Author: Diana Ciuperca
Date: 09/06/2023
Luna link: https://i505708.luna.fhict.nl/
Git repository: https://git.fhict.nl/I505708/individual-assignment-semester-2.git

Anime Tracker Application

User requirement specifications

Table of Contents

Introduction …………………………………………………………….……………………….……. 1
Functional Requirements ……………………..……………………………………………………… 1
Non- functional Requirements ……………………..……………………………………………… 2
Use Cases – Desktop application ………………………………………………………………….. 2

Introduction

The purpose of this User Requirement Specifications (URS) document is to outline the requirements for the development of an anime tracker application. The application will allow users to keep track of the anime they are watching, see detailed descriptions and receive notifications for new releases. They are able to create an account in order to discover new anime based on their preferences and customize their own profiles.

Functional Requirements

FR-01: Users can log into their account.
Account must already be in the system.

FR-02: Maintainer/Admin can manage anime content.

FR-03: Maintainer/Admin can manage manga content.

FR-04: Maintainer/Admin can manage characters.

FR-05: Maintainer/Admin can delete reviews from registered web users.

FR-06: Admin can manage other accounts (both desktop and web accounts).

FR-07: Web users can manage their profiles.

FR-08: Web users can manage their reviews.

FR-09: All users can see information on the website.

Non-functional requirements

NFR-01 – Performance: The application shall be able to handle a large number of users and requests without significant slowdown or downtime. The application shall have a fast response time, with minimal latency and page load times, and be optimized for performance on both high-end and low-end devices.

NFR-02 – Usability: The application shall have a user-friendly interface that is easy to navigate and understand for both novice and experienced users.

NFR-03 – Reliability: The application shall be reliable, with minimal crashes or errors, and have a backup system in place to prevent data loss.

NFR-04: Security: The application shall protect sensitive data and prevent unauthorized access. It includes factors such as authentication and encryption.

NFR-05: Maintainability: This requirement specifies how easy it is to modify, extend, or fix the system. It might include factors such as modularity, code readability, and documentation.

Use Cases - Desktop application

Use Case 1: Log into the application | FR-01

Actor: Maintainer/Admin

Description: Login can only be done via submitting the correct credentials on the login page.

Prerequisites:

- Actor must have an account already created in the system and a running application
- System requests for login information

Main Success Scenario:

1. Actor enters their personal credentials (username and password) and confirms.
2. System validates the actor's credentials and logs them into their account.

Extensions:

1a: Actor enters wrong credentials and confirms.

1. System displays an error message.
2. Return to MSS1.

1b: Actor indicates they want to set or reset their password.

1. System registers the action, redirects the actor to another login page and requests for a new password.

2. Actor provides a new password and confirms the action.
3. System encrypts and stores the new password, then logs in the actor.
4. End of use case.

Use case 2: Create anime | FR-02

Actor: Maintainer/Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Anime* and indicates they want to add an anime.
2. System displays a popup window and requests for data.
3. Actor provides details about the new anime and confirms the addition.
4. System registers the action, informs the actor the action was successful and returns to the original state.

Extensions:

3a. Actor enters incorrect or invalid data.

1. System displays an error message informing the actor which data is invalid.
2. Return to MSS1.

3b: Actor cancels the addition of the new anime.

1. System returns to the original state.
2. End of use case.

Use case 3: Search anime | FR-02

Actor: Maintainer/Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Anime* and indicates they want to look for a group of anime by using the search feature.
2. System displays a list of anime that match the criteria provided.

Extensions:

1a: Actor doesn’t provide any data to the search feature.

1. System displays all currently existing anime.
2. End of use case.

Use case 4: Update anime | FR-02

Actor: Maintainer/Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Anime* and searches for a specific anime using the search feature (see UC-03 for details).
2. System displays a list of anime that matched the criteria provided (if the information provided is very specific, only one anime will be displayed).
3. Actor selects the anime they were looking for and indicates they want to update its details.
4. System displays a popup window with the current details and requests for changes.
5. Actor changes the relevant details and confirms his action.
6. System registers the action, informs the actor the action was successful and returns to the original state.

