



SOFTWARE ENGINEERING

GROUP NAME - NomNom

MEMBERS

Aneeta Olorode - OLO22612303

Sumana Asgar - ASG22615054

Maahia Rahman - RAH23614335

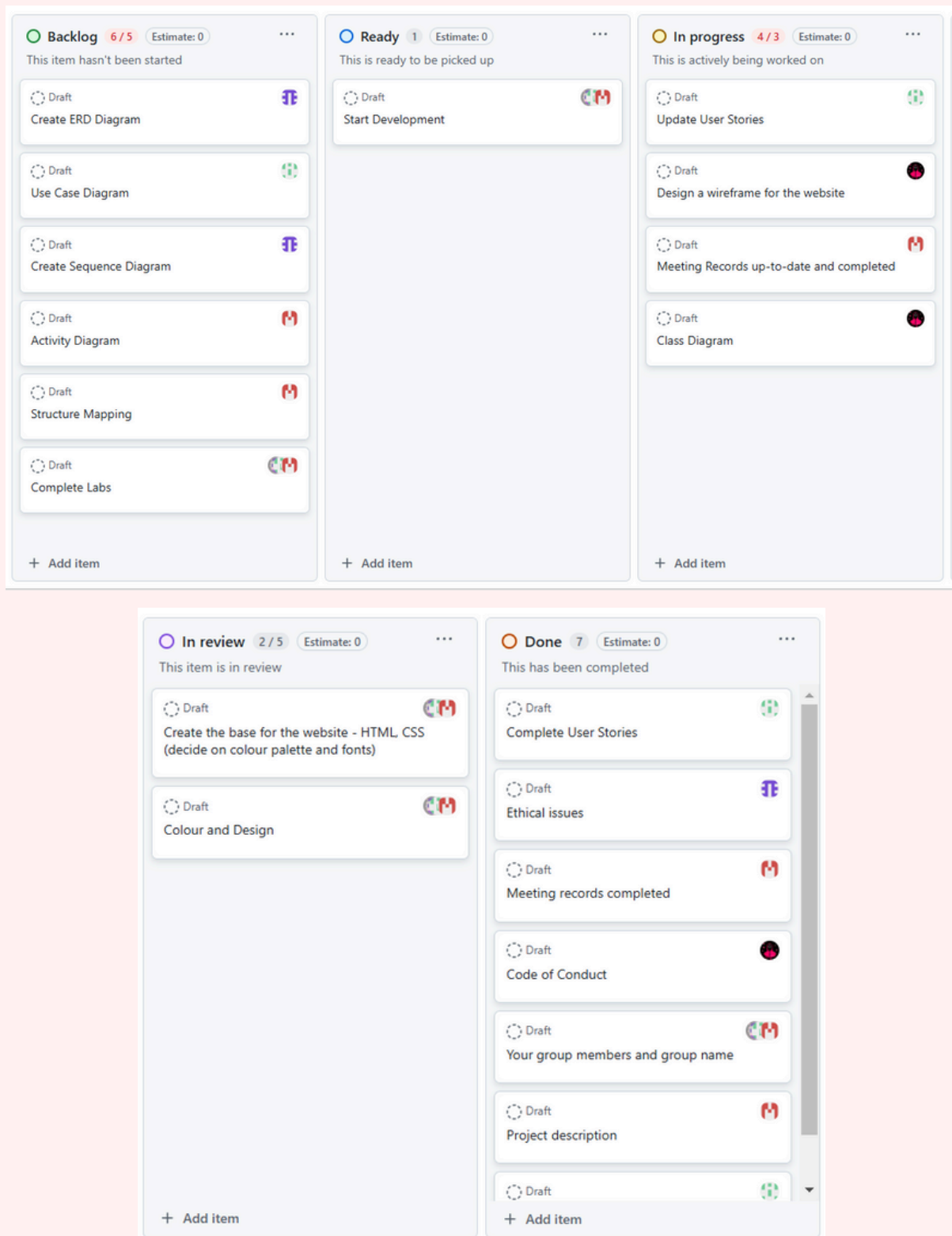
Shaiza Sultana - SUL22528984

Contents

1	Overview - Kanban Board
2	Design Outline
3	Wireframes
4	Structure Map
5	Class Diagram
6	Use Case Diagram
7	ERD Diagram
8	Activity Diagrams
9	Sequence Diagram
10	User Stories
11	Meeting Minutes

Overview

For Sprint 2, we will outline the application's features in the form of user stories, present use case diagrams, wireframes, and an ERD diagram. Additionally, we will showcase our updated Kanban board, along with the chosen colour scheme and design concepts.

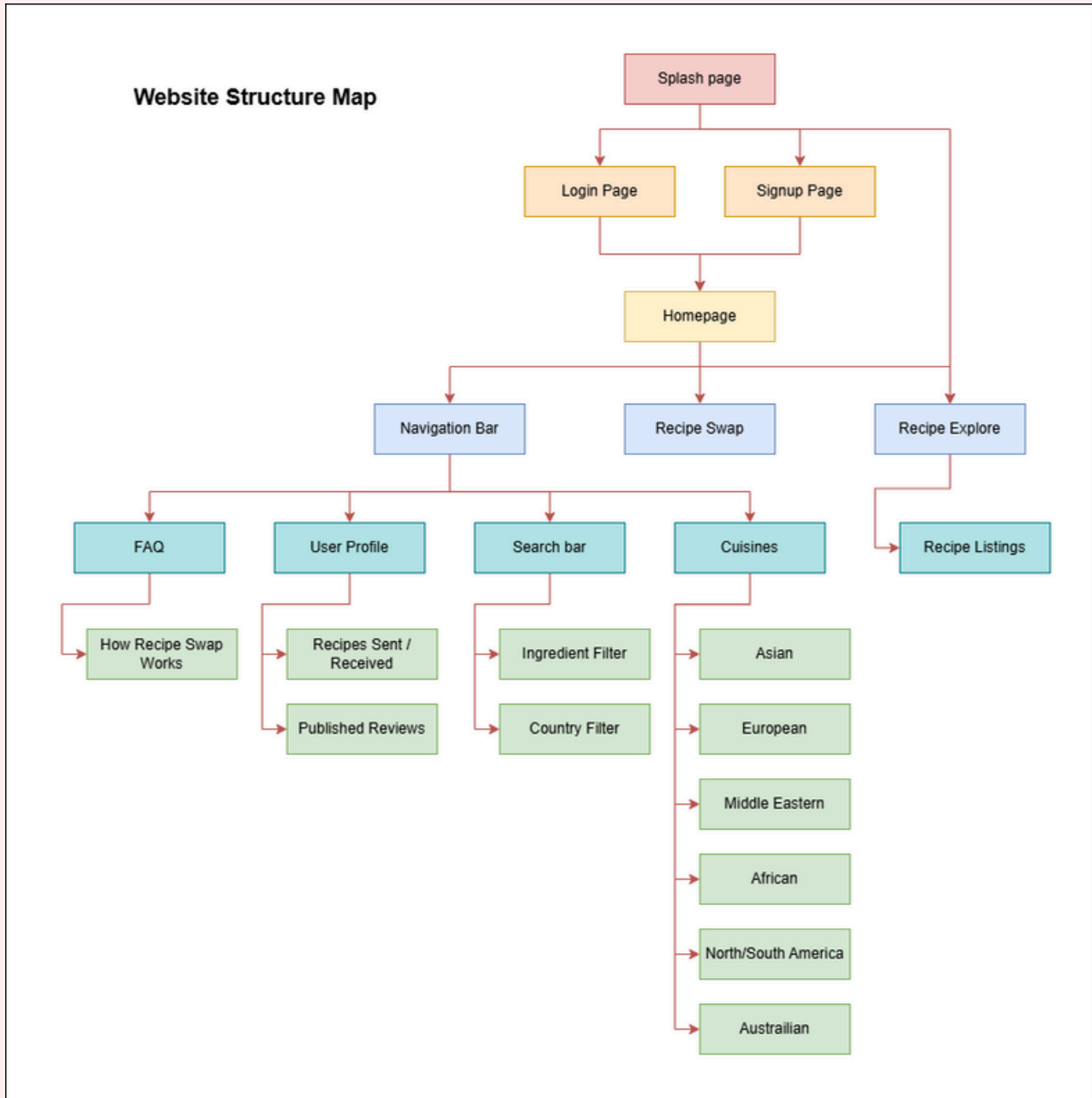


Design Outline

The Recipe Swap platform uses a bold orange, red, and white colour scheme to create a warm and engaging user experience. Red and orange highlight key actions, while a white background ensures readability. Dark red text provides contrast, and gradients in headers add depth. The grid-based layout, inspired by MVC principles, ensures scalability and responsiveness across devices. A mix of serif and sans-serif fonts enhances readability, balancing professionalism with a fun aesthetic.

Navigation is intuitive, with global cuisine categories, search, profile, and dark mode toggles, and a dedicated Recipe Swap section. The swap feature follows an MVC approach, where users submit recipes (Model), swaps are matched dynamically (Controller), and available trades are displayed (View). Inspired by Pinterest and AllRecipes, this encourages user interaction and engagement. The design prioritises usability, scalability, and a visually striking experience.

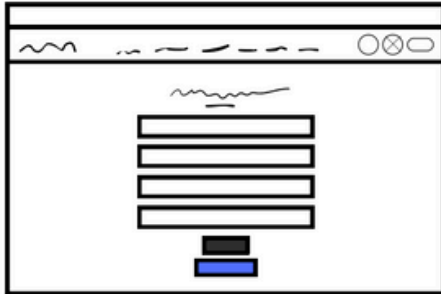
Structure Map



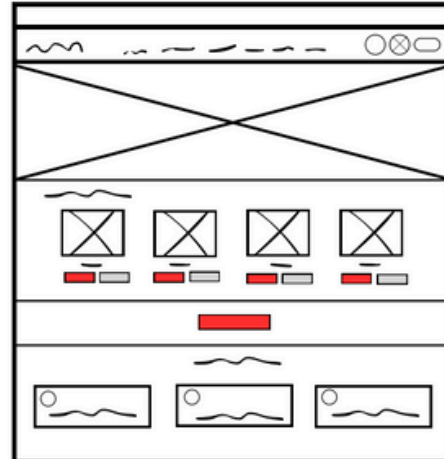
Wireframe

Low Fidelity:

