:

* The movie is not rented

**When cancelling a rental, the following requirements must be met:**

* The rental must exist in the clients library.

**Behaviour:**

When cancelling a rental with the correct credentials, the rental is removed from the client’s library.

When editing a movie with incorrect credentials:

* The rental doesn’t exist

### Give feedback

**You can:**

* Rate a movie and write a review for a movie.

**When rating a movie, the following requirements must be met:**

* The rating has to be selected.
* The review cannot be an empty String or null.
* The review cannot have swear words.

**Behaviour:**

When rating and writing a review for a movie with the correct credentials, the message “Changes will appear once you enter the window again ” is displayed.

When rating and writing a review for a movie with incorrect credentials:

* The rating is not selected: Error message “For input string”“
* Empty or null review: Error message “Invalid comment”
* The review has swear words: Error message “Swear words are not allowed!”

**Error messages:**

Creating a new user:

|  |  |
| --- | --- |
| **Error** | **Message** |
| Empty String name | Invalid name |
| Empty String username | Invalid username |
| The username already exists in the system | Username already exists! |
| Empty or null phone number | Invalid phone number |
| Age is not an integer between 0 and 150 | Invalid age |
| Empty or null password | Invalid password |

Logging-in:

|  |  |
| --- | --- |
| **Error** | **Message** |
| Empty String username | Invalid username |
| The username already exists in the system | Username already exists! |
| Empty or null password | Invalid password |
| Password doesn’t match the password of the user | Password is incorrect |

Adding a new movie:

|  |  |
| --- | --- |
| **Error** | **Message** |
| Empty or null title | Please fill in all fields |
| Empty String length | Please fill in all fields |
| Empty or null director | Please fill in all fields |
| Empty String average rating | Please fill in all fields |
| Empty or null genre | Please fill in all fields |
| Empty String release year | Please fill in all fields |
| Empty or null description | Please fill in all fields |

Editing a movie:

|  |  |
| --- | --- |
| **Error** | **Message** |
| Empty or null title | Please fill in all fields |
| Empty String length | Please fill in all fields |
| Empty or null director | Please fill in all fields |
| Empty String average rating | Please fill in all fields |
| Empty or null genre | Please fill in all fields |
| Empty String release year | Please fill in all fields |
| Empty or null description | Please fill in all fields |

Rating and writing a review for a movie:

|  |  |
| --- | --- |
| **Error** | **Message** |
| The rating is not selected | For input string “” |
| Empty or null review | Invalid comment |
| The review has swear words | Swear words are not allowed! |

## TEST CASES

Manage movies

### ADD MOVIE

#### Base sequence

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **TC1** | **TC2** | **TC3** | **TC4** |
|  | Start Add |  |  |  |
| **1** | R | E | W | L |
| **2** | Add Movie |  |  |  |
| **3** | Done |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Add** |  |  |  |  |
| **1-Insert movie data** | Title, Length, Director, Release year, Average rating, Description, Genre | Toy Story, 81, John Lasseter, 1995, 4.7, A cowboy…, Children’s film |  |  |
| **2-Add movie** |  |  | The movie is added to the system. | The movie is added to the system. |
| **3-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 2** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Add** |  |  |  |  |
| **1-Insert movie data** | Title, Length, Director, Release year, Average rating, Description, Genre | <Empty> | No movie is added. End of the sequence | No movie is added. The error message is displayed. End of the sequence |
| **2-Add movie** |  |  |  |  |
| **3-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 3** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Add** |  |  |  |  |
| **1-Insert movie data** | Title, Length, Director, Release year, Average rating, Description, Genre | Toy Story, 81min, John Lasseter, 1995year, 4.7, A cowboy… | Error is displayed, no movie is added. End of the sequence | Error is displayed, no movie is added. End of the sequence |
| **2-Add movie** |  |  |  |  |
| **3-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 4** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Add** |  |  |  |  |
| **1-Insert movie data** | Title, Length, Director, Release year, Average rating, Description, Genre | Toy Story and the attack of the astronaut fighting for the attention: chapter one, 81, John Lasseter, 1995, 4.7, A cowboy…, Children’s film | Value out of bounds. Error is displayed, no movie is added. End of the sequence | Value out of bounds. Error is displayed, no movie is added. End of the sequence |
| **2-Add movie** |  |  |  |  |
| **3-Done** |  |  |  |  |