Extensions:

5a: Actor confirms the changes with incomplete or invalid data.

1. System displays an error message informing the actor which data is invalid.
2. Return to MSS5.

5b: Actor cancels the update.

1. System returns to the original state.
2. End of use case.

Use case 5: Delete anime | FR-02

Actor: Maintainer/Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Anime* and searches for a specific anime using the search feature (see UC-03 for details).
2. System displays a list of anime that matched the criteria provided (if the information provided is very specific, only one anime will be displayed).
3. Actor selects the anime they were looking for and indicates they want to delete it.
4. System displays a popup window and requests for confirmation.
5. Actor confirms the action.
6. System registers the action and returns to the original state

Extensions:

5a: Actor doesn’t confirm the action.

1. System cancels the removal of the anime and returns to the original state.
2. End of use case.

✚ Use case 6: Sort anime | FR-02

Actor: Maintainer/Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Anime* and indicates they want to see all anime ordered by a category using the sorting feature.
2. System displays a list of all anime ordered by the category indicated.

✚ Use case 7: Create manga | FR-03

Actor: Maintainer/Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Manga* and indicates they want to add a manga.
2. System displays a popup window and requests for data.
3. Actor provides details about the new manga and confirms the addition.
4. System registers the action, informs the actor the action was successful and returns to the original state.

Extensions:

3a. Actor enters incorrect or invalid data.

1. System displays an error message informing the actor which data is invalid.
2. Return to MSS3.

3b: Actor cancels the addition of the new manga.

1. System returns to the original state.
2. End of use case.

✚ Use case 8: Search manga | FR-03

Actor: Maintainer/Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Manga* and indicates they want to look for a group of manga by using the search feature.
2. System displays a list of manga that match the criteria provided.

Extensions:

1a: Actor doesn't provide any data to the search feature.

1. System displays all currently existing manga.
2. End of use case.

Use case 9: Update manga | FR-03

Actor: Maintainer/Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Manga* and searches for a specific manga using the search feature (see UC-08 for details).
2. System displays a list of manga that match the criteria provided (if the information provided is very specific, only one manga will be displayed).
3. Actor selects the manga they were looking for and indicates they want to update its details.
4. System displays a popup window with the current details and requests for changes.
5. Actor changes the relevant details and confirms his action.
6. System registers the action, informs the actor the action was successful and returns to the original state.

Extensions:

5a: Actor confirms the changes with incomplete or invalid data.

1. System displays an error message informing the actor which data is invalid.
2. Return to MSS5.

5b: Actor cancels the update.

1. System returns to the original state.
2. End of use case.

Use case 10: Delete manga | FR-03

Actor: Maintainer/Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Manga* and searches for a specific manga using the search feature (see UC-08 for details).
2. System displays a list of manga that matched the criteria provided (if the information provided is very specific, only one manga will be displayed).
3. Actor selects the manga they were looking for and indicates they want to delete it.
4. System displays a popup window and requests for confirmation.
5. Actor confirms the action.

6. System registers the action and returns to the original state

Extensions:

5a: Actor doesn’t confirm the action.

1. System cancels the removal of the manga and returns to the original state.
2. End of use case.

Use case 11: Sort manga | FR-03

Actor: Maintainer/Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Manga* and indicates they want to see all manga ordered by a category using the sorting feature.
2. System displays a list of all manga ordered by the category indicated.

Use case 12: Create character | FR-04

Actor: Maintainer/Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Characters* and indicates they want to add a character.
2. System displays a popup window and requests for data.
3. Actor provides details about the new character and searches for an already existing anime and/or manga they are related to.
4. Systems displays the anime and/or manga that match the information provided.
5. Actor selects the content needed and confirms the addition of the new character.
6. System registers the action, informs the actor the action was successful and returns to the original state.

Extensions:

5a. Actor enters incorrect or invalid data.

1. System displays an error message informing the actor which data is invalid.
2. Return to MSS5.

5b: Actor cancels the addition of the new character.

1. System returns to the original state.
2. End of use case.

Use case 13: Search character | FR-04

Actor: Maintainer/Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Characters* and indicates they want to look for a group of characters by using the search feature.
2. System displays a list of characters that match the criteria provided.