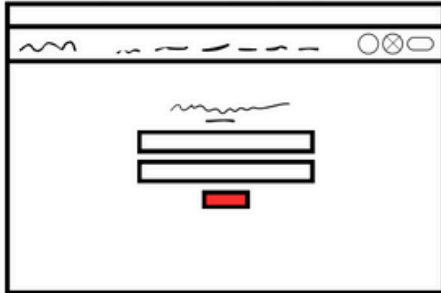
Create account page



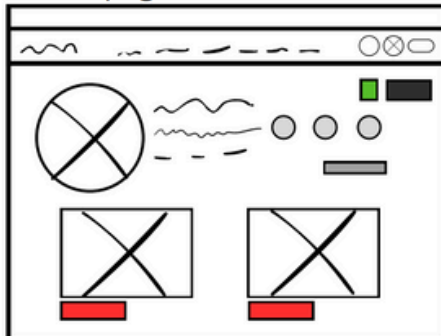
Main explore page



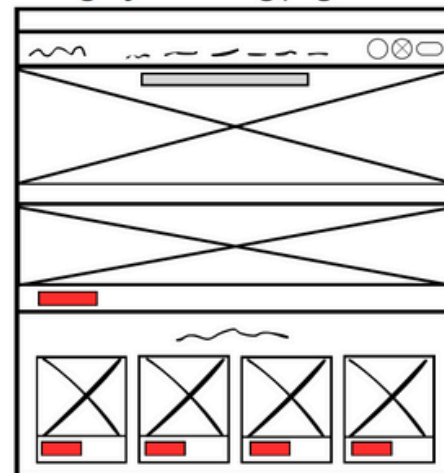
Existing users login page



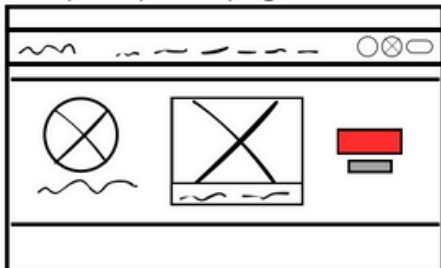
Profile page



Category browsing page

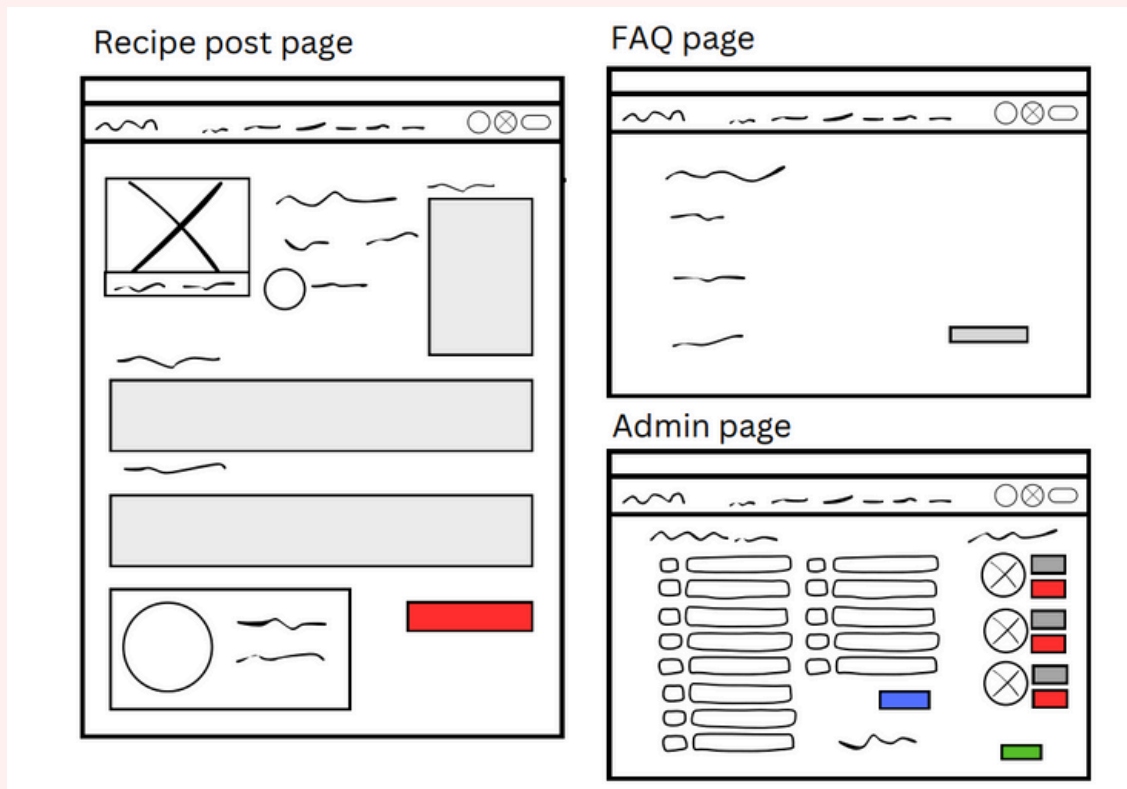


Swap Requests page



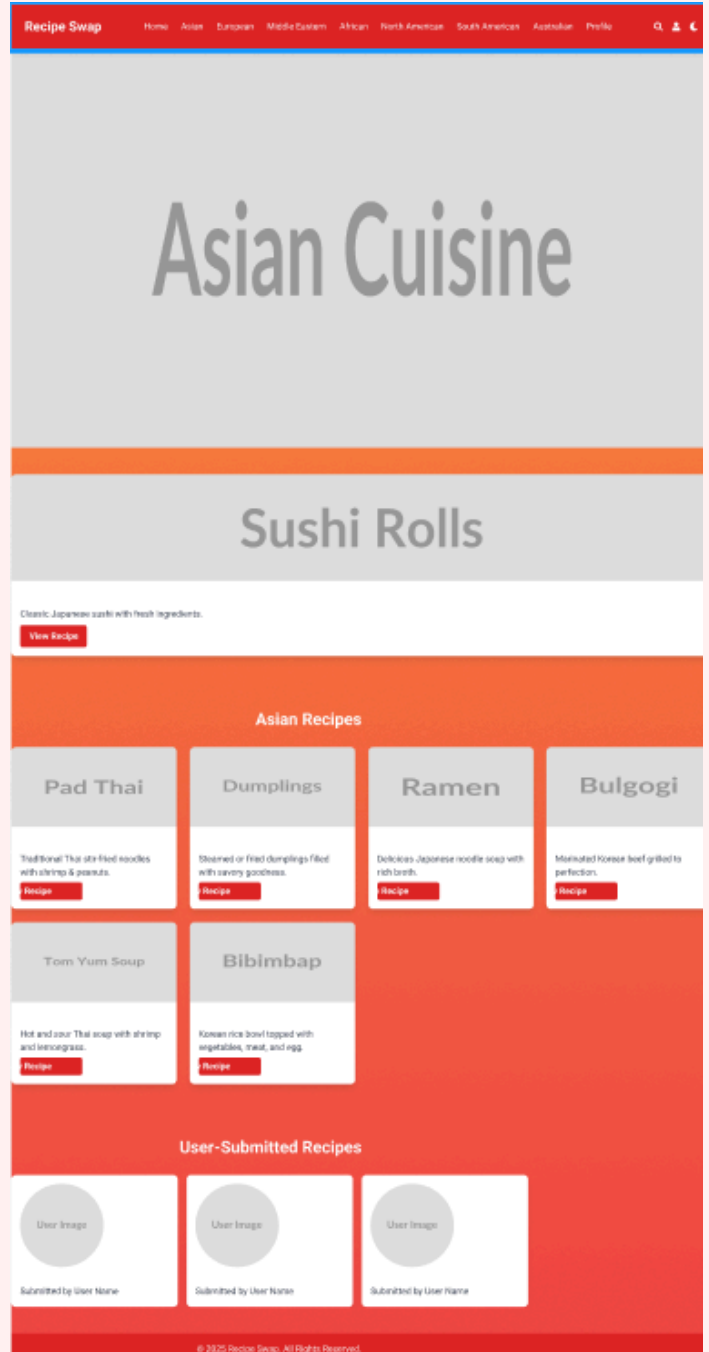
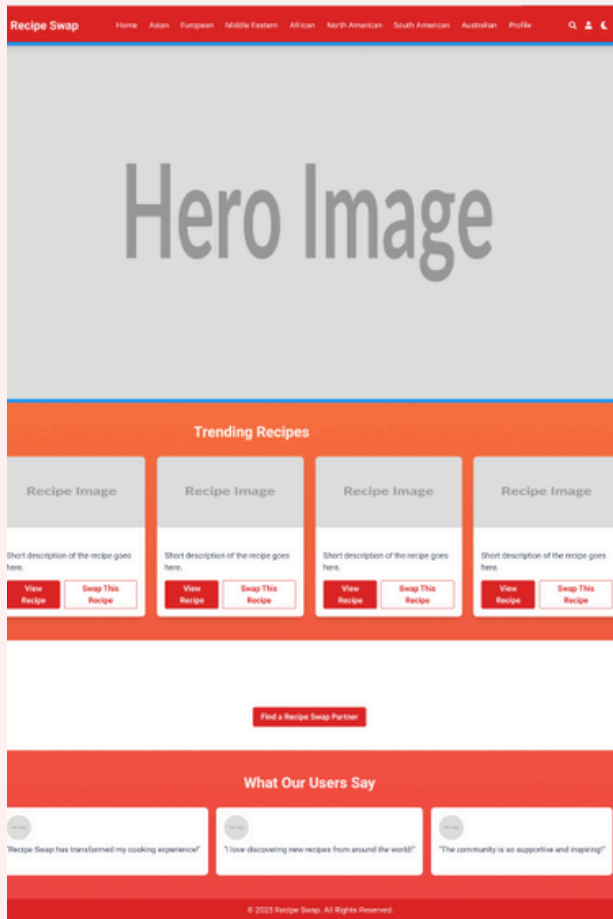
Wireframe

Low Fidelity:

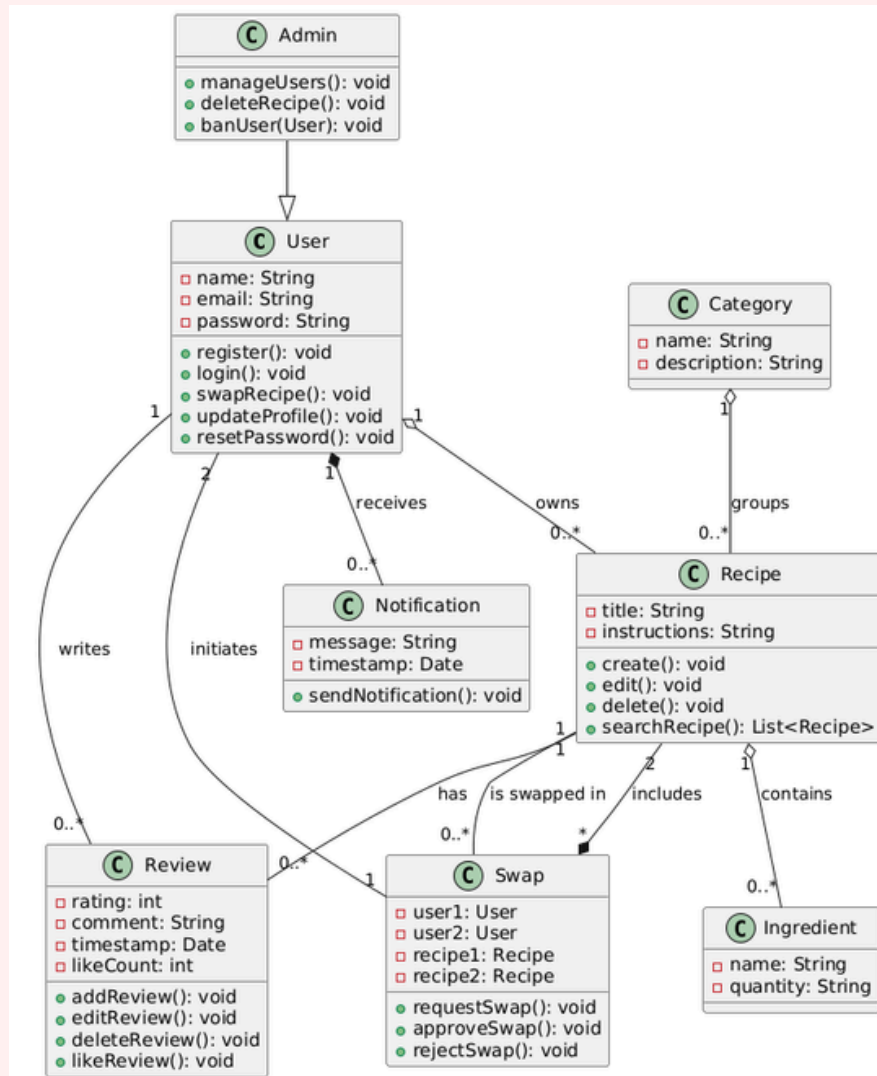


Wireframe

High Fidelity:



Class Diagram

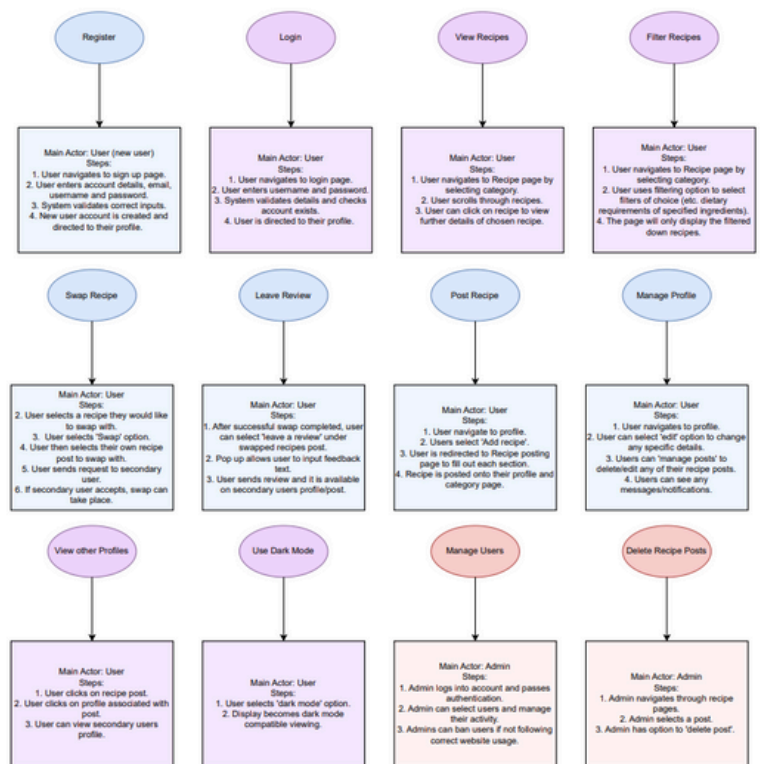


The class diagram uses various relationships, **including association, aggregation, and inheritance**.