### EDIT MOVIE

#### Base sequence

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **TC1** | **TC2** | **TC3** | **TC4** |
|  | Start Edit | Start Edit | Start Edit | Start Edit |
| **4** | Go to the list of movies | Go to the list of movies | Go to the list of movies | Go to the list of movies |
| **5** | Select the Movie | Select the Movie | Select the Movie | Select the Movie |
| **6** | R | E | W | L |
| **7** | Done | Done | Done | Done |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Edit** |  |  |  |  |
| **4-Go to the list of movies** |  |  | The list of movies is displayed | The list of movies is displayed |
| **5-Select movie** |  |  | The movie is selected | The movie is selected |
| **6-Edit the movie information** | Title, Length, Director, Release year, Average rating, Description, Genre | Toy Story, 81, J, 1995, 4.7, A cowboy…, Children’s film | The movie is edited in the system. | The movie is edited in the system. |
| **7-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 2** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Edit** |  |  |  |  |
| **4-Go to the list of movies** |  |  | The list of movies is displayed | The list of movies is displayed |
| **5-Select movie** |  |  | The movie is selected | The movie is selected |
| **6-Edit the movie information** | Title, Length, Director, Release year, Average rating, Description, Genre | <Empty> | The movie is not edited. End of sequence. | The movie is not edited. End of sequence. |
| **7-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 3** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Edit** |  |  |  |  |
| **4-Go to the list of movies** |  |  | The list of movies is displayed | The list of movies is displayed |
| **5-Select movie** |  |  | The movie is selected | The movie is selected |
| **6-Edit the movie information** | Title, Length, Director, Release year, Average rating, Description, Genre | Toy Story, 81min, John Lasseter, 1995year, 4.7, A cowboy…, Children’s film | Error is displayed, the movie is not edited. End of sequence. | Error is displayed, the movie is not edited. End of sequence. |
| **7-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 4** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Edit** |  |  |  |  |
| **4-Go to the list of movies** |  |  | The list of movies is displayed | The list of movies is displayed |
| **5-Select movie** |  |  | The movie is selected | The movie is selected |
| **6-Edit the movie information** | Title, Length, Director, Release year, Average rating, Description, Genre | Toy Story and the attack of the astronaut fighting for the attention: chapter one, 81, John Lasseter, 1995, 4.7, A cowboy…, Children’s film | Value out of bounds. Error is displayed, the movie is not edited. End of sequence. | Value out of bounds. Error is displayed, the movie is not edited. End of sequence. |
| **7-Done** |  |  |  |  |

### REMOVE MOVIE

#### Base sequence

|  |  |  |
| --- | --- | --- |
| **Step** | **TC1** |  |
|  | Start Remove |  |
| **4** | Go to the list of movies |  |
| **5** | Select the Movie |  |
| **8** | Remove the movie |  |
| **9** | Done |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Remove** |  |  |  |  |
| **4-Go to the list of movies** |  |  | The list of movies is displayed | The list of movies is displayed |
| **5-Select movie** |  |  | The movie is selected | The movie is selected |
| **8-Remove the movie** |  |  | The movie is removed from the system. | The movie is removed from the system. |
| **7-Done** |  |  |  |  |

