Requirement analysis

1. The project topic chosen was an anime recommendation system for content streamed in Crunchyroll.

Functional Requirements	• The system must allow the user to create a personal account with a username and password. • The system must allow the user to login to their account after providing their login credentials. • The system must allow the user to log out of their account at any time. • The system must show all available animes in the Crunchyroll streaming platform. • There should be a search function that allows the user to search for a specific anime in the database by entering the name or a key word. • The system must allow the users to filter down their search according to genre tags. • The user should be able to select an anime of their interest and get redirected to the info page of that anime, which will contain data like stars per genre, episodes, etc. • The info page should contain an "add to favorites" and "remove from favorites" button that will allow the user to personalize their favorites list. • The user should be able to access their favorites list. • The system must provide the user with relevant information on database statistics through charts.
Non-functional requirements	**Operational:** • The website should be able to run on any browser and screen resolution. **Interface:** • The website should contain a consistent format that is pleasant for the user's eyes. • The buttons and functions of the website should be user friendly, intuitive for non-tech enthusiasts. **Security:**

	• The system must hash the passwords of users before storing them in the database. • The only people allowed to access this database are the MySQL users with privileges. **Maintainability:** • Because the system will need to handle several databases and entries, the code needs to be kept in an organized manner so that future modification won't result in coordination failures. **Constraints:** • The system needs to be implemented in html, php and MySQL.

Use Case Analysis

Use Case Name: Login to account	ID: UC-1	Priority: High
Actors: User, System		
Description: The use case describes what happens when a user tries to login to their account		
Trigger: The user clicks on the login button		
Preconditions: The anime recommendation system website is functional and online. The user has clicked on the website's login link		
Basic path: 1. The user enters its username and password. 2. The user clicks on the login button. 3. The system validates credentials. 4. The user is redirected to the home page. 5. The user is greeted with the anime database ranked by score.		

Alternative path 1 (The user doesn't have an account):

A1. The user clicks on the "Register here" hyperlink (branch from step 1).

A2. The user enters his username and password (twice for confirmation).

A3. The user clicks on the register button.

A4. The system validates the username and password entered.

A5. The user is redirected to the login page for further login.

Postconditions:

The user has logged in to their account and redirected to the website's home page.

Exceptions:

E1. The user doesn't have an account, or the credentials are invalid (fire from step 1)

   System displays a message saying that the credentials are invalid.

E2. The username is already in use or has invalid characters (fire from step A3)

   System displays a message telling the user to choose another username.

E3. The passwords don't match (fire from step A3)

   System displays a message saying that the passwords do not match.

Use Case Name: Search for an anime	**ID:** UC-2	**Priority:** High
Actors: User, System		
Description: The use case describes what happens when a user searches for an anime.		
Trigger: The user clicks on the search button.		
Preconditions: The anime recommendation system website is functional and online. The user is currently on the website's home page ("index.php")		
Basic path: 1. The user enters an anime title on the search bar.		

| 2. The user clicks on the search button. |
| 3. The system validates the contents of the input. |
| 4. The system fetches the data from the database. |
| 5. The user is redirected to a search page with the fetched contents. |

Alternative path 1 (The user specifies a genre tag):

A1. The user selects one of the genre checkboxes (branch from step 1).

A2. The user clicks on the search button.

A3. The system fetches the data that satisfies these conditions.

A4. The user is redirected to a search page with the fetched data.

Postconditions:

The system has displayed the requested data to the user.

Exceptions:

E1. The user has provided a name not included in the database (fire from step 3)

  System displays a message saying that there are no animes that match that name.

E2. The user has provided a combination of genres or name and genres not included in the

database (fire from step A2)

  System displays a message saying that there are no animes that match those conditions.

Use Case Name: Add/remove an anime from favorites	**ID:** UC-3	**Priority:** High
Actors:		
User, System		
Description:		
The use case describes what happens when a user adds or removes an anime from their favorites.		
Trigger:		
The user clicks on the add to favorites or remove from favorites button.		

Preconditions:

The anime recommendation system website is functional and online.

The user is currently on an anime's info page.

Basic path:

1. The user has clicked on an anime's image.

2. The user is redirected to its info page.

3. The user clicks on the add to favorites button.

4. The system checks for the existence of the same anime in favorites.

5. The system adds the anime to the favorites table mapped with the user's name.

6. The user is redirected to the favorites list page.

Alternative path 1 (The user clicks on remove from favorite):

A1. The user clicks on the remove from favorites button (branch from step 3).

A2. The system checks for validity of delete query.

A3. The system removes the anime from the favorites table.

A4. The user is redirected to the favorites list page.