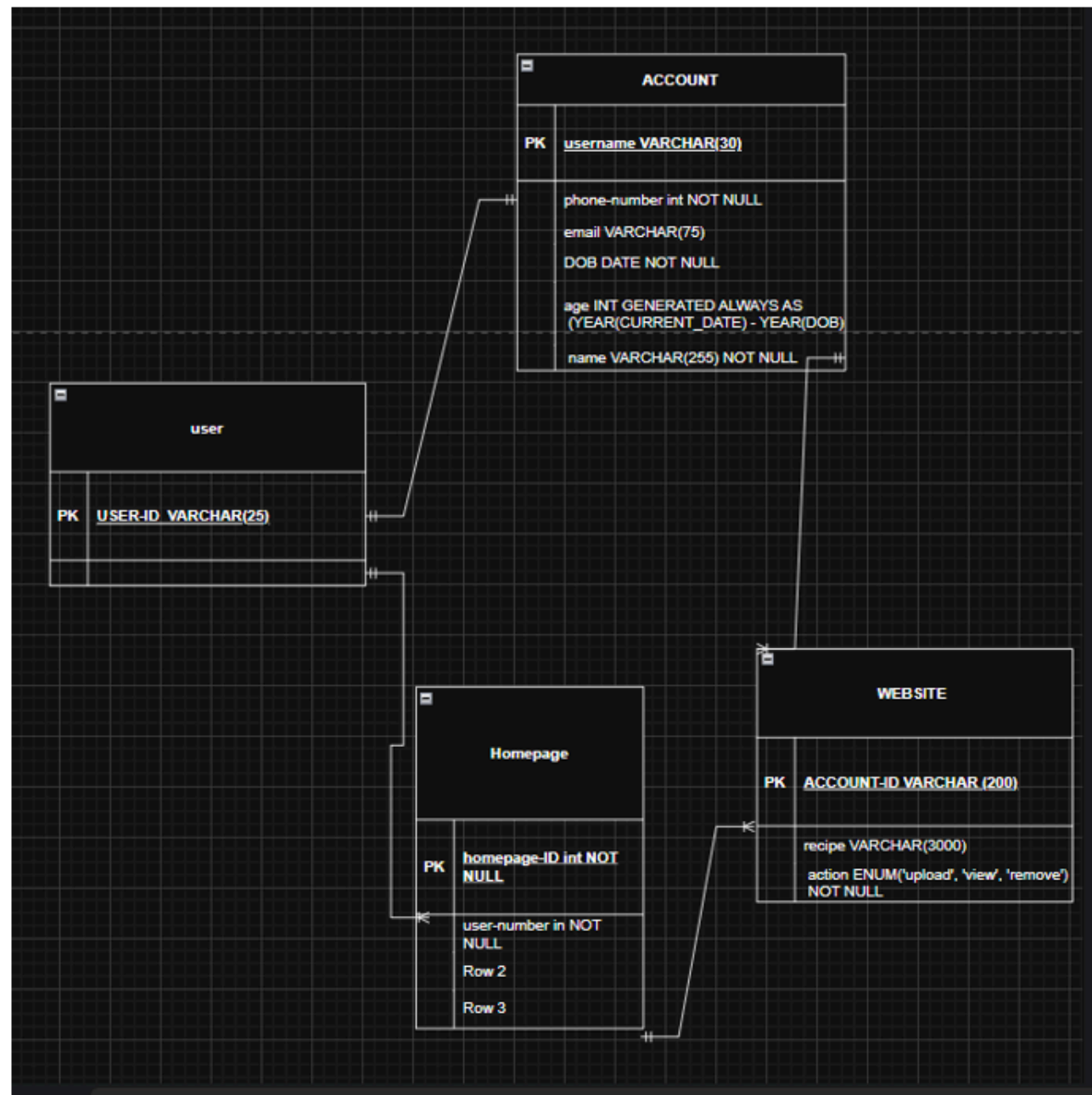
- **Inheritance:** Admin inherits from User, meaning an admin has additional privileges like managing users and deleting recipes.
- **Aggregation:** Recipe is associated with multiple Ingredient objects, meaning a recipe contains ingredients, but ingredients can exist independently. Similarly, Category groups multiple recipes.
- **Composition:** User writes Reviews and initiates Swaps, implying strong ownership where reviews belong to users and swaps involve specific users and recipes.
- **Associations:**
 - User owns multiple Recipes.
 - User receives Notifications.
 - Swap involves two Users and two Recipes.
 - Recipe includes Ingredients.

These relationships define how components interact within the system.

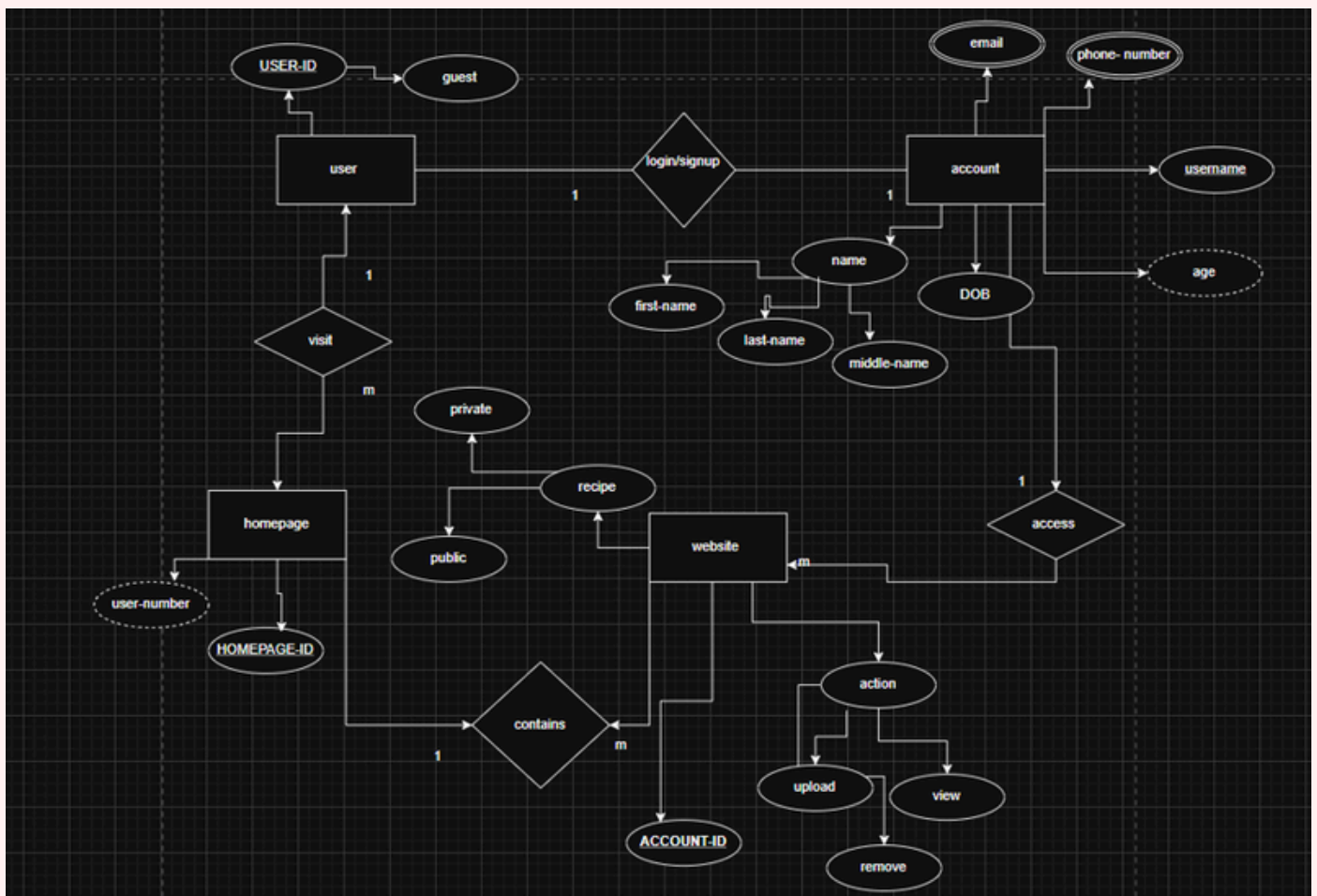
Use Case Diagram



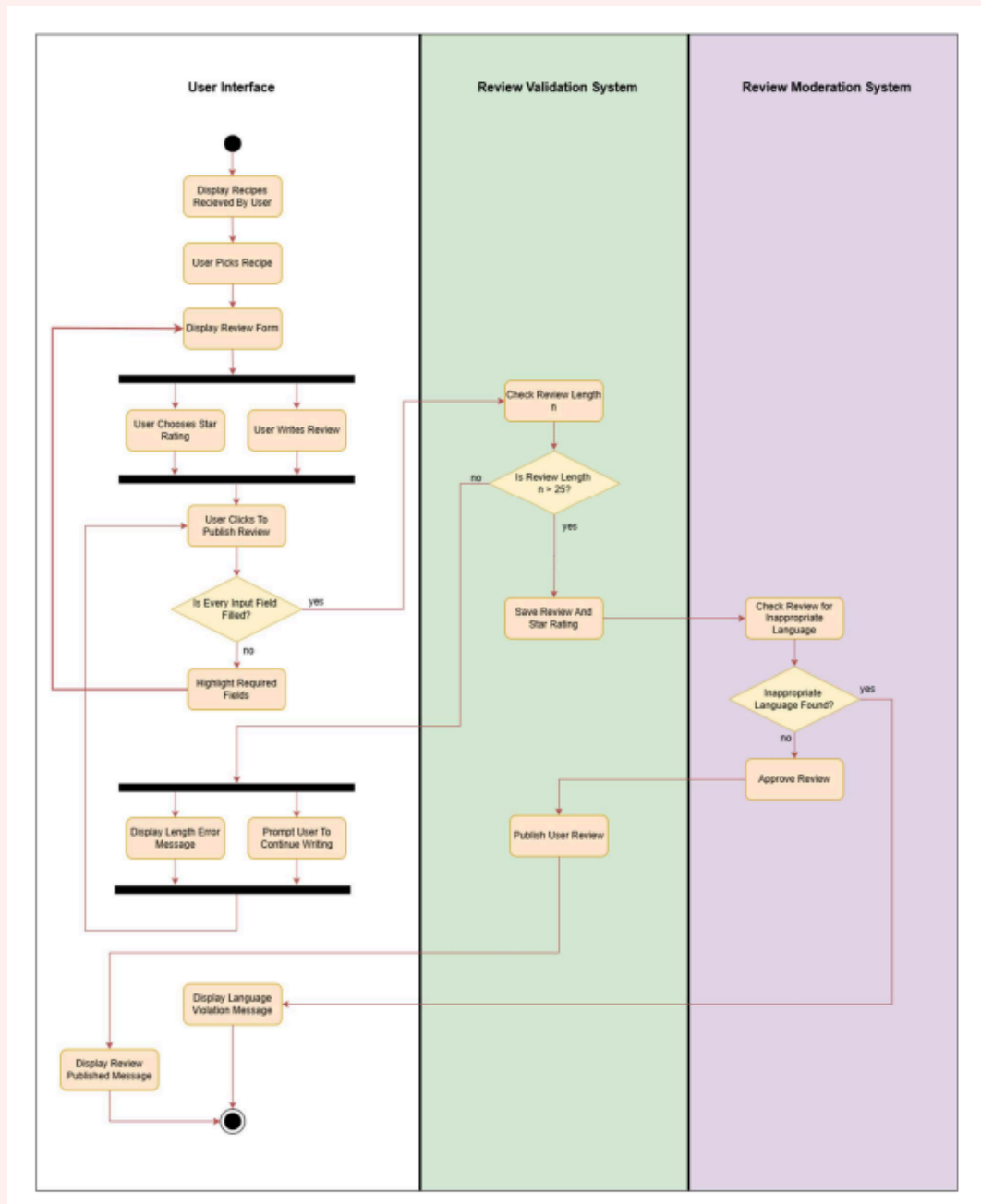
ERD



ERD (2)