Manage rentals

### RENT

#### Base sequence

|  |  |
| --- | --- |
| **Step** | **TC1** |
|  | Start Rent |
| **1** | Go to the list of movies |
| **2** | Select Movie |
| **3** | Rent Movie |
| **4** | Done |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Input field** | **Value** | **Expected result** | **Actual result** |
| **Start Rental** |  |  |  |  |
| **1-Go to the list of movies** |  |  | The list of movies is displayed | The list of movies is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to rent | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to rent |
| **3-Rent Movie** |  |  | The movie is added to the client’s library and the system updates the list of rentals | The movie is added to the client’s library and the system updates the list of rentals |
| **4-Done** |  |  |  |  |

### CANCEL

#### Base sequence

|  |  |
| --- | --- |
| **Step** | **TC1** |
|  | Cancel Rent |
| **1** | Go to the list of rented movies |
| **2** | Select Movie |
| **5** | Rent Movie |
| **6** | Done |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Cancel Rental** |  |  |  |  |
| **1-Go to the list of rented movies** |  |  | The list of rented movies is displayed. | The list of rented movies is displayed. |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to cancel rent. | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to cancel rent. |
| **5-Rent Movie** |  |  | The movie is removed from the client’s library and the system updates the list of rentals. | The movie is removed from the client’s library and the system updates the list of rentals. |
| **6-Done** |  |  |  |  |

#### Alternative sequence

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Step** | **TC1** | **TC2** | **TC3** | **TC4** | **TC5** | **TC6** |
|  | Start Rent | Start Rent | Start Rent | Start Rent | Start Rent | Start Rent |
| **1.a** | R | E | S | R | E | S |
| **2** | Select Movie | Select Movie | Select Movie | Select Movie | Select Movie | Select Movie |
| **3** | Rent Movie | Rent Movie | Rent Movie | Rent Movie | Rent Movie | Rent Movie |
| **4** | Done | Done | Done | Done | Done | Done |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rental** |  |  |  |  |
| **1.a- Filter the list of movies by title or genre** | Movie title | Toy Story | The list of movies is displayed | The list of movies is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to rent | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to rent |
| **3-Rent Movie** |  |  | The movie is added to the client’s library and the system updates the list of rentals | The movie is added to the client’s library and the system updates the list of rentals |
| **4-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 2** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rental** |  |  |  |  |
| **1.a- Filter the list of movies by title or genre** | Movie Title | <Empty> | The list of all movies is displayed | The list of all movies is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to rent | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to rent |
| **3-Rent Movie** |  |  | The movie is added to the client’s library and the system updates the list of rentals | The movie is added to the client’s library and the system updates the list of rentals |
| **4-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 3** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rental** |  |  |  |  |
| **1.a- Filter the list of movies by title or genre** | Movie title | S | The list of movies containing the value is displayed | The list of movies containing the value is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to rent | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to rent |
| **3-Rent Movie** |  |  | The movie is added to the client’s library and the system updates the list of rentals | The movie is added to the client’s library and the system updates the list of rentals |
| **4-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 4** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rental** |  |  |  |  |
| **1.a- Filter the list of movies by title or genre** | Genre | Fantasy | The list of movies in that genre is displayed | The list of movies in that genre is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to rent | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to rent |
| **3-Rent Movie** |  |  | The movie is added to the client’s library and the system updates the list of rentals | The movie is added to the client’s library and the system updates the list of rentals |
| **4-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 5** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rental** |  |  |  |  |
| **1.a- Filter the list of movies by title or genre** | Genre | <Empty> | The list of all movies is displayed | The list of all movies is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to rent | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to rent |
| **3-Rent Movie** |  |  | The movie is added to the client’s library and the system updates the list of rentals | The movie is added to the client’s library and the system updates the list of rentals |
| **4-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 6** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rental** |  |  |  |  |
| **1.a- Filter the list of movies by title or genre** | Genre | A | The system will filter the genres that have the value in the genre name and return the movies within those genres | The system will filter the genres that have the value in the genre name and return the movies within those genres |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to rent | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to rent |
| **3-Rent Movie** |  |  | The movie is added to the client’s library and the system updates the list of rentals | The movie is added to the client’s library and the system updates the list of rentals |
| **4-Done** |  |  |  |  |