Postconditions:

The system has updated the favorites list and the user is redirected to the favorites page.
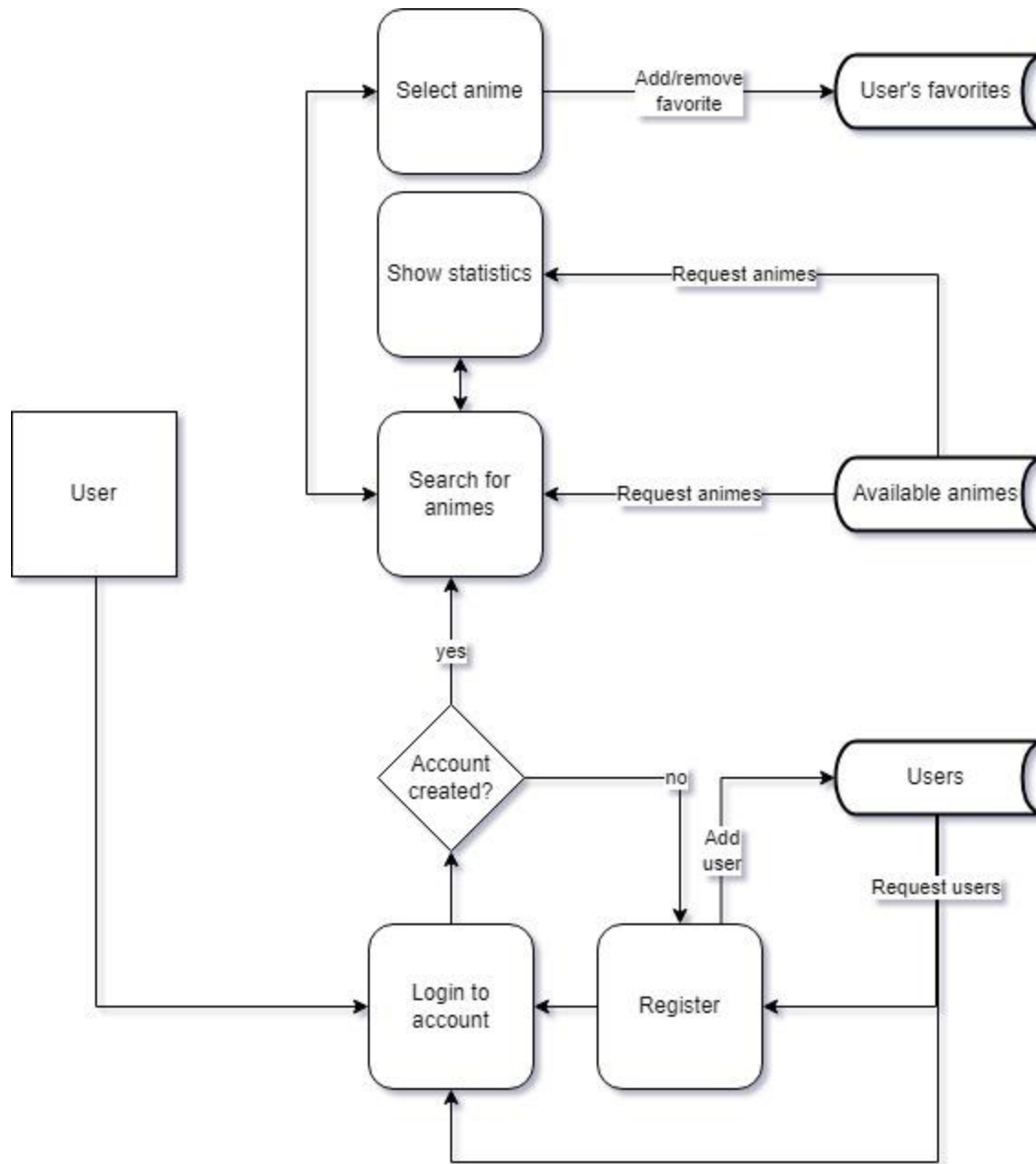
Exceptions:

E1. The user tries to remove an anime that is not in its favorites (fire from step A3)

    System goes over instruction and shows the user its current favorites.

E2. The user tries to add an anime that is already in its favorites (fire from step 3)

    System goes over instruction and shows the user its current favorites.

Data Flow Diagram

Entity Relationship Diagram

animes **users**
- ID : int(11) unsigned
- name : varchar(15)
- password : varchar(255)

animes **user_anime**
- ID : int(11)
- userID : int(11) unsigned
- animeID : int(11) unsigned

animes **animes**
- ID : int(11) unsigned
- title : varchar(80)
- url : varchar(105)
- img : varchar(92)
- # episodes : smallint(6)
- # weight : int(11)
- # rate_1 : int(11)
- # rate_2 : int(11)
- # rate_3 : int(11)
- # rate_4 : int(11)
- # rate_5 : int(11)

animes **anime_genre**
- ID : int(11)
- animeID : int(11) unsigned
- genreID : int(11) unsigned

animes **genres**
- ID : int(11) unsigned
- name : varchar(25)