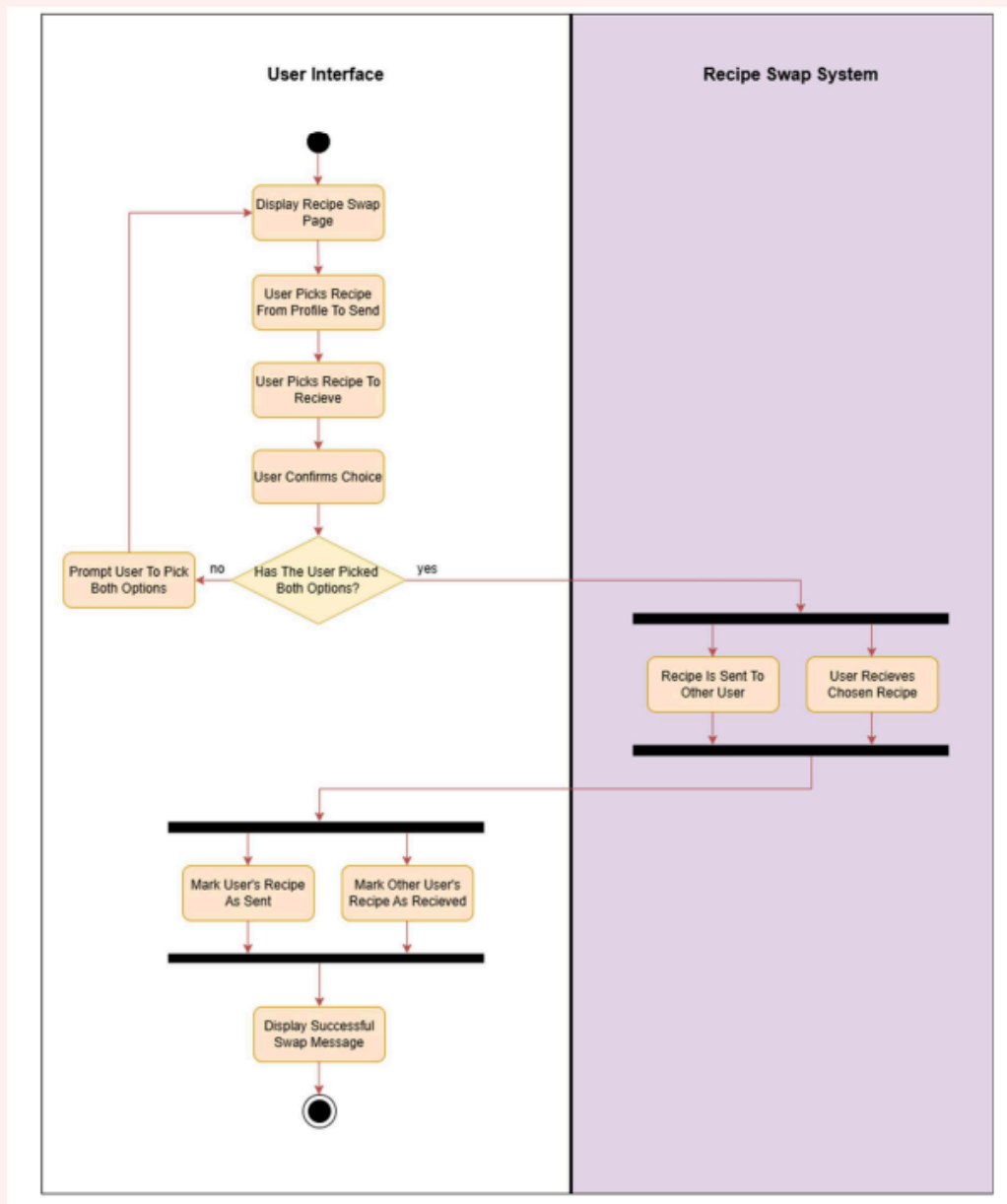


Activity Diagram



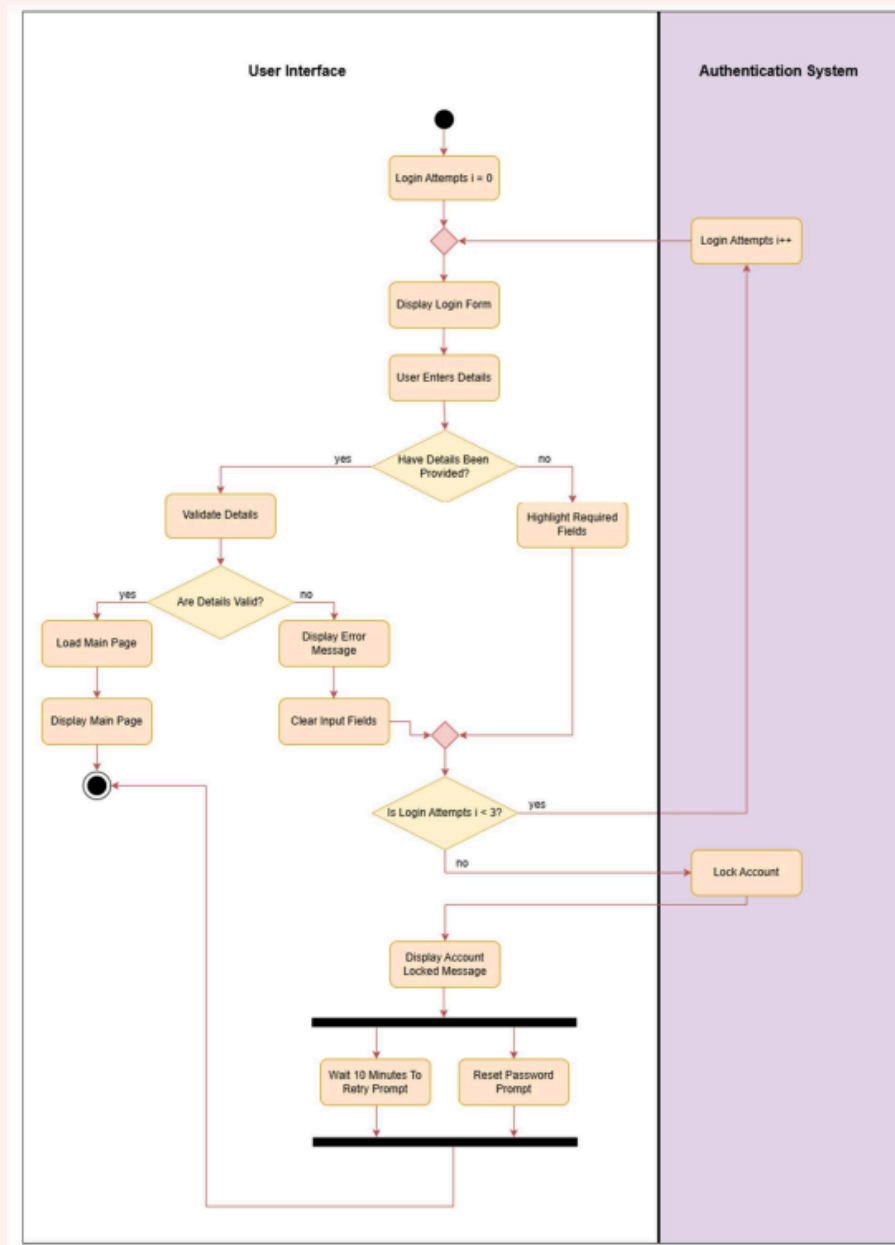
The diagram shows the reviewing system. When a user has received and tried a recipe, they are able to post a review. They are given a review form where they can add a star rating and written review. There are checks for the length of the written review and the content when they try to publish it. This is for moderation checks, to make sure the website remains constructive and positive. If the user doesn't meet the length or content requirements, they are notified and can continue to write the review or start again. If all the content is fine, the review is published.

Activity Diagram (2)



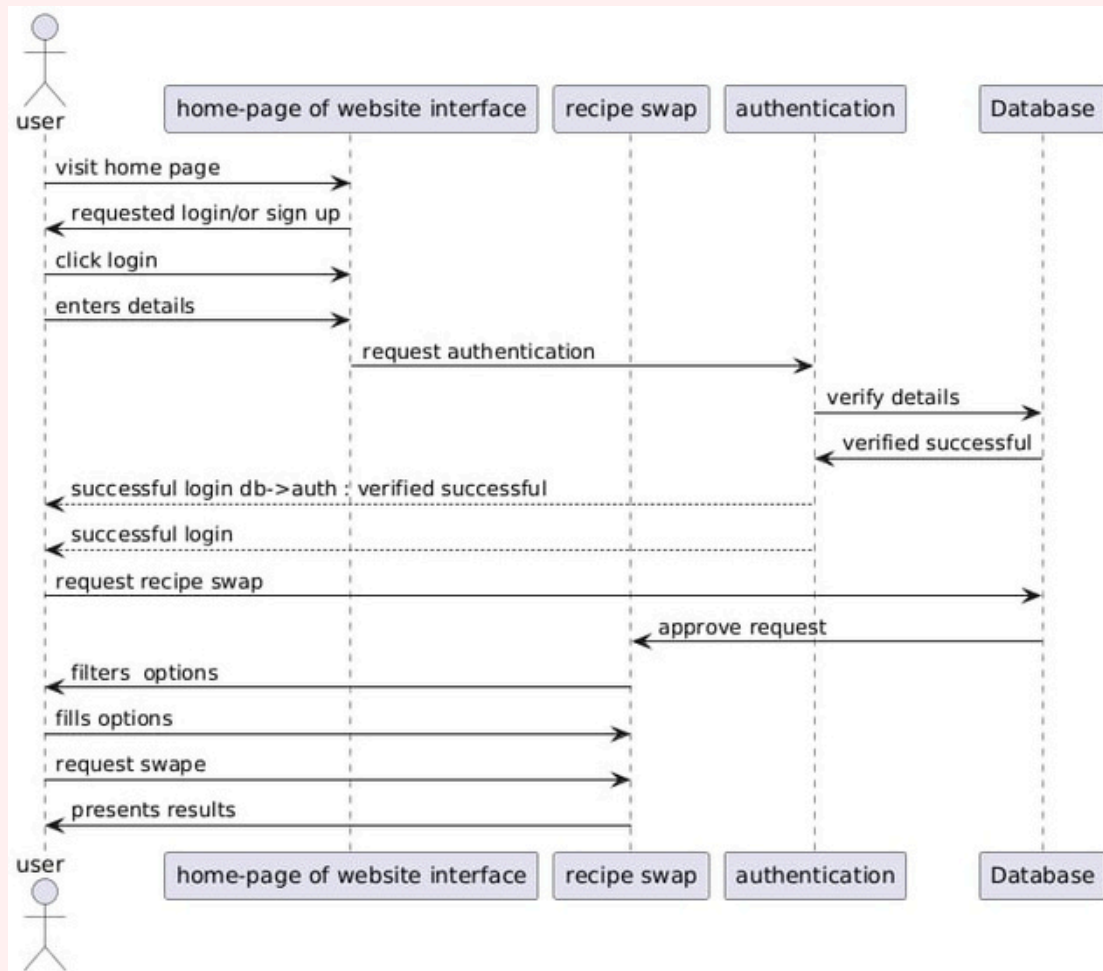
For the recipe swap system, the user is only able to swap a recipe if they already have one on their profile. If they do, they choose the recipe they want to swap then the one they want to receive. When the user confirms, there is a check to make sure they've picked both options and if they haven't, they are prompted to do so. When both recipes have been chosen, the user receives the recipe and theirs is sent at the same time. The user then sees confirmation that the recipes have been swapped successfully.

Activity Diagram (3)



This diagram shows the ideal flow for the login system for when the user has an existing account. On the interface side, the user is shown the login form and starts with 0 login attempts. They then input their login details and if they match what is in the database, they are logged in. If the details are wrong, the user is notified and are then prompted to try again. If the details are inputted incorrectly 3 times, the account is locked, and they are given the option to wait and try again or reset their password.

