**User Requirements Specification**

Individual Project – Stream Sage

**Title Page**



* Date: 05th April 2024
* Location: Fontys R10
* Member: Georgi Tinchev
* Tutor: Bhardwaj Sachin

**Table of Contents**

* [Title Page](#TitlePage)
* [Functional Requirements](#Functional)
* [Non-Functional Requirements](#NonFunctional)
* [Use Cases](#UseCases)
* [Use Case Diagram](#UseCaseDiagram)
* [Desktop App GUI](#DesktopAppGUI)
* [Website GUI](#WebAppGUI)

Functional Requirements

System Capabilities Desktop App:

* 1. Movie Management
* ID: FRD-1[[1]](#footnote-1)
* Name: Manage movie database.
* Description: Allow administrators to add, edit, and delete from the database.
* Constraints: Administrators can create, read, update, and delete movie records. Each movie should have a unique identifier (MovieID), title, release date, description, poster image URL, trailer URL, runtime in minutes, and average rating.
  1. User Management
* ID: FRD-2
* Name: Perform user authentication and authorization.
* Constraints: Users should be able to register, login, and manage their profiles. User authentication should be secure, and only authenticated users can access certain functionalities like rating movies, submitting interpretations, and writing reviews.  
  1. Genre, Actor, Director Management
* ID: FRD-3
* Name: Manage genre, actor, and director information.
* Constraints: Administrators should be able to manage information about movie genres, actors, and directors. This includes adding, updating, and deleting records for genres, actors, and directors.
  1. User Authentication
* ID: FRD-4
* Name: Implement user authentication.
* Description: Users should be able to log in, and log out of the system securely.
  1. Interpretation Management
* ID: FRD-5
* Name: Allow users to submit their interpretations of movies.
* Constraints: Users should be able to write interpretations of movies they have watched. Each interpretation should be associated with a specific movie and user. Interpretations should include text content and the date of submission.
  1. Review Management
* ID: FRD-6
* Name: Allow users to write reviews for movies.
* Constraints: Users should be able to write reviews for movies they have watched. Each review should include a rating, text content, and the date of submission. Reviews should be associated with both the movie being reviewed and the user writing the review.
  1. Recommendation system
* ID: FRD-7
* Name: Manage the recommendation system
* Description: Manage the recommendation system which is used within the website version of the application.
  1. Statistics and analysis
* ID: FRD-8
* Name: Statistics and analysis on usage
* Description: Display some statistics or analysis based on the CRUD operations and usage of the application by the end-users.

System Capabilities Website:

1. Movie Browsing and Searching

* **ID:** FRW-1[[2]](#footnote-2)
* **Name:** Enable users to browse and search for movies.
* **Description:** Users should be able to view a list of available movies, filter them by genre, search for specific titles, and view detailed information about each movie.

1. User Registration and Authentication

* **ID:** FRW-2
* **Name:** Facilitate user registration and authentication.
* **Description:** Users should be able to register for an account, log in securely, and log out when they're done. Authentication should ensure the security of user accounts and data.

1. Profile Management

* **ID:** FRW-3
* **Name:** Allow users to manage their profiles.
* **Description:** Users should be able to update their profile information, change passwords, and manage account settings such as email preferences and privacy settings.

1. Interactivity and Engagement

* **ID:** FRW-4
* **Name:** Provide interactive features for user engagement.
* **Description:** Users should be able to rate movies, submit interpretations, write reviews, and interact with other users through comments or discussions.

1. Recommendations and Personalization

* **ID:** FRW-5
* **Name:** Implement a recommendation system for personalized content.
* **Description:** The system should analyze user preferences, viewing history, and interactions to provide personalized movie recommendations tailored to each user's interests.

1. Accessibility and Usability

* **ID:** FRW-6
* **Name:** Ensure accessibility and usability across devices.
* **Description:** The web app should be user-friendly, responsive, and accessible across various devices and screen sizes to provide a seamless experience for all users.

1. Social Integration

* **ID:** FRW-7
* **Name:** Integrate social media features for sharing and engagement.
* **Description:** Users should be able to share movie recommendations, reviews, and interpretations on social media platforms, fostering engagement and expanding the user base.