Extensions:

1a: Actor doesn't provide any data to the search feature.

1. System displays all currently existing characters.
2. End of use case.

Use case 14: Update character | FR-04

Actor: Maintainer/Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Characters* and searches for a specific character using the search feature (see UC-13 for details).
2. System displays a list of characters that match the criteria provided (if the information provided is very specific, only one character will be displayed).
3. Actor selects the character they were looking for and indicates they want to update its details.
4. System displays a popup window with the current details and requests for changes.
5. Actor changes the relevant details and confirms his action.
6. System registers the action, informs the actor the action was successful and returns to the original state.

Extensions:

5a: Actor confirms the changes with incomplete or invalid data.

3. System displays an error message informing the actor which data is invalid.
4. Return to MSS5.

5b: Actor cancels the update.

3. System returns to the original state.
4. End of use case.

Use case 15: Delete character | FR-04

Actor: Maintainer/Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Characters* and searches for a specific character using the search feature (see UC-13 for details).
2. System displays a list of characters that match the criteria provided (if the information provided is very specific, only one character will be displayed).
3. Actor selects the character they were looking for and indicates they want to delete it.
4. System displays a popup window and requests for confirmation.
5. Actor confirms the action.
6. System registers the action and returns to the original state

Extensions:

5a: Actor doesn't confirm the action.

3. System cancels the removal of the character and returns to the original state.
4. End of use case.

✚ Use case 16: Sort characters | FR-04

Actor: Maintainer/Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Characters* and indicates they want to see all characters ordered by a category using the sorting feature.
2. System displays a list of all characters ordered by the category indicated.

✚ Use case 17: Search review | FR-05

Actor: Maintainer/Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Reviews* and indicates they want to look for a group of reviews by using the search feature.
2. System displays a list of reviews that match the criteria provided.

Extensions:

1a: Actor doesn't provide any data to the search feature.

1. System displays all currently existing reviews.
2. End of use case.

Use case 18: Delete review | FR-05

Actor: Maintainer/Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Reviews* and searches for a specific review using the search feature (see UC-17 for details).
2. System displays a list of reviews that match the criteria provided (if the information provided is very specific, only one character will be displayed).
3. Actor selects the review they were looking for and indicates they want to delete it.
4. System displays a popup window and requests for confirmation.
5. Actor confirms the action.
6. System registers the action and returns to the original state.

Extensions:

5a: Actor doesn’t confirm the action.

1. System cancels the removal of the review and returns to the original state.
2. End of use case.

Use case 19: Sort reviews | FR-05

Actor: Maintainer/Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Reviews* and indicates they want to see all reviews ordered by a category using the sorting feature.
2. System displays a list of all reviews ordered by the category indicated.

Use case 20: Create account | FR-06

Actor: Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Accounts* and indicates they want to add an account.
2. System displays a popup window and requests for data.
3. Actor provides details about the new account and confirms the addition.
4. System registers the action, informs the actor the action was successful and returns to the original state.

Extensions:

3a. Actor enters incorrect or invalid data.

3. System displays an error message informing the actor which data is invalid.
4. Return to MSS3.

3b: Actor cancels the addition of the new character.

3. System returns to the original state.
4. End of use case.

Use case 21: Search account | FR-06

Actor: Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Accounts* and indicates they want to look for a group of accounts by using the search feature.
2. System displays a list of accounts that match the criteria provided.

Extensions:

1a: Actor doesn't provide any data to the search feature.

1. System displays all currently existing accounts.
2. End of use case.

Use case 22: Delete account | FR-06

Actor: Admin

Prerequisites:

- Application is open and the actor is already logged in

Main Success Scenario:

1. Actor navigates to the tab *Accounts* and searches for a specific account using the search feature (see UC-18 for details).
2. System displays a list of accounts that match the criteria provided (if the information provided is very specific, only one account will be displayed).
3. Actor selects the account they were looking for and indicates they want to delete it.
4. System displays a popup window and requests for confirmation.
5. Actor confirms the action.
6. System registers the action and returns to the original state

Extensions:

5a: Actor doesn't confirm the action.