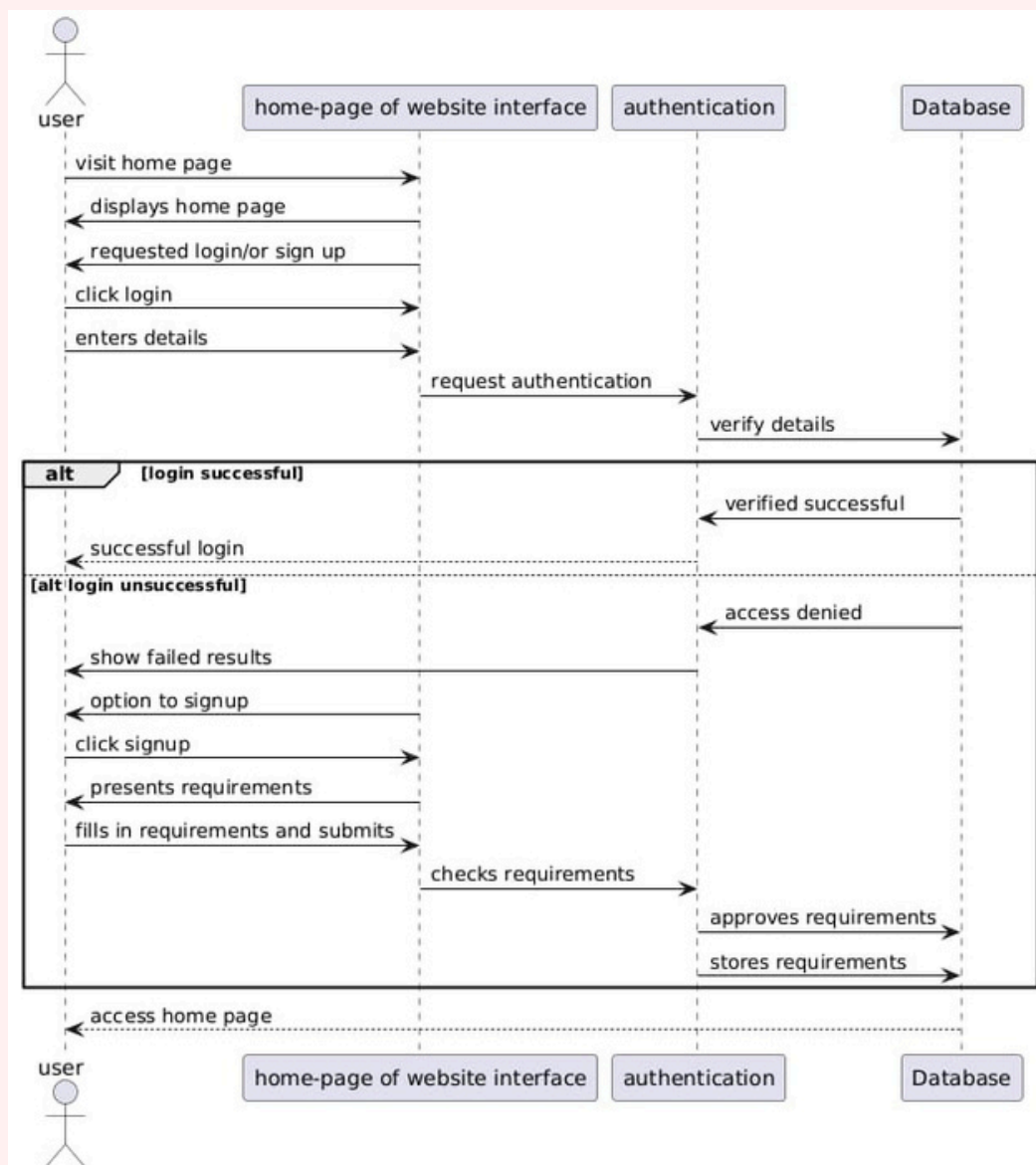
Sequence Diagram



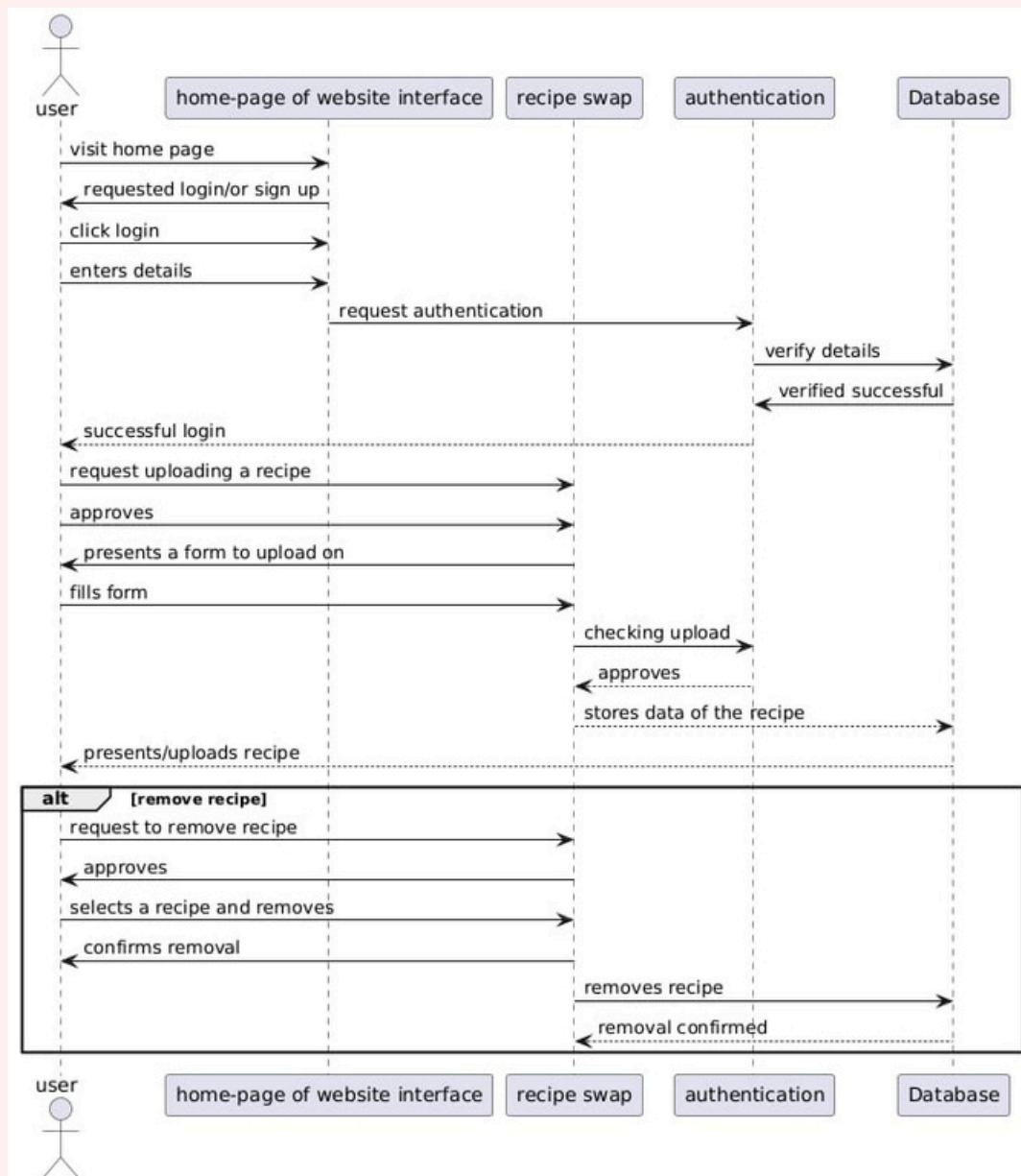
These sequence diagrams represent interactions in a recipe management system, focusing on user authentication, recipe management recipe review, and recipe swapping. Users can log in, sign up, upload, remove, review, and swap recipes. The system authenticates users, verifies credentials, and manages recipes through approvals and database storage.

Admins oversee recipe removal, while the system provides filtering and approval mechanisms for recipe swaps. The diagrams highlight how different actors (users, authentication, database, and the website interface) collaborate in these processes.

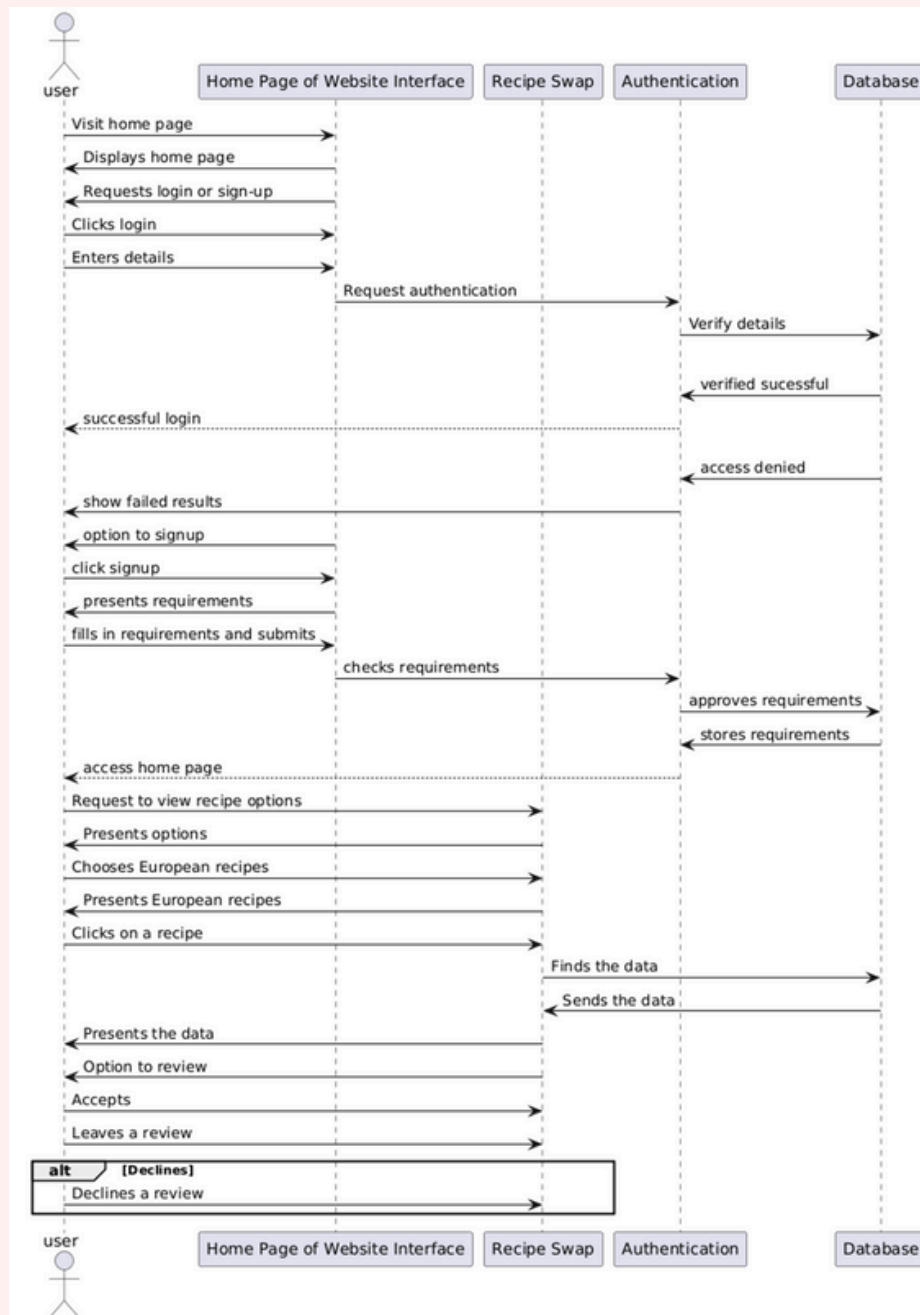
Sequence Diagram (2)



Sequence Diagram (3)



Sequence Diagram (4)



User Stories

User Stories

1. As a **[user]** I want to **[post my recipes]** to a wider range of people.
 2. As a **[user]** I want to **[browse recipes]** with a filtered approach.
 3. As a **[user]** I want to **[view my profile]** where I can **[edit my details and posts]**.
 4. As a **[user]** I want to **[select a recipe]** so that I can **[swap my own with it]**.
 5. As a **[user]** I want to **[leave reviews]** after trying out a recipe swap.
 6. As a **[user]** I want to **[use a dark-mode option]** to **[decrease my eye strain]**.
 7. As a **[user]** I want to [easily navigate the instructions of a recipe] with titled sections.
 8. As a **[user]** I want to **[view others profiles]** so that I can [network with them and follow more of their recipes].
 9. As an **[admin]** I want to **[manage users]** so that I can **[check their activity and ban any unauthorised users]**.
 10. As an **[admin]** I want to **[delete recipe posts]** if they do not meet the guidelines.
 11. As a **[user]** I want to **[view others profiles]** so that I can **[network with them and follow more of their recipes]**.
 12. As an **[admin]** I want to **[view user made content like reviews and recipes]** so I can **[moderate the content]**.
 13. As an **[admin]** I need **[special privileges that let me suspend accounts and delete user made content that violate guidelines]** to **[maintain a positive environment]**
-

meeting minutes

meetings

Project Name	Recipe Swap
Date and Time	27/01/25 16:00
Meeting Goal	<ul style="list-style-type: none"> • agree on code of conduct • choose group name • choose project
Facilitator	N/A
Note Taker	Aneeta
Attendees	Aneeta, Maahia, Shaiza, Sumana
Roundtable Updates	N/A
Discussion Points	<ul style="list-style-type: none"> • group name - we decided on NomNom • project selection - we chose recipe swap. reading through the list, we already had lots of ideas for the design and content • meeting days - we'll meet at least once a week to show each other what we've done and give feedback
Actions	<ul style="list-style-type: none"> • Project description refined – Aneeta • Code of Conduct – Sumana • Personas – Maahia • Ethical Issues – Shaiza • Meeting minutes – Aneeta • Create wireframe – all

meetings

Project Name	Recipe Swap
Date and Time	03/02/25 14:54
Meeting Goal	<ul style="list-style-type: none"> • discuss any updates
Facilitator	Aneeta
Note Taker	Aneeta
Attendees	Aneeta, Maahia, Shaiza, Sumana
Roundtable Updates	<ul style="list-style-type: none"> • Maahia – completed 2 personas • Sumana – completed code of conduct • Aneeta – completed project description • Shaiza – completed and uploaded ethical issues to GitHub
Discussion Points	<ul style="list-style-type: none"> • What we got done – we all managed to complete our designated tasks but some of us need to upload it to github. We set up the new repository and invited everyone as collaborators. Also discussed what should be on the Kanban board
Actions	<ul style="list-style-type: none"> • Upload any outstanding tasks to github – all • Complete Kanban board – all • Add all items into one pdf file – Aneeta • Set up docker environment – all • Upload document to Moodle - Shaiza

meetings

Project Name	Recipe Swap
Date and Time	10/02/2025 2-5 PM
Meeting Goal	<ul style="list-style-type: none"> • Brainstorm coding ideas and methods for the project · • Review sprint 2 task and splitting them • Deciding on a colour scheme
Facilitator	Shaiza
Note Taker	Shaiza
Attendees	Aneeta, Maahia, Shaiza, Sumana
Roundtable Updates	<ul style="list-style-type: none"> • Aneeta – looking for API to add to the product • Maahia – discuss sprint 1 and suggest ideas for sprint 2 • Shaiza – completing note sheet • Sumana – setting up docker environment All – editing/improving sprint 1
Discussion Points	<ul style="list-style-type: none"> • · Review the code of conduct and add the missing information • How the swape idea and how it will function • Colour scheme for the project • Layout for home pages and other pages • Add GDPR and data privacy · Copyright scan/filter per recipe
Actions	<ul style="list-style-type: none"> • Aneeta – colour scheme and finding useful coding skills for the project • Maahia – edit user stories based on feedback • Shaiza – editing ethical issues based on feedback • Sumana – edit code of conduct based on feedback • All – being clear on what is needed to be done for this project

meetings

Project Name	Recipe Swap
Date and Time	17/02/2025 2-5 pm
Meeting Goal	<ul style="list-style-type: none">• Brainstorm coding ideas and methods for the project• Review sprint 2 task and splitting them• Deciding on a colour scheme
Facilitator	Shaiza
Note Taker	Shaiza
Attendees	Aneeta, Maahia, Shaiza, Sumana
Roundtable Updates	<ul style="list-style-type: none">• Select tasks for sprint 2• Finish lab work• Discuss the points for the website so everything is clear and easy to understand• Aneeta – structure diagrams• Maahia – use case diagrams• Shaiza –sequence diagrams• Sumana – wireframes•
Discussion Points	<ul style="list-style-type: none">• colour schemes , designs• function• splitting tasks for sprint 2• labs activities
Actions	<ul style="list-style-type: none">• Aneeta – activity diagrams, structure diagram• Maahia – user stories , use case diagram• Shaiza – sequence diagrams, ERD diagram• Sumana – edit document, wireframes, class diagram• All – designs, colour schemes, Kanban board

LINKS

GitHub

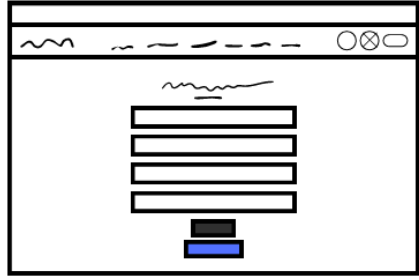
<https://github.com/maahiarahman/Software-Engineering-Project>

Kanban Board

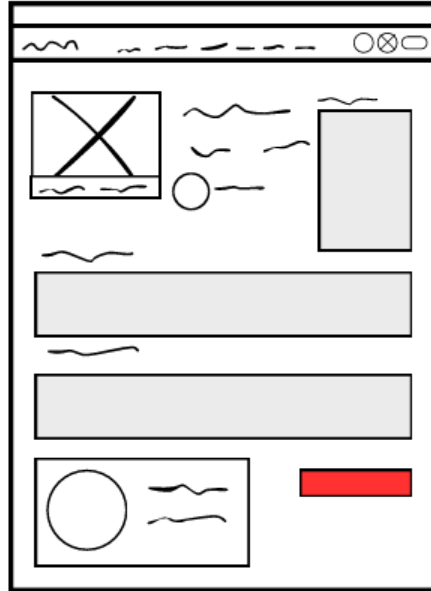
<https://github.com/users/maahiarahman/projects/1>

“As a [user] I want to [post my recipes] to a wider range of people.”

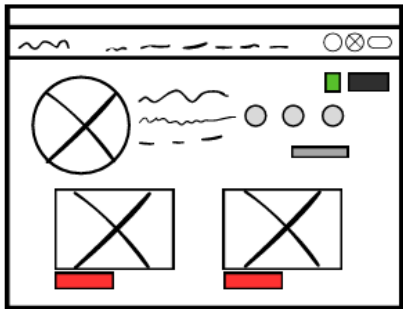
>>Account creation/Login



>>Recipe post page with sections to fill



>>View profile



- User login or create their account.
- go to their profile and create a post. They must then fill in the required sections.
- They can click publish when done.

Register

Post Recipe

Use case scenarios

Main Actor: User (new user)

Steps:

1. User navigates to sign up page.
2. User enters account details, email, username and password.
3. System validates correct inputs.
4. New user account is created and directed to their profile.