|  |  |
| --- | --- |
| **Step** | **TC1** |
|  | Start Cancel |
| **1.b** | Go to the client’s library |
| **2** | Select Movie |
| **5** | Cancel rental |
| **6** | Done |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Cancel** |  |  |  |  |
| **1.b. A rental can be canceled from the client’s library, in the client’s profile.** |  |  | The client’s library is displayed. | The client’s library is displayed. |
| **2-Select movie** |  |  |  |  |
| **5-Cancel the rental** |  |  | The system removes the rental from the client’s library. | The system removes the rental from the client’s library. |
| **6-Done** |  |  |  |  |

#### Exception sequence

|  |  |  |
| --- | --- | --- |
| **Step** | **TC1** | **TC2** |
|  | Start Rent | Start Rent |
| **1.a.i** | R | R |
| **2** |  |  |
| **3** |  |  |
| **4** |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rental** |  |  |  |  |
| **1.a- Filter the list of movies by title or genre** | Movie title | Toy Story 21 | The movie doesn’t exist. End of the sequence. | The movie doesn’t exist. End of the sequence. |
| **2-Select movie** |  |  |  |  |
| **3-Rent Movie** |  |  |  |  |
| **4-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 2** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rental** |  |  |  |  |
| **1.a- Filter the list of movies by title or genre** | Genre | History | The movie with that Genre doesn’t exist. End of the sequence. | The movie with that Genre doesn’t exist. End of the sequence. |
| **2-Select movie** |  |  |  |  |
| **3-Rent Movie** |  |  |  |  |
| **4-Done** |  |  |  |  |

|  |  |
| --- | --- |
| **Step** | **TC1** |
|  | Start Cancel |
| **1.b.i** |  |
| **2** |  |
| **5** |  |
| **6** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Cancel** |  |  |  |  |
| **1.b. A rental can be canceled from the client’s library, in the client’s profile.** |  |  | The client’s library is empty. End of sequence. | The client’s library is empty. End of sequence. |
| **2-Select movie** |  |  |  |  |
| **5-Cancel rental** |  |  |  |  |
| **6-Done** |  |  |  |  |

VIEW PROFILE

### Display

|  |  |
| --- | --- |
| **Step** | **TC1** |
|  | Start Display |
| **1** | Go to the profile |
| **2** | Done |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Display** |  |  |  |  |
| **1-Go to the profile** |  |  | The system displays the client’s information such as username, full name, phone number, and age. A list of the client’s rentals is also displayed. | The system displays the client’s information such as username, full name, phone number, and age. A list of the client’s rentals is also displayed. |
| **2-Done** |  |  |  |  |

Give feedback

### RATE

|  |  |
| --- | --- |
| **Step** | **TC1** |
|  | Start Rate |
| **1** | Go to the list of movies |
| **2** | Select a movie |
| **3** | Give a rating |
| **4** | Done |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rate** |  |  |  |  |
| **1-Go to the list of movies** |  |  | The list of movies is displayed | The list of movies is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to give a rating | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to give a rating |
| **3-Give a rating** |  |  | The system updates the average rating for the movie | The system updates the average rating for the movie |
| **4-Done** |  |  |  |  |