1. System cancels the removal of the account and returns to the original state.
2. End of use case.

Use Cases - Web application

Use case 1: Create account | FR-07

Actor: Any user that doesn’t have an account

Prerequisites:

- Actor has the website open

Main Success Scenario:

1. Actor navigates to the page *Register* and indicates they want to create an account.
2. System requests for data.
3. Actor provides their details and confirms the creation.
4. System registers the action, stores the account and redirects the actor to their profile page.

Extensions:

3a. Actor enters incorrect or invalid data.

1. System displays an error message informing the actor which data is invalid.
2. Return to MSS3.

3b: Actor enters an email that is already in use.

1. System displays an error that the email is unavailable.
2. Return to MSS3.

Use Case 2: Log into the account | FR-07

Actor: Registered user

Description: Login can only be done via submitting the correct credentials on the login page.

Prerequisites:

- Actor must have an account already created in the system and an open website
- System requests for login information

Main Success Scenario:

1. Actor enters their personal credentials (username and password) and confirms.
2. System validates the actor's credentials and logs them into their account, redirecting the actor to their profile page.

Extensions:

1a: Actor enters wrong credentials and confirms.

1. System displays an error message.
2. Return to MSS1.

1b: Actor indicates they want to reset their password.

1. System registers the action, redirects the actor to another login page and requests for a new password.
2. Actor provides a new password and confirms the action.

3. System encrypts and stores the new password, then logs in the actor.
4. End of use case.

Use case 3: Manage profile | FR-07

Actor: Registered user

Prerequisites:

- Actor is already logged in

Main Success Scenario:

1. Actor navigates to the page *Profile*.
2. System displays their profile information.

Extensions:

1a. Actor indicates they want to modify their preferred genres.

1. System displays a popup window with all the genres available.
2. Actor selects the ones they are interested in and confirms the action.
3. System registers the action and closes the popup.
4. End of use case.

1b: Actor indicates they want to modify their username.

1. System requests for a new username.
2. Actor provides a new username and confirms the action.
3. System registers the action and updates the profile page.
4. End of use case.

1c: Actor indicates they want to modify their profile picture.

1. System requests for an image.
2. Actor provides an image and confirms the action.
3. System registers the action and updates the profile page.
4. End of use case.

Use case 4: Manage anime lists | FR-07

Actor: Registered user

Prerequisites:

- Actor is already logged in

Main Success Scenario:

1. Actor navigates to one of the *Anime* pages.
2. System displays a list with all available anime.
3. Actor selects an anime indicating they want to see more details about it.
4. System redirects the user to another page containing all details about the selected anime.
5. Actor indicates they want to add the anime the library.
6. System displays a popup window with all anime lists.

7. Actor selects one or more lists and confirms the action.
8. System registers the action and closes the popup.

Extensions:

1a. Actor indicates they want to search for a specific anime by proving a name or a keyword for the name.

1. System displays a list with all anime that match the provided criteria.
2. Return to MSS3.

5a: Actor indicates they want to remove the anime from a list.

1. System displays a popup with all anime lists. The ones the selected anime is part of are displayed as checked while the rest of them are unchecked.
2. Actor unchecks the list they wish to remove the anime from and confirms the action.
3. System registers the action and closes the popup.
4. End of use case.

✚ Use case 5: Manage manga lists | FR-07

Actor: Registered user

Prerequisites:

- Actor is already logged in

Main Success Scenario:

1. Actor navigates to one of the *Manga* pages.
2. System displays a list with all available manga.
3. Actor selects a manga indicating they want to see more details about it.
4. System redirects the user to another page containing all details about the selected manga.
5. Actor indicates they want to add the manga the library.
6. System displays a popup window with all manga lists.
7. Actor selects one or more lists and confirms the action.
8. System registers the action and closes the popup.

Extensions:

1a. Actor indicates they want to search for a specific manga by proving a name or a keyword for the name.

1. System displays a list with all manga that match the provided criteria.
2. Return to MSS3.

5a: Actor indicates they want to remove the anime from a list.