Main Actor: User

Steps:

1. User navigate to profile.
2. Users select 'Add recipe'.
3. User is redirected to Recipe posting page to fill out each section.
4. Recipe is posted onto their profile and category page.

Recipe

title: String

instructions: String

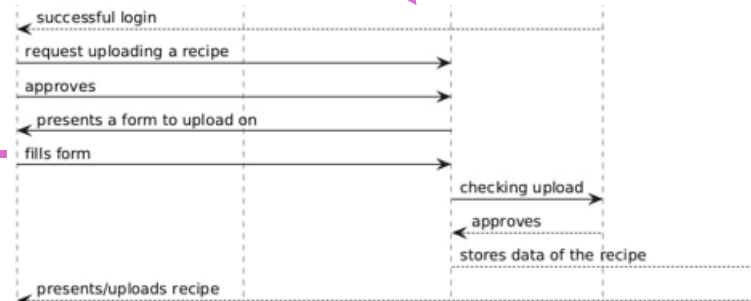
create(): void

edit(): void

delete(): void

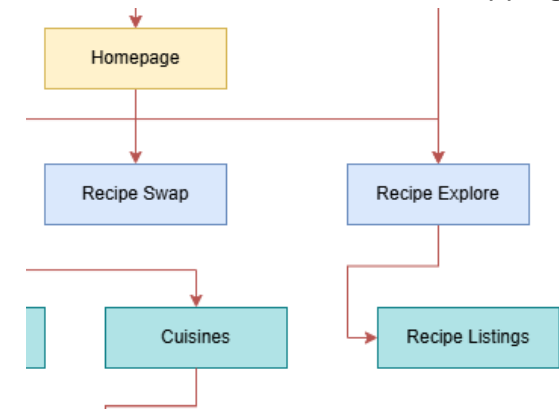
searchRecipe(): List<Recipe>

Class diagram representation



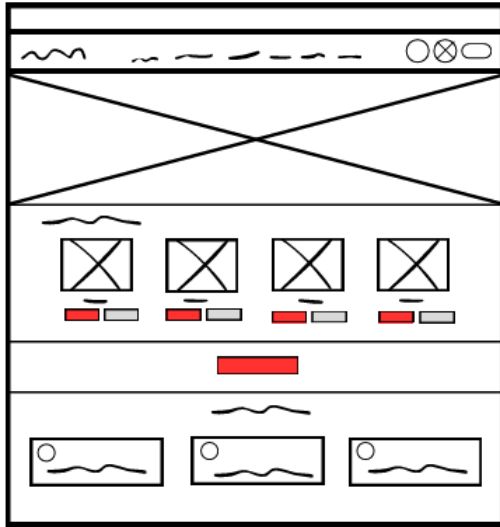
Sequence diagram incorporation

Structure mapping

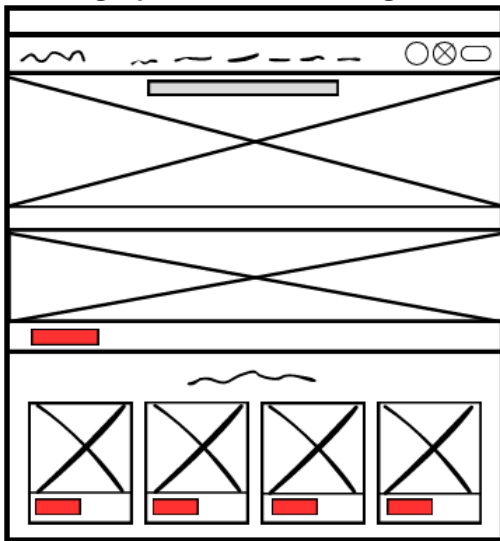


“As a [user] I want to [browse recipes] with a filtered approach.”

>>Homepage view



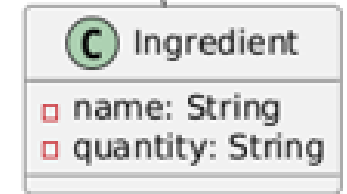
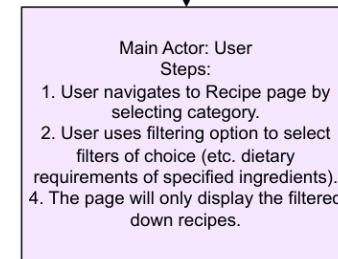
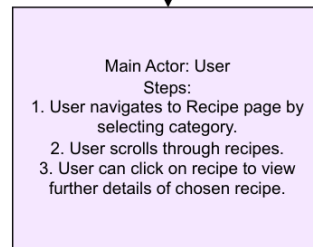
>>Category view with filtering/search



- Users select a category.
- Users can further filter options via search bar or selection.

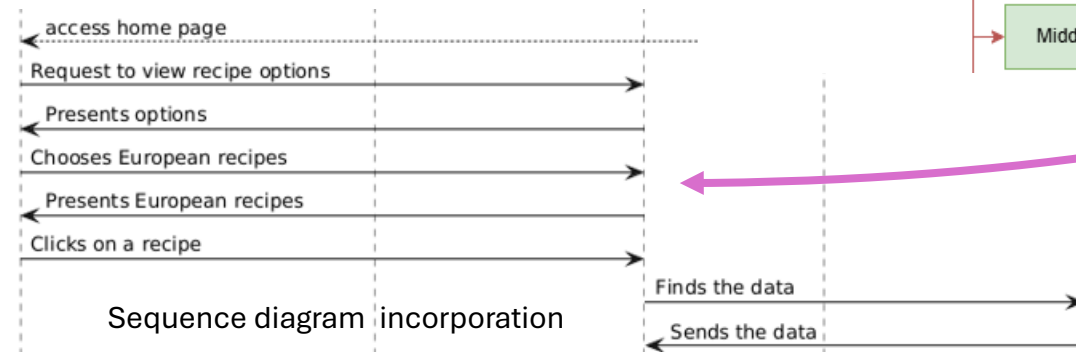
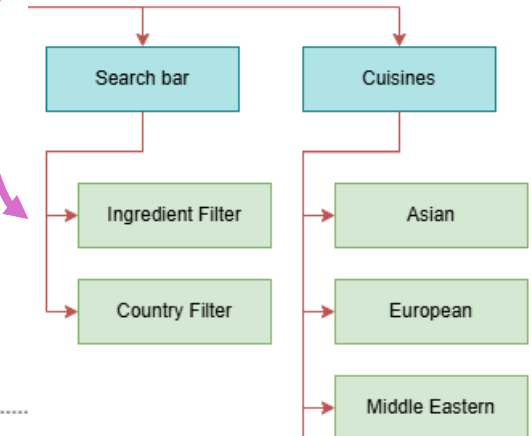


Use case scenarios



Class diagram representation

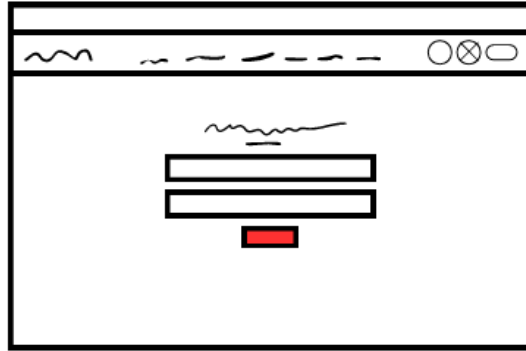
Structure mapping



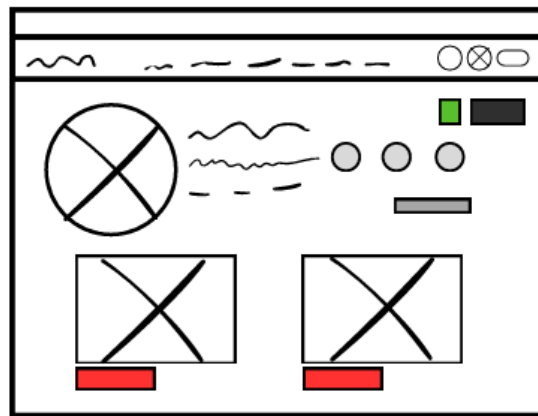
Sequence diagram incorporation

“As a [user] I want to [view my profile] where I can [edit my details and posts].”

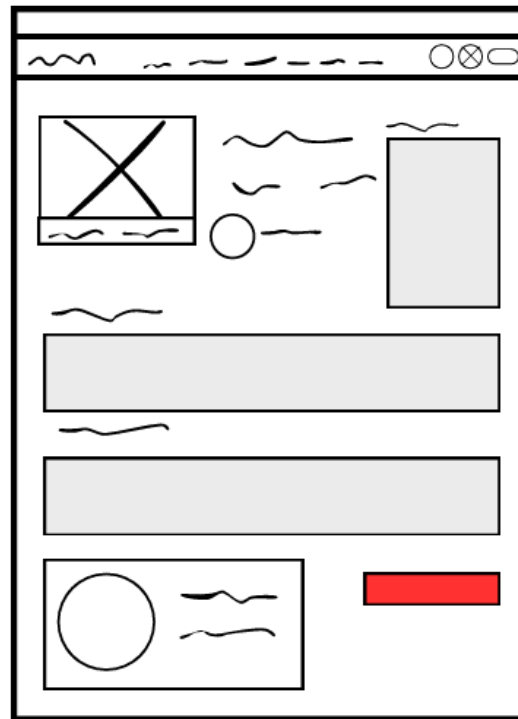
>>Login page



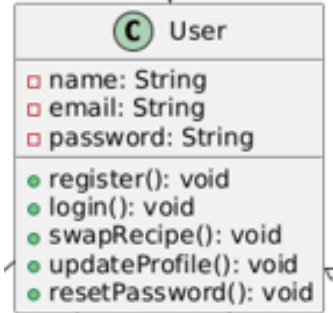
>>Profile view



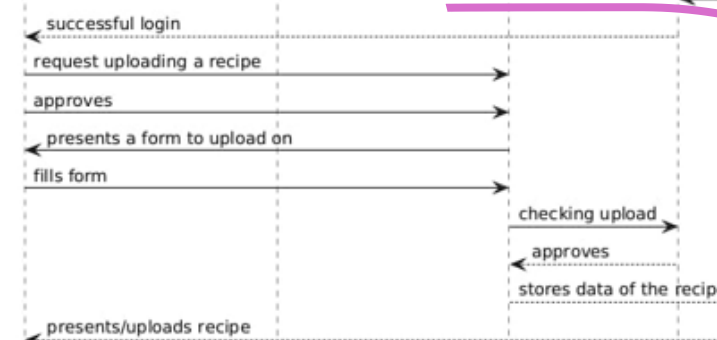
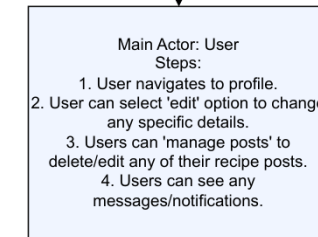
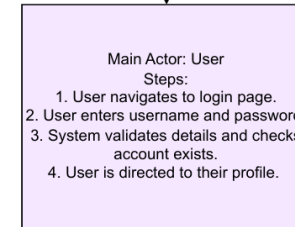
>>Recipe post page



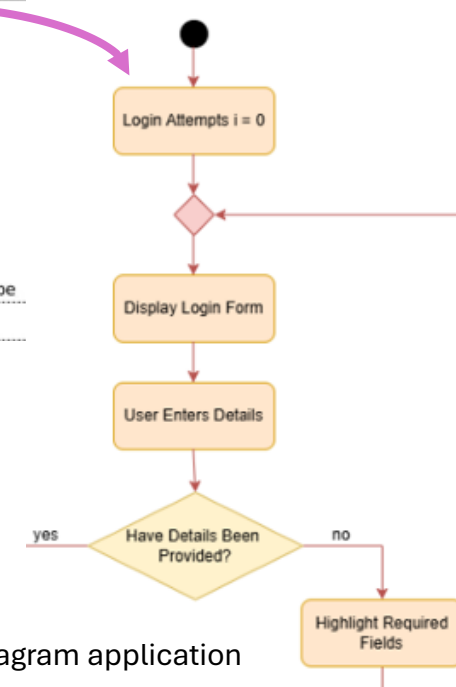
Class diagram representation



Use case scenarios



Sequence diagram incorporation

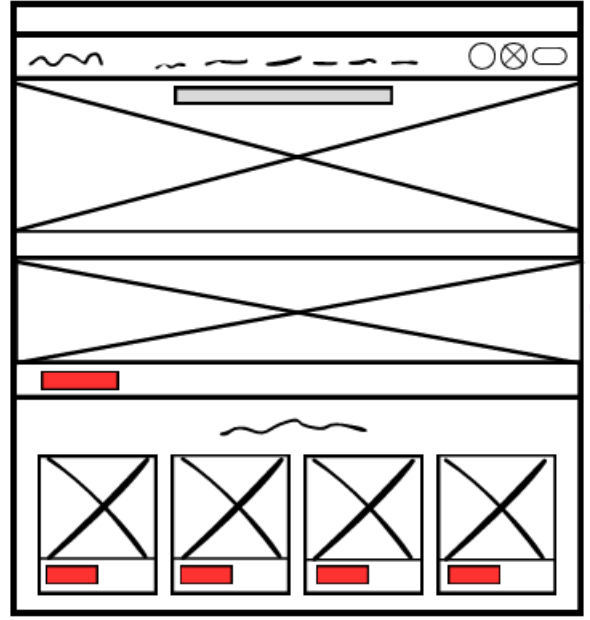


Activity diagram application

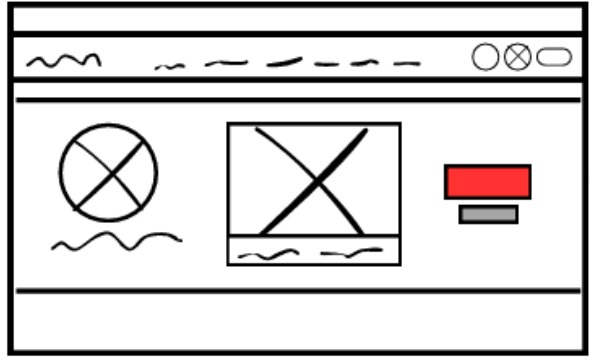
- Users' login with their existing details or they can create an account.
- They navigate to their profile page to view current posts.
- They select a post they want to edit.
- All filled in sections are available for editing.
- They republish when done and can save and exit.