### REVIEW

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **TC1** | **TC2** | **TC3** | **TC3** |
|  | Start Review | Start Review | Start Review | Start Review |
| **1** | Go to the list of movies | Go to the list of movies | Go to the list of movies | Go to the list of movies |
| **2** | Select a movie | Select a movie | Select a movie | Select a movie |
| **5** | R | E | W | L |
| **6** | Submit | Submit | Submit | Submit |
| **7** | Done | Done | Done | Done |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Review** |  |  |  |  |
| **1-Go to the list of movies** |  |  | The list of movies is displayed | The list of movies is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review |
| **5-Write a review** | Review text | It was good :) |  |  |
| **6-Submit** |  |  | The system adds the review to the movie | The system adds the review to the movie |
| **7-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 2** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Review** |  |  |  |  |
| **1-Go to the list of movies** |  |  | The list of movies is displayed | The list of movies is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review |
| **5-Write a review** | Review text | <Empty> |  |  |
| **6-Submit** |  |  | The review cannot be submitted. End of the sequence. | The review cannot be submitted. End of the sequence. |
| **7-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 3** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Review** |  |  |  |  |
| **1-Go to the list of movies** |  |  | The list of movies is displayed | The list of movies is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review |
| **5-Write a review** | Review text | The movie was fucking awesome! |  |  |
| **6-Submit** |  |  | The review cannot be submitted. End of the sequence. | The review cannot be submitted. End of the sequence. |
| **7-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 4** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Review** |  |  |  |  |
| **1-Go to the list of movies** |  |  | The list of movies is displayed | The list of movies is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review |
| **5-Write a review** | Review text | I really really really really... (1000+) |  |  |
| **6-Submit** |  |  | Error is displayed, the review cannot be submitted. End of the sequence. | Error is displayed, the review cannot be submitted. End of the sequence. |
| **7-Done** |  |  |  |  |

#### Alternative sequence

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Step** | **TC1** | **TC2** | **TC3** | **TC4** | **TC5** | **TC6** |
|  | Start Rate | Start Rate | Start Rate | Start Rate | Start Rate | Start Rate |
| **1.a** | R | E | S | R | E | S |
| **2** | Select Movie |  | Select Movie | Select Movie |  | Select Movie |
| **3** | Give a rating |  | Give a rating | Give a rating |  | Give a rating |
| **4** | Done |  | Done | Done |  | Done |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rate** |  |  |  |  |
| **1.a**- **Filter the list of movies by title or genre** | Movie title | Toy Story | The list of movies is displayed | The list of movies is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to give a rating | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to give a rating |
| **3-Give a rating** |  |  | The system updates the average rating for the movie | The system updates the average rating for the movie |
| **4-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 2** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rate** |  |  |  |  |
| **1.a**- **Filter the list of movies by title or genre** | Movie Title | <Empty> | The list of all the movies is displayed | The list of all the movies is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to give a rating | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to give a rating |
| **3-Give a rating** |  |  | The system updates the average rating for the movie | The system updates the average rating for the movie |
| **4-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 3** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rate** |  |  |  |  |
| **1.a**- **Filter the list of movies by title or genre** | Movie title | S | The list of movies containing the value is displayed | The list of movies containing the value is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to give a rating | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to give a rating |
| **3-Give a rating** |  |  | The system updates the average rating for the movie | The system updates the average rating for the movie |
| **4-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 4** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rate** |  |  |  |  |
| **1.a**- **Filter the list of movies by title or genre** | Genre | Fantasy | The list of movies in that genre is displayed | The list of movies in that genre is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to give a rating | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to give a rating |
| **3-Give a rating** |  |  | The system updates the average rating for the movie | The system updates the average rating for the movie |
| **4-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 5** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rate** |  |  |  |  |
| **1.a**- **Filter the list of movies by title or genre** | Genre | <Empty> | The list of all the movies is displayed | The list of all the movies is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to give a rating | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to give a rating |
| **3-Give a rating** |  |  | The system updates the average rating for the movie | The system updates the average rating for the movie |
| **4-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 6** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rate** |  |  |  |  |
| **1.a**- **Filter the list of movies by title or genre** | Genre | A | The system will filter the genres that have the value in the genre name and return the movies within those genres | The system will filter the genres that have the value in the genre name and return the movies within those genres |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to give a rating | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to give a rating |
| **3-Give a rating** |  |  | The system updates the average rating for the movie | The system updates the average rating for the movie |
| **4-Done** |  |  |  |  |