Non-Functional Requirements

1. Performance

* ID: NFR-1[[3]](#footnote-3)
* Name: Ensure the system performs efficiently even with many movies and users.
* Constraints: Response time should be within acceptable limits for all user interactions, even as the size of the database grows.

1. Security

* ID: NFR-2
* Name: Implement security measures to protect user data and prevent unauthorized access.
* Constraints: User authentication should be secure, passwords should be stored securely using hashing algorithms, and sensitive data should be protected from unauthorized access.

1. Scalability

* ID: NFR-3
* Name: Design the system to handle future growth in terms of users, movies, and data.
* Constraints: The system should be able to scale to accommodate an increasing number of users, movies, and interactions without sacrificing performance or reliability.

**Use Cases**

CRUD Operations on Movies (Desktop App)

* UC ID: UC-1[[4]](#footnote-4)
* Description: Administrator performs CRUD operations on movies.
* Trigger: Administrator initiates the operation.
* Pre-condition: Administrator is logged in.
* Post-condition: Changes to the movie database are saved.

MSS[[5]](#footnote-5):

1. Administrator selects a movie to perform CRUD operations.
2. Administrator performs the desired operation (create, read, update, delete) and submits changes.
3. System validates the changes and updates the movie database.

Extensions:

* 2a: Administrator cancels the operation.
  + MSS Steps:
    1. System cancels the operation.
    2. End of use case.

CRUD Operations on Users (Desktop App)

* UC ID: UC-2
* Description: Administrator performs CRUD operations on users.
* Trigger: Administrator initiates the operation.
* Pre-condition: Administrator is logged in.
* Post-condition: Changes to the user database are saved.

MSS:

1. Administrator selects a user to perform CRUD operations.
2. Administrator performs the desired operation (create, read, update, delete) and submits changes.
3. System validates the changes and updates the user database.

Extensions:

* 2a: Administrator cancels the operation.
  + MSS Steps:
    1. System cancels the operation.
    2. End of use case.

CRUD Operations on Reviews (Web App)

* UC ID: UC-3
* Description: User performs CRUD operations on reviews.
* Trigger: User initiates the operation.
* Pre-condition: User is logged in.
* Post-condition: Changes to the review database are saved.

MSS:

1. User selects a review to perform CRUD operations.
2. User performs the desired operation (create, read, delete) and submits changes.
3. System validates the changes and updates the review database.

Extensions:

* 2a: User cancels the operation.
  + MSS Steps:
    1. System cancels the operation.
    2. End of use case.

CRUD Operations on Interpretations (Web App)

* UC ID: UC-4
* Description: User performs CRUD operations on interpretations.
* Trigger: User initiates the operation.
* Pre-condition: User is logged in and the interpretation is his own.
* Post-condition: Changes to the interpretation database are saved.

MSS:

1. User selects an interpretation to perform CRUD operations.
2. User performs the desired operation (create, read, delete) and submits changes.
3. System validates the changes and updates the interpretation database.

Extensions:

* 2a: User cancels the operation.
  + MSS Steps:
    1. System cancels the operation.
    2. End of use case.

Registration & Login (Authentication) (Web App)

* UC ID: UC-5
* Description: New user registers for an account and logs in.
* Trigger: User initiates the registration and login process.
* Pre-condition: User is not logged in.
* Post-condition: User account is created, and user is logged in.

MSS:

1. User enters required information and submits the registration form.
2. System validates the information and creates a new user account.
3. User enters login credentials and submits them.
4. System validates the credentials and logs the user in.

Extensions:

* 2a: User cancels the registration process.
  + MSS Steps:
    1. System cancels the operation.
    2. End of use case.
* 4a: User enters incorrect credentials.
  + MSS Steps:
    1. System shows an error message.
    2. End of use case.

Making User Admin (Desktop App)

* UC ID: UC-6
* Description: Administrator makes a user an admin.
* Trigger: Administrator initiates the process.
* Pre-condition: Administrator is logged in.
* Post-condition: User is made an admin.

MSS:

1. Administrator selects a user to make admin.
2. Administrator submits the change.
3. System validates the change and updates the user's role to admin.

Extensions:

* 2a: Administrator cancels the operation.
  + MSS Steps:
    1. System cancels the operation.
    2. End of use case.