“As a [user] I want to [select a recipe] so that I can [swap my own with it].”

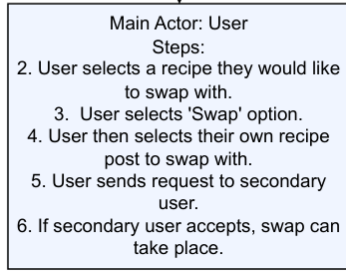
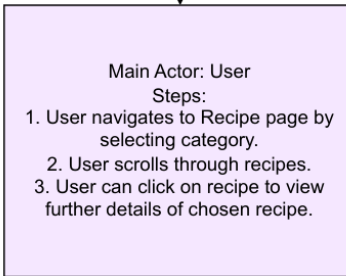
>>Recipe browsing



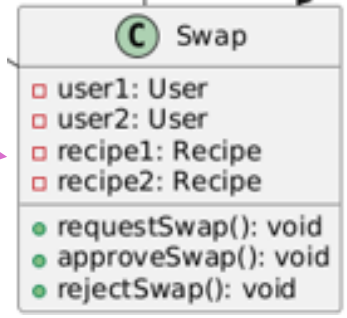
>>Swap requesting



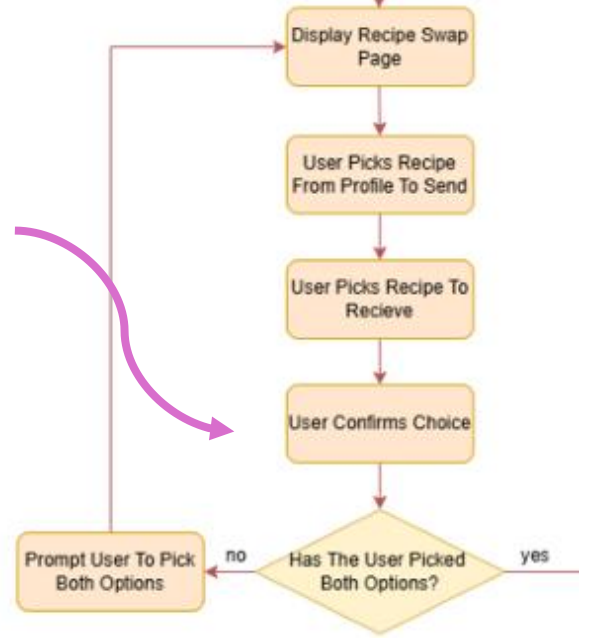
Use case scenarios



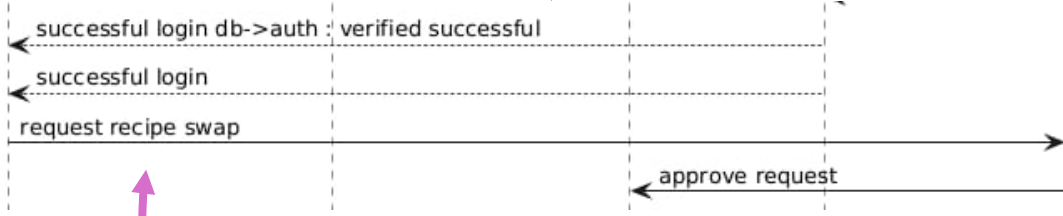
Class diagram representation



Activity diagram application



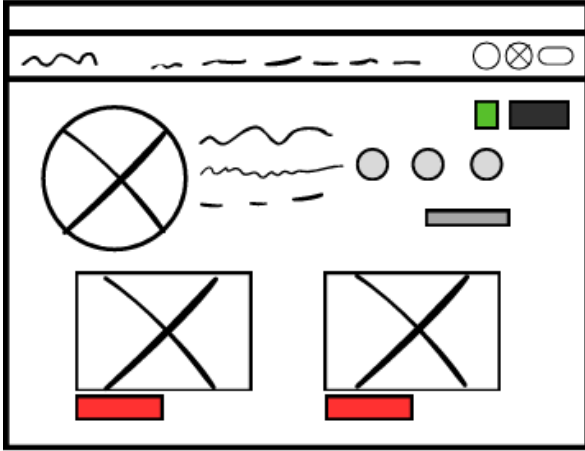
Sequence diagram incorporation



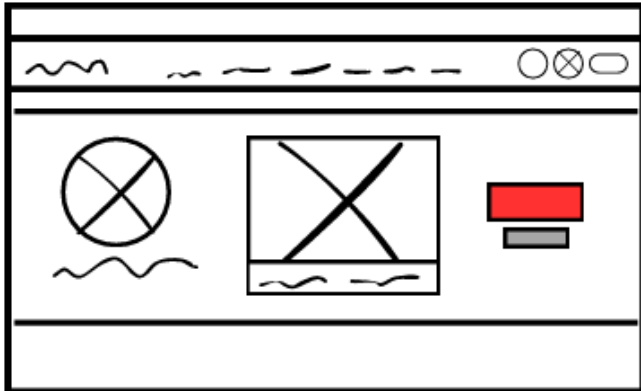
- Users browse through recipes.
- They select the one they would like to swap with.
- They pick their own post that they will be exchanging.
- They send the request.

“As a [user] I want to [leave reviews] after trying out a recipe swap.”

>>Profile view



>>Review recipe swap



Manage Profile

Leave Review

Use case scenarios

Main Actor: User
Steps:

1. User navigates to profile.
2. User can select 'edit' option to change any specific details.
3. Users can 'manage posts' to delete/edit any of their recipe posts.
4. Users can see any messages/notifications.

Main Actor: User
Steps:

1. After successful swap completed, user can select 'leave a review' under swapped recipes post.
2. Pop up allows user to input feedback text.
3. User sends review and it is available on secondary users profile/post.



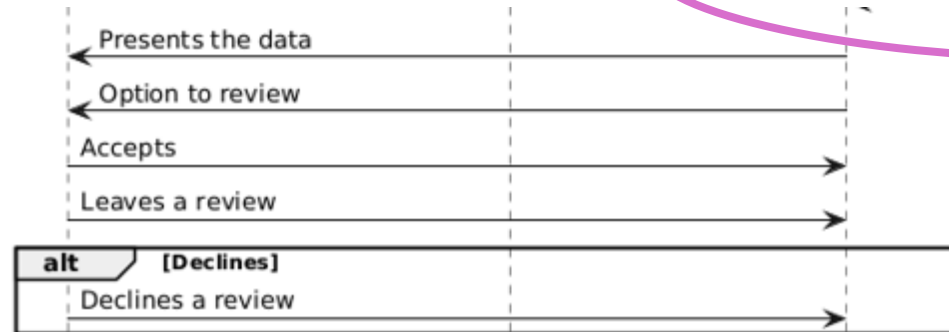
Class diagram representation

Structure mapping

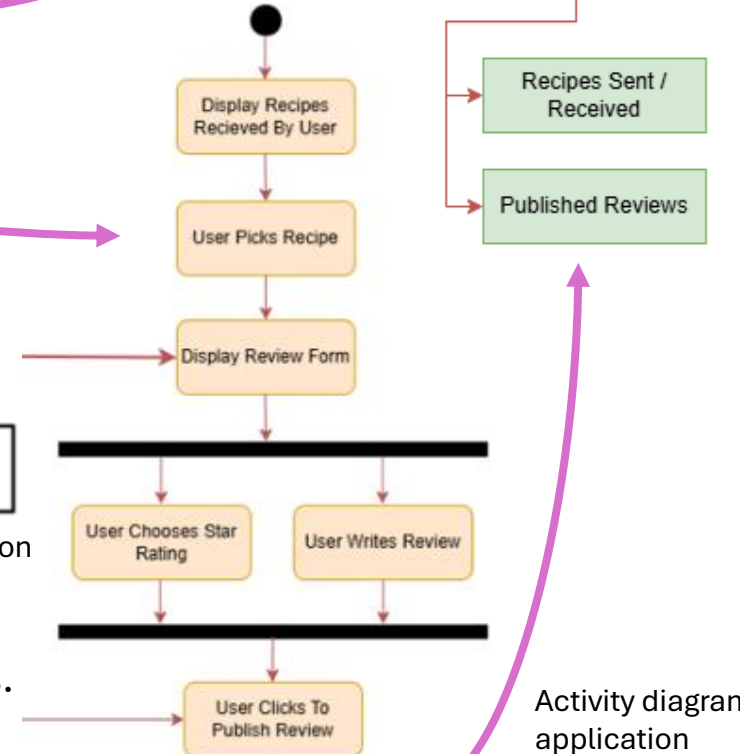
User Profile

Recipes Sent / Received

Published Reviews



Sequence diagram incorporation

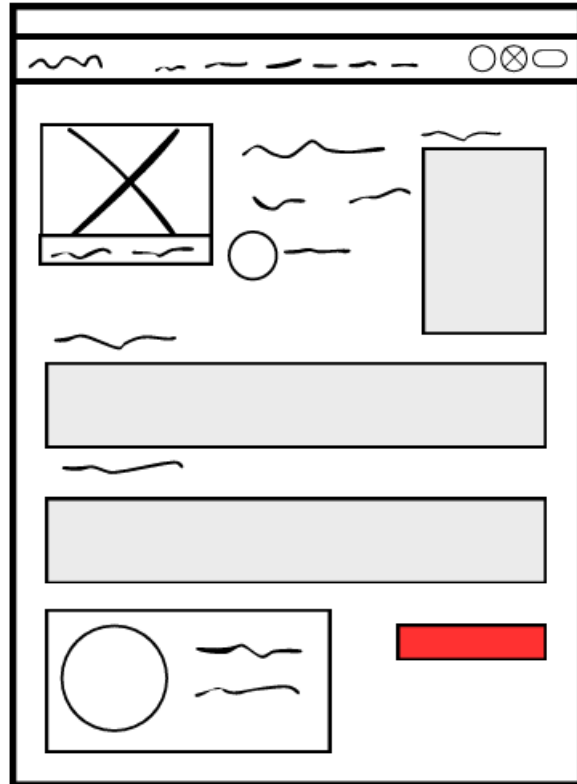


Activity diagram application

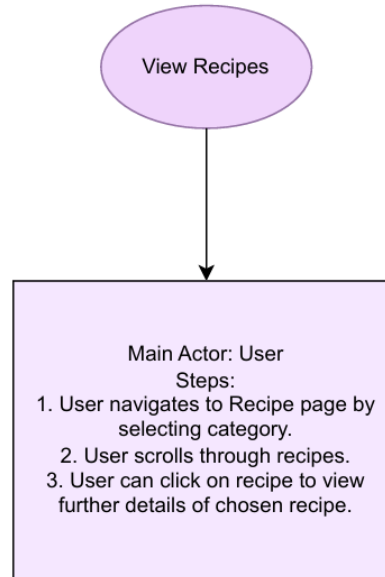
- Users go to profile to view completed recipe swaps.
- Users select the recipe's review option.
- Users fill in review and publish.

“As a [user] I want to [easily navigate the instructions of a recipe] with titled sections.”

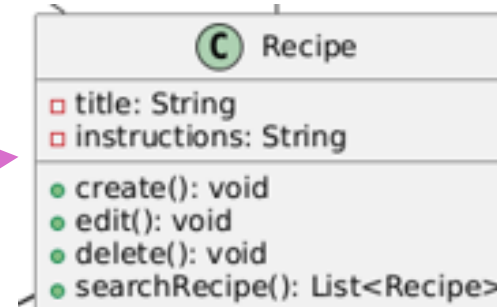
>>Recipe post



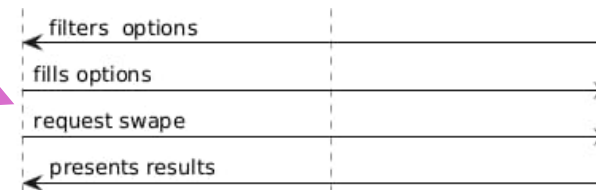
Use case scenario



Class diagram representation



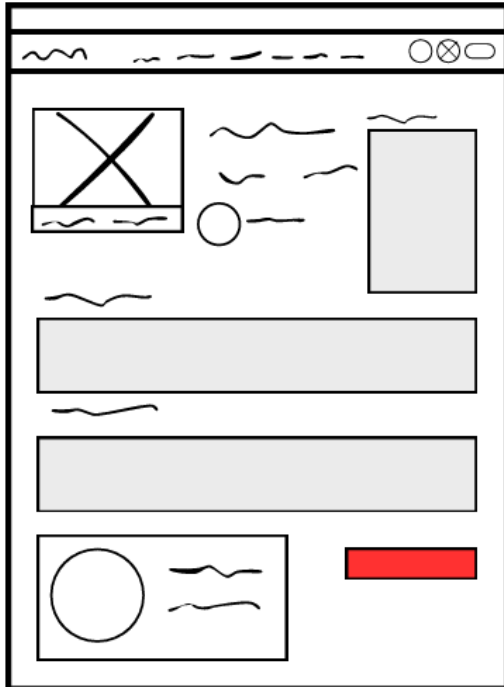
Sequence diagram incorporation



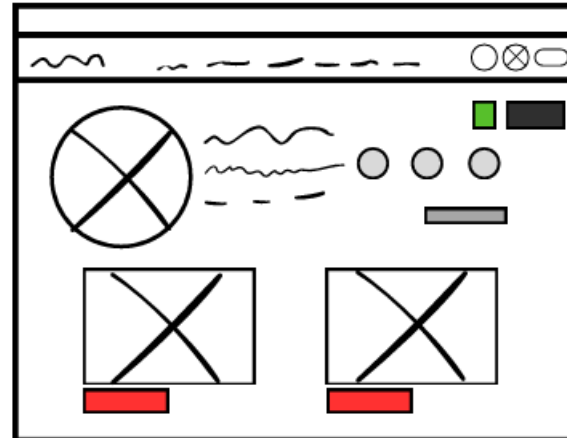
- Users can click on a recipe.
- They will be directed to the recipe’s full post which has all details in specified sections.