|  |  |
| --- | --- |
| **Step** | **TC1** |
|  | Start Rate |
| **1.b** | Go to the client’s library |
| **2** | Select Movie |
| **3** | Give a rating |
| **4** | Done |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rate** |  |  |  |  |
| **1.b. Go to the client’s library in the client’s profile.** |  |  | The client’s library is displayed. | The client’s library is displayed. |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to give a rating | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to give a rating |
| **3-Give a rating** |  |  | The system updates the average rating for the movie | The system updates the average rating for the movie |
| **4-Done** |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Step** | **TC1** | **TC2** | **TC3** | **TC4** | **TC5** | **TC6** |
|  | Start Review | Start Review | Start Review | Start Review | Start Review | Start Review |
| **1.a** | R | E | S | R | E | S |
| **2** | Select Movie | Select Movie | Select Movie | Select Movie | Select Movie | Select Movie |
| **5** | R\* | R\* | R\* | R\* | R\* | R\* |
| **6** | Submit | Submit | Submit | Submit | Submit | Submit |
| **7** | Done | Done | Done | Done | Done | Done |

\*Step 3 follows the same test cases as the base sequence

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Review** |  |  |  |  |
| **1.a**- **Filter the list of movies by title or genre** | Movie title | Toy Story | The list of movies is displayed | The list of movies is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review |
| **5-Write a review** | Review text | It was good :) |  |  |
| **6-Submit** |  |  | The system adds the review to the movie | The system adds the review to the movie |
| **7-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 2** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Review** |  |  |  |  |
| **1.a**- **Filter the list of movies by title or genre** | Movie Title | <Empty> | The list of all the movies is displayed | The list of all the movies is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review |
| **5-Write a review** | Review text | It was good :) |  |  |
| **6-Submit** |  |  | The system adds the review to the movie | The system adds the review to the movie |
| **7-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 3** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Review** |  |  |  |  |
| **1.a**- **Filter the list of movies by title or genre** | Movie title | S | The list of movies containing the value is displayed | The list of movies containing the value is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review |
| **5-Write a review** | Review text | It was good :) |  |  |
| **6-Submit** |  |  | The system adds the review to the movie | The system adds the review to the movie |
| **7-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 4** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Review** |  |  |  |  |
| **1.a**- **Filter the list of movies by title or genre** | Genre | Fantasy | The list of movies in that genre is displayed | The list of movies in that genre is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review |
| **5-Write a review** | Review text | It was good :) |  |  |
| **6-Submit** |  |  | The system adds the review to the movie | The system adds the review to the movie |
| **7-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 5** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Review** |  |  |  |  |
| **1.a**- **Filter the list of movies by title or genre** | Genre | <Empty> | The list of all the movies is displayed | The list of all the movies is displayed |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review |
| **5-Write a review** | Review text |  |  |  |
| **6-Submit** |  |  | The system adds the review to the movie | The system adds the review to the movie |
| **7-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 6** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Review** |  |  |  |  |
| **1.a**- **Filter the list of movies by title or genre** | Genre | A | The system will filter the genres that have the value in the genre name and return the movies within those genres | The system will filter the genres that have the value in the genre name and return the movies within those genres |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to write a review |
| **5-Write a review** | Review text | It was good :) |  |  |
| **6-Submit** |  |  | The system adds the review to the movie | The system adds the review to the movie |
| **7-Done** |  |  |  |  |

|  |  |
| --- | --- |
| **Step** | **TC1** |
|  | Start Review |
| **1.b** | Go to the client’s library |
| **2** | Select Movie |
| **5** | R\* |
| **6** | Submit |
| **7** | Done |

\*Step 3 follows the same test cases as the base sequence

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Review** |  |  |  |  |
| **1.b. Go to the client’s library in the client’s profile.** |  |  | The client’s library is displayed. | The client’s library is displayed. |
| **2-Select movie** |  |  | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to give a rating | Information regarding the movie (title, director, length, description, average rating, release year, genre, and reviews) is displayed along with the possibility to give a rating |
| **5-Write a review** | Review text | It was good :) |  |  |
| **6-Submit** |  |  | The system adds the review to the movie | The system adds the review to the movie |
| **7-Done** |  |  |  |  |