Genre, Actor, Director CRUD Operations (Desktop App)

* UC ID: UC-7
* Description: Administrator performs CRUD operations on genre, actor, and director information.
* Trigger: Administrator initiates the operation.
* Pre-condition: Administrator is logged in.
* Post-condition: Changes to the genre, actor, and director information are saved.

MSS:

1. Administrator selects a genre, actor, or director to perform CRUD operations.
2. Administrator performs the desired operation (create, read, update, delete) and submits changes.
3. System validates the changes and updates the information.

Extensions:

* 2a: Administrator cancels the operation.
  + MSS Steps:
    1. System cancels the operation.
    2. End of use case.

Viewing Recommendations (Web App)

* UC ID: UC-8
* Description: User views movie recommendations.
* Trigger: User accesses the recommendation feature.
* Pre-condition: User is logged in.
* Post-condition: Movie recommendations are displayed to the user.

MSS:

1. User accesses the recommendation feature.
2. System analyzes user preferences and viewing history.
3. System displays personalized movie recommendations.

Extensions:

* 2a: User cancels the recommendation process.
  + MSS Steps:
    1. System cancels the operation.
    2. End of use case.

Searching for Movies (Web App)

* UC ID: UC-9
* Description: User searches for movies.
* Trigger: User initiates the search process.
* Pre-condition: User is logged in.
* Post-condition: Search results are displayed to the user.

MSS:

1. User enters search criteria.
2. System retrieves and displays movies that match the criteria.

Extensions:

* 2a: User cancels the search process.
  + MSS Steps:
    1. System cancels the operation.
    2. End of use case.

Filtering Movies (Web App)

* UC ID: UC-10
* Description: User filters movies based on different criteria.
* Trigger: User initiates the filtering process.
* Pre-condition: User is logged in.
* Post-condition: Filtered movies are displayed to the user.

MSS:

1. User selects filtering criteria.
2. System retrieves and displays movies that match the criteria.

Extensions:

* 2a: User cancels the filtering process.
  + MSS Steps:
    1. System cancels the operation.
    2. End of use case.

Password Reset (Web App)

* **UC ID:** UC-11
* **Description:** User resets their password.
* **Trigger:** User initiates the password reset process.
* **Pre-condition:** User has a registered account.
* **Post-condition:** User's password is reset.

**MSS:**

1. User enters their email address and requests a password reset.
2. System sends a password reset link to the user's email.
3. User clicks on the link and enters a new password.
4. System validates and updates the new password.

**Extensions:**

* **2a:** User enters an email address that is not associated with any account.
  + **MSS Steps:**
    1. System shows an error message.
    2. End of use case.

Contact Administrator (Web App)

* **UC ID:** UC-12
* **Description:** User contacts the administrator.
* **Trigger:** User initiates the contact process.
* **Pre-condition:** User has a registered account.
* **Post-condition:** User's message is sent to the administrator.

**MSS:**

1. User enters their message and submits it.
2. System sends the message to the administrator.

**Extensions:**

* **2a:** User cancels the contact process.
  + **MSS Steps:**
    1. System cancels the operation.
    2. End of use case.

View User Profile (Web App**)**

* **UC ID:** UC-13
* **Description:** User views their profile.
* **Trigger:** User initiates the view.
* **Pre-condition:** User is logged in.
* **Post-condition:** User's profile is displayed.

**MSS:**

1. User selects to view their profile.
2. System retrieves and displays the user's profile.

**Extensions:**

* **2a:** User cancels the view.
  + **MSS Steps:**
    1. System cancels the operation.
    2. End of use case.

Update User Profile (Web App)

* **UC ID:** UC-14
* **Description:** User updates their profile.
* **Trigger:** User initiates the update process.
* **Pre-condition:** User is logged in.
* **Post-condition:** User's profile is updated.

**MSS:**

1. User selects to update their profile.
2. User makes changes and submits them.
3. System validates the changes and updates the user's profile.

**Extensions:**

* **2a:** User cancels the update process.
  + **MSS Steps:**
    1. System cancels the operation.
    2. End of use case.

Logout (Web App & Desktop App)

* **UC ID:** UC-15
* **Description:** User logs out of the system.
* **Trigger:** User initiates the logout process.
* **Pre-condition:** User is logged in.
* **Post-condition:** User is logged out.