“As a [user] I want to [view other’s profiles] so that I can [network with them and follow more of their recipes].”

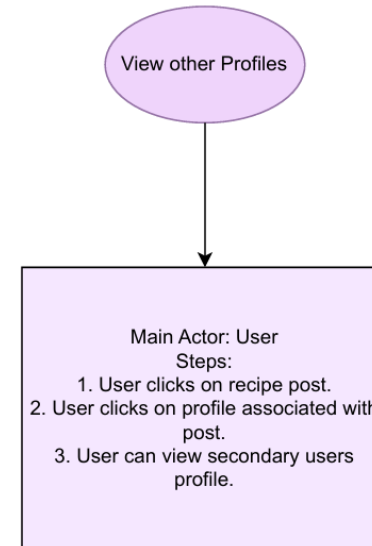
>>Recipe post, publisher below



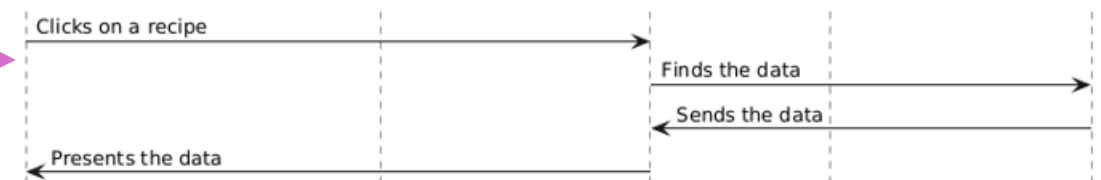
>>Publisher profile



Use case scenario

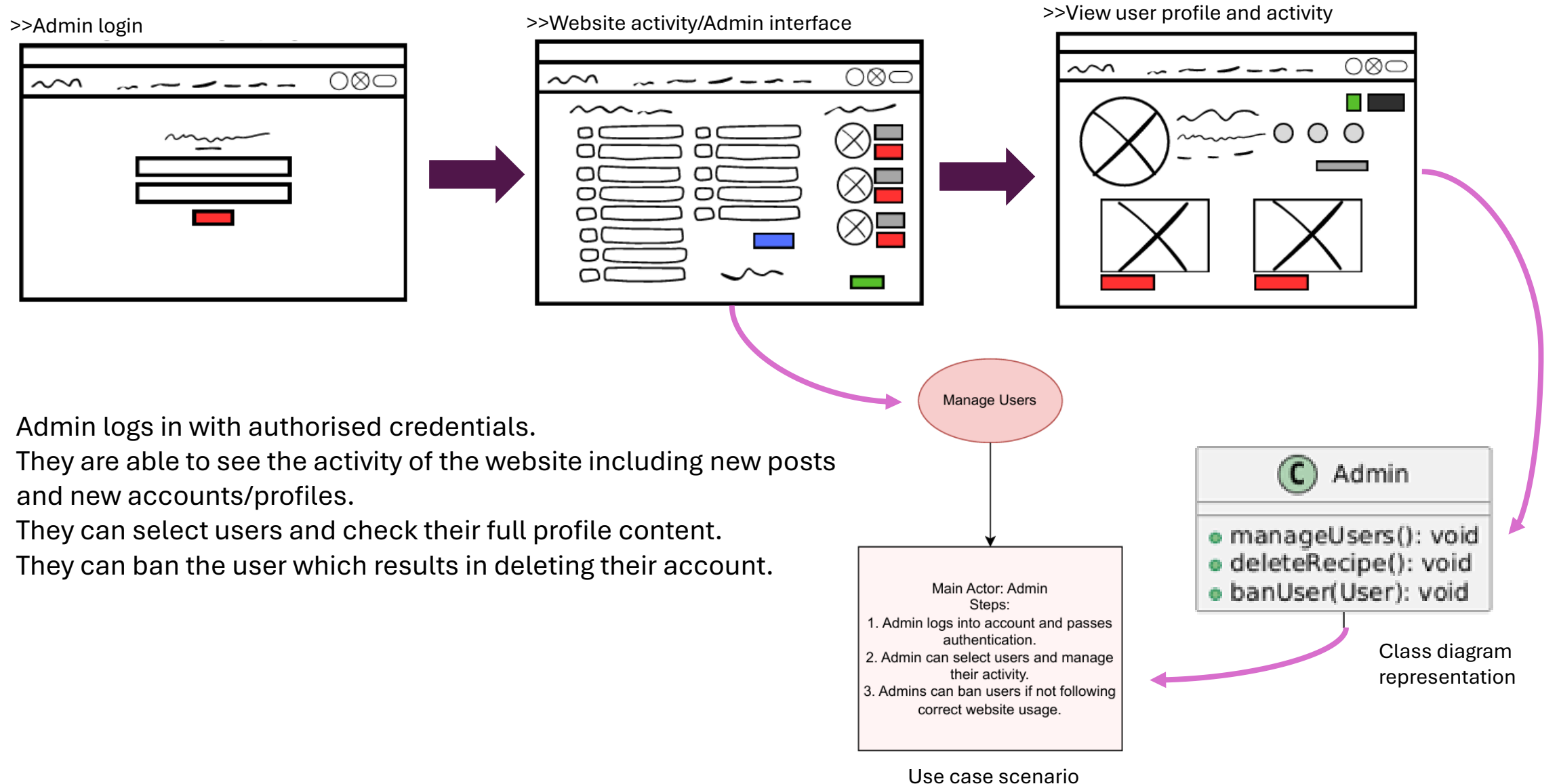


- Users select a recipe and view full details.
- They scroll down and can see the ‘publisher’ of the recipe.
- They can click on their profile.
- This redirects them to the recipe publishers’ profile.
- They can view details of them, including networking options.



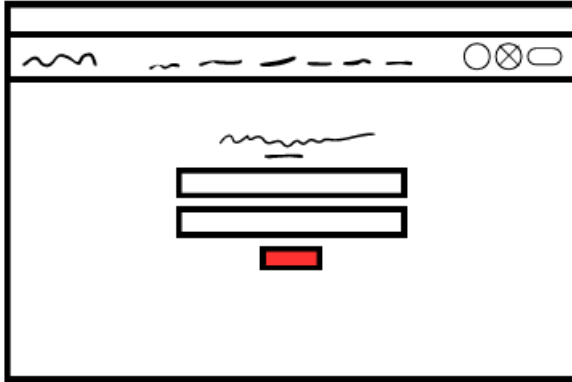
Sequence diagram incorporation

“As an [admin] I want to [manage users] so that I can [check their activity and ban any unauthorised users].”

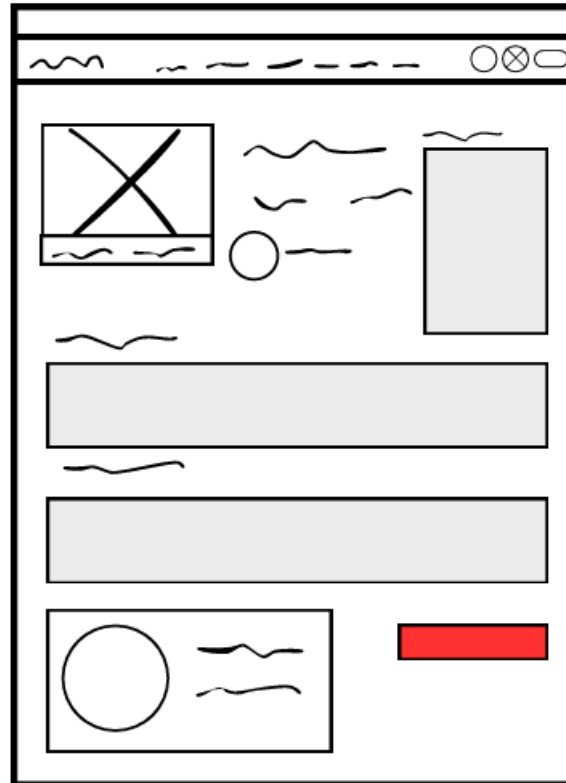


“As an [admin] I want to [delete recipe posts] if they do not meet the guidelines.”

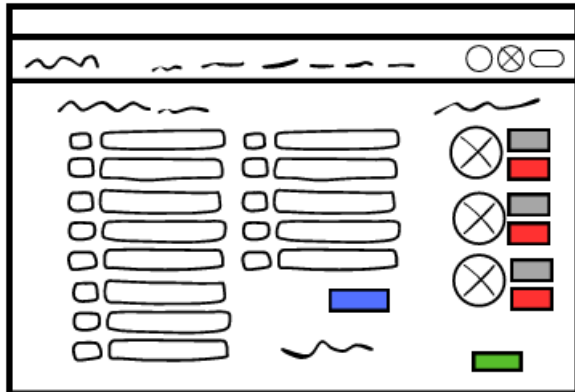
>>Admin login



>>Recipe post

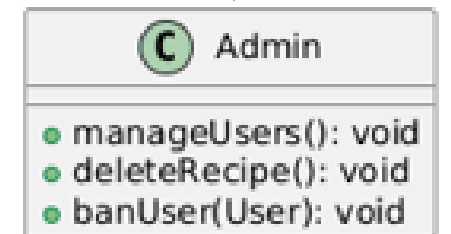
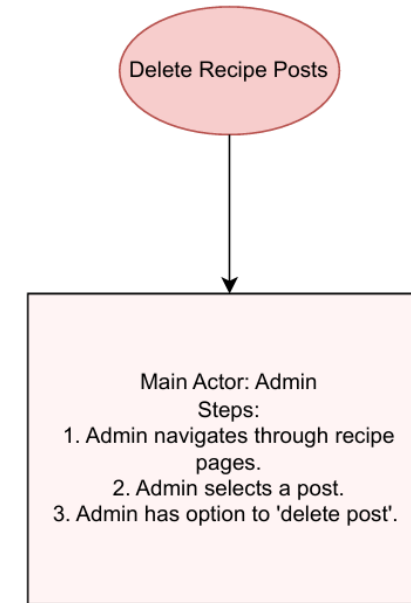


>>Website activity/Admin interface



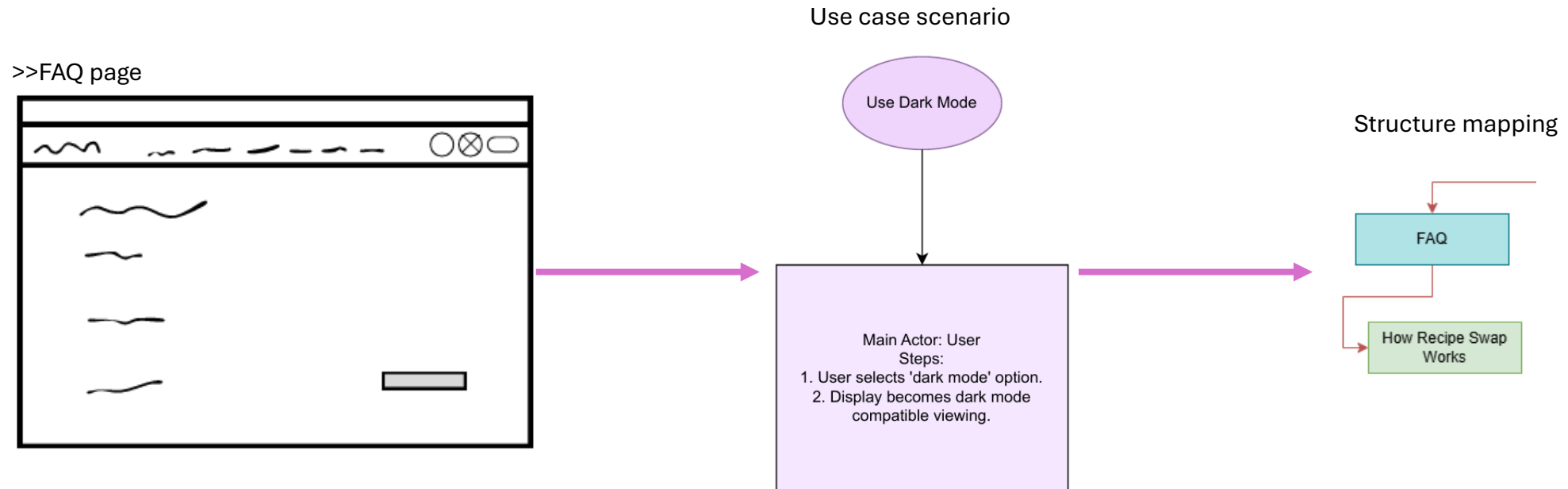
- Admin logs in with authorised credentials.
- They are able to see the activity of the website including all accounts.
- They can select users and check their posts.
- They can view the post and delete if it doesn't meet guidelines.

Use case scenario



Class diagram representation

“As a [user] I want to [use a dark-mode option] to [decrease my eye strain].”



- Users can click the toggle on the top right for dark mode.
- If they are unsure, they can go to FAQs.
- Instructions will be found in the FAQs.