#### Exception sequence

|  |  |  |
| --- | --- | --- |
| **Step** | **TC1** | **TC2** |
|  | Start Rate | Start Rate |
| **1.a.i** | R | R |
| **2** |  |  |
| **3** |  |  |
| **4** |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rate** |  |  |  |  |
| **1. - Filter the list of movies by title or genre** | Movie title | Toy Story 21 | The movie doesn’t exist. End of the sequence. | The movie doesn’t exist. End of the sequence. |
| **2-Select movie** |  |  |  |  |
| **3-Give a rating** |  |  |  |  |
| **4-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 2** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rate** |  |  |  |  |
| **1. - Filter the list of movies by title or genre** | Genre | History | The movie doesn’t exist. End of the sequence. | The movie doesn’t exist. End of the sequence. |
| **2-Select movie** |  |  |  |  |
| **3-Give a rating** |  |  |  |  |
| **4-Done** |  |  |  |  |

|  |  |  |
| --- | --- | --- |
| **Step** | **TC1** | **TC2** |
|  | Start Review | Start Review |
| **1.a.i** | R | R |
| **2** |  |  |
| **5** |  |  |
| **6** |  |  |
| **7** |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Review** |  |  |  |  |
| **1. - Filter the list of movies by title or genre** | Movie title | Toy Story 21 | The movie doesn’t exist. End of the sequence. | The movie doesn’t exist. End of the sequence. |
| **2-Select movie** |  |  |  |  |
| **5-Write a review** | Review text |  |  |  |
| **6-Submit** |  |  |  |  |
| **7-Done** |  |  |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 2** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Review** |  |  |  |  |
| **1.a- Filter the list of movies by title or genre** | Genre | History | The Genre doesn’t exist. End of the sequence. | The Genre doesn’t exist. End of the sequence. |
| **2-Select movie** |  |  |  |  |
| **5-Write a review** | Review text |  |  |  |
| **6-Submit** |  |  |  |  |
| **7-Done** |  |  |  |  |

|  |  |
| --- | --- |
| **Step** | **TC1** |
|  | Start Rate |
| **1.b.i** |  |
| **2** |  |
| **3** |  |
| **4** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Rate** |  |  |  |  |
| **1. - Go to the client’s library in the client’s profile.** |  |  | The client’s library is empty. End of sequence. | The client’s library is empty. End of sequence. |
| **2-Select movie** |  |  |  |  |
| **3-Give a rating** |  |  |  |  |
| **4-Done** |  |  |  |  |

|  |  |
| --- | --- |
| **Step** | **TC1** |
|  | Start Review |
| **1.b.i** |  |
| **2** |  |
| **3** |  |
| **4** |  |
| **5** |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | **Test Case 1** |  |  |
| **Step** | Input field | Value | Expected result | Actual result |
| **Start Review** |  |  |  |  |
| **1. - Go to the client’s library in the client’s profile.** |  |  | The client’s library is empty. End of sequence. | The client’s library is empty. End of sequence. |
| **2-Select movie** |  |  |  |  |
| **5-Write a review** |  |  |  |  |
| **6-Submit** |  |  |  |  |
| **7-Done** |  |  |  |  |

# Results and Discussion

The purpose of the results and discussion section is to present the outcome and achieved results of the project.

# Conclusions

The world is evolving in ways that were unthinkable in the past, technologies continue to develop and as such, old ideas are perhaps forgotten, even if there is a demand for them. Movie rentals could enter this category, even though some services are now offering rental possibilities, their range is far from perfect.

This topic was chosen from the idea, of the lack of availability of this service. Therefore, a lot of ideas had to be discussed, mostly on what would customers like to do with this resource. Multiple requirements were identified from these simple questions, from this, the idea of VibeFlix was conceived, a service that could tackle the problems presented and would improve the idea of streaming systems as a whole.