**MSS:**

1. User selects to log out.
2. System logs the user out.

**Extensions:**

* **2a:** User cancels the logout process.
  + **MSS Steps:**
    1. System cancels the operation.
    2. End of use case.

View Movie Trailer (Web App)

* **UC ID:** UC-16
* **Description:** User views a movie trailer.
* **Trigger:** User selects to view the trailer.
* **Pre-condition:** User is logged in.
* **Post-condition:** Movie trailer is displayed.

**MSS:**

1. User selects a movie whose trailer they want to view.
2. System retrieves and displays the movie trailer.

**Extensions:**

* **2a:** User cancels the view.
  + **MSS Steps:**
    1. System cancels the operation.
    2. End of use case.

Watch Movie (Web App)

* **UC ID:** UC-17
* **Description:** User watches a movie.
* **Trigger:** User selects to watch the movie.
* **Pre-condition:** User is logged in and the movie is available for streaming.
* **Post-condition:** Movie is streamed to the user.

**MSS:**

1. User selects a movie they want to watch.
2. System retrieves and streams the movie.

**Extensions:**

* **2a:** User cancels the streaming.
  + **MSS Steps:**
    1. System cancels the operation.
    2. End of use case.

View User Activity (Desktop App)

* **UC ID:** UC-18
* **Description:** Administrator views user activity.
* **Trigger:** Administrator initiates the view.
* **Pre-condition:** Administrator is logged in.
* **Post-condition:** User activity is displayed.

**MSS:**

1. Administrator selects a user whose activity they want to view.
2. System retrieves and displays the user's activity.

**Extensions:**

* **2a:** Administrator cancels the view.
  + **MSS Steps:**
    1. System cancels the operation.
    2. End of use case.

View User Activity (Web App)

* UC ID: UC-18
* Description: User views their activity.
* Trigger: User selects to view their activity.
* Pre-condition: User is logged in.
* Post-condition: User's activity is displayed.
* MSS:
  + User selects to view their activity.
  + System retrieves and displays the user's activity.
* Extensions:
  + 2a: User decides to stop viewing their activity.
    1. MSS Steps:
       1. System stops displaying the activity.
       2. End of use case.

View Leaderboard (Web App)

* UC ID: UC-19
* Description: User views the leaderboard.
* Trigger: User selects to view the leaderboard.
* Pre-condition: User is logged in.
* Post-condition: Leaderboard is displayed to the user.
* MSS:
  + User selects to view the leaderboard.
  + System retrieves and displays the leaderboard.
* Extensions:
  + 2a: User decides to stop viewing the leaderboard.
    1. MSS Steps:
       1. System stops displaying the leaderboard.
       2. End of use case.

View Analytics Dashboard (Desktop App)

* UC ID: UC-20
* Description: User views the analytics dashboard.
* Trigger: User selects to view the analytics dashboard.
* Pre-condition: User is logged in.
* Post-condition: Analytics dashboard is displayed to the user.
* MSS:
  + User selects to view the analytics dashboard.
  + System retrieves and displays the analytics dashboard.
* Extensions:
  + 2a: User decides to stop viewing the analytics dashboard.
    1. MSS Steps:
       1. System stops displaying the analytics dashboard.
       2. End of use case.

Customize Analytics Dashboard (Desktop App)

* UC ID: UC-21
* Description: User customizes the analytics dashboard.
* Trigger: User selects to customize the analytics dashboard.
* Pre-condition: User is logged in and the analytics dashboard is displayed.
* Post-condition: Analytics dashboard is customized according to user preferences.
* MSS:
  + User selects to customize the analytics dashboard.
  + System allows user to make changes to the dashboard.
  + User saves the changes.
  + System updates the dashboard according to the changes.
* Extensions:
  + 2a: User decides to cancel the customization.
    1. MSS Steps:
       1. System discards any changes made.
       2. End of use case.

**Use Case Diagram:**

A diagram of a system

Description automatically generated with medium confidence

**Desktop App GUI:**

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer

Description automatically generated

**Web App GUI:**

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

1. Functional Requirement Desktop [↑](#footnote-ref-1)
2. Functional Requirement Website [↑](#footnote-ref-2)
3. Non-functional Requirement [↑](#footnote-ref-3)
4. Use Case [↑](#footnote-ref-4)
5. Main Success Scenario [↑](#footnote-ref-5)