1. System displays a popup with all manga lists. The ones the selected manga is part of are displayed as checked while the rest of them are unchecked.
2. Actor unchecks the list they wish to remove the manga from and confirms the action.
3. System registers the action and closes the popup.
4. End of use case.

Use case 6: Rate a character | FR-07

Actor: Registered user

Prerequisites:

- Actor is already logged in

Main Success Scenario:

1. Actor navigates to one of the *Character* pages.
2. System displays a list with all available characters.
3. Actor selects the full-heart icon of a character indicating they love it.
4. System registers the action, adds the character to favourites, increases the total number of likes for that character and updates the page.

Extensions:

1a. Actor indicates they want to search for a specific character by proving a name or a keyword for the name.

1. System displays a list with all characters that match the provided criteria.
2. Return to MSS3.

3a: Actor selects the broken-heart icon of a character indicating they hate it.

1. System registers the action, adds the character to disliked, increases the total number of dislikes for that character and updates the page.
2. End of use case.

Use case 7: Unrate a character | FR-07

Actor: Registered user

Prerequisites:

- Actor is already logged in

Main Success Scenario:

1. Actor navigates to their *Profile* page.
2. System displays their details and custom lists.
3. Actor navigates to Favourite Characters list, selects a character and indicates they want to remove it.
4. System registers the action, removes the character from favourites, decreases the total number of likes for that character and updates the page.

Extensions:

1a. Actor navigates to Disliked Characters list, selects a character and indicates they want to remove it.

1. System registers the action, removes the character from disliked, decreases the total number of dislikes for that character and updates the page.
2. Return to MSS3.

Use case 8: Add a review | FR-08

Actor: Registered user

Prerequisites:

- Actor is already logged in

Main Success Scenario:

1. Actor navigates to one of the *Anime* or *Manga* pages.
2. System displays a list with all available content.
3. Actor selects a content indicating they want to see more details about it.
4. System redirects the user to another page containing all details about the selected content.
5. Actor indicates they want to leave a review by providing the necessary data and confirming the action.
6. System registers the action and returns the page to the original state.

Extensions:

1a. Actor indicates they want to search for a specific content by proving a name or a keyword for the name.

1. System displays a list with all content that matches the provided criteria.
2. Return to MSS3.

Use case 9: Update review | FR-08

Actor: Registered user

Prerequisites:

- Actor is already logged in

Main Success Scenario:

1. Actor navigates to one of the *Review* pages.
2. System displays a list with all reviews.
3. Actor indicates they want to modify one of their reviews.
4. System displays the current details in an editable format and requests for changes.
5. Actor provides the necessary data and confirms the action.
6. System registers the action and returns the updated page to the original state.

Extensions:

1a. Actor indicates they want to search for a specific content by proving a name or a keyword for the name.

1. System displays a list with all content that matches the provided criteria.
2. Return to MSS3.

Use case 10: Delete review | FR-08

Actor: Registered user

Prerequisites:

- Actor is already logged in

Main Success Scenario:

1. Actor navigates to one of the *Review* pages.
2. System displays a list with all reviews.
3. Actor indicates they want to delete one of their reviews.
4. System displays a popup window asking for confirmation.
5. Actor confirms the action.
6. System registers the action and returns the updated page to the original state.

Extensions:

5a: Actor doesn’t confirm the action.

1. System cancels the removal of the account and returns to the original state.
2. End of use case.

- **Use case 10: View content | FR-09**

Actor: Anyone

Prerequisites:

- Website is already open

Main Success Scenario:

1. Actor navigates to one of the available pages.
2. System displays all content from the selected category.
3. Actor selects a content indicating they want to see more details about it.
4. System redirects the user to another page containing all details about the selected content.

Extensions:

1a. Actor indicates they want to search for a specific content by proving a name or a keyword for the name.

1. System displays a list with all content that matches the provided criteria.
2. Return to MSS3.

4a. Actor indicates they want to use one of the available features (custom library, reviews, rate characters).

1. System displays a message that they need to log in to access those features or redirects them to Login.
2. End of use case.