Initially, most of these requirements and ideas were identified during the project description. As time passed, and stages such as analysis, design and implementation these ideas were further improved.

One of the most important facts of this project was the lessons gathered from last semester. Organization and time distribution were one of the key factors that led to the previous project not being fully completed and missing features. Furthermore, the success of this project can be attributed to teamwork as well as the use of SCRUM and UP, which presented new ways to approach Problem Based Learning.

The usage of SCRUM allowed for the distribution of tasks for each member of the group in a fair and reasonable way. Moreover, it allowed the team to focus on what actually matters for the project as well as allow for enough time for members to rest. However, SCRUM was not alone in this project, Unified Process was used along with it, which resulted in SCRUM objectives being based on the different phases of UP.

Finally, the project can be considered as achieved. This project managed to accomplish total functionality based on the identified requirements. Furthermore, aesthetical details were added due to the project finishing earlier than expected, which also allowed for documentation to happen in a calm and productive manner. It is possible to make rentals, cancel them as well as create an account, log in and manage the system.

# Project future

The project from a technical viewpoint is a success. However, everything can be improved.

Firstly, an interesting feature nonetheless would be the system reproducing the movies listed. Even though this implies copyright and licensing issues, it could be useful to reproduce these movies after the rental has been processed.

Secondly, filter restrictions on the movies. The system did not account for the different ratings of movies, it is important to showcase movies appropriate for each age group by using the “Motion Picture Association film rating system” and limit the access to some of them.

Finally, this system could be adapted to a multitude of purposes, it could be a game, bicycle, items or house rentals. Therefore, it would be important to keep this system in mind for the future, as a base on what should be done.

# Sources of information

**Note: Use the standard reference method: Harvard Anglia. A very good reference tool is Mendeley** (Mendeley.com 2016), **ask VIA Library if you need help.**

Banger, D., 2014. A Basic Non-Functional Requirements Checklist « Thoughts from the Systems front line.... Available at: https://dalbanger.wordpress.com/2014/01/08/a-basic-non-functional-requirements-checklist/ [Accessed January 31, 2017].

Business Analyst Learnings, 2013. MoSCoW : Requirements Prioritization Technique — Business Analyst Learnings. , pp.1–5. Available at: https://businessanalystlearnings.com/ba-techniques/2013/3/5/moscow-technique-requirements-prioritization [Accessed January 31, 2017].

Dawson, C.W., 2009. *Projects in Computing and Information Systems*, Available at: http://www.sentimentaltoday.net/National\_Academy\_Press/0321263553.Addison.Wesley.Publishing.Company.Projects.in.Computing.and.Information.Systems.A.Students.Guide.Jun.2005.pdf.

Gamma, E. et al., 2002. *Design Patterns – Elements of Reusable Object-Oriented Software*, Available at: http://books.google.com/books?id=JPOaP7cyk6wC&pg=PA78&dq=intitle:Design+Patterns+Elements+of+Reusable+Object+Oriented+Software&hl=&cd=3&source=gbs\_api%5Cnpapers2://publication/uuid/944613AA-7124-44A4-B86F-C7B2123344F3.

IEEE Computer Society, 2008. *IEEE Std 829-2008, IEEE Standard for Software and System Test Documentation*,

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

Mendeley.com, 2016. Homepage | Mendeley. Available at: https://www.mendeley.com/ [Accessed February 2, 2017].

YourCoach, S.M.A.R.T. goal setting | SMART | Coaching tools | YourCoach Gent. Available at: http://www.yourcoach.be/en/coaching-tools/smart-goal-setting.php [Accessed August 19, 2017].

# Appendices

The purpose of your appendices is to provide extra information to the expert reader. List the appendices in order of mention.

Examples of appendices

·        Project Description

·        User Guide

·        Source code – source documentation

·        Diagrams

·        Data sheets

·        Etc.

**Appendix A Project Description**

Insert the original Project Description here