Hungerger  
Brief Descriptions of All the Use-Cases

| **REVISIONS** | | | |
| --- | --- | --- | --- |
| **Rev. No** | **Description** | **Date** | **Person** |
| 0.1 | Creating the document | 04.11.2023 | Aslı |
| 0.2 | “Edit Profile” was edited. | 11.11.2023 | Aslı |
| 0.3 | “Delete Profile” was changed to “Delete Account”.  Design details were omitted for all use cases between 1 and 12. | 23.11.2023 | Aslı |
| 0.4 | The “handle” was changed to “username. | 25.11.2023 | Aslı |
| 0.5 | Use cases have been corrected to appear more UI-free.  The “Visitor” was added to the “Create Account” use case. | 15.12.2023 | Aslı |

1. **Create Account:**

Anyone who wants to access the application content shall have an account on the system. Therefore, when a Visitor accesses the application interface, the system displays a page where the Visitor can enter the account information (See Supplementary Requirement WC-1 for account information). The Visitor enters and then submits the account information. The system validates account information and then saves it to the database. The system displays a message regarding the identity verification step (See Supplementary Requirement WC-2 for identity verification). The Visitor verifies their identity. The system shows a message regarding a successful operation. The Visitor becomes a User of the system.

1. **Edit Profile:**

A User who wants to edit their profile shall have an account on the system and be logged in. The User requests to edit their profile. The system displays a page where the User can edit the profile information (See Special Requirement WC-3 for profile information). The User enters profile information. The User submits the changes. The system validates the profile information. The system asks the User to verify their identity. The User validates their identity. The system updates its database and shows a message regarding a successful operation.

1. **Delete Account:**

A User who wants to delete their account shall have an account on the system and be logged in. The User requests to delete their account. The system asks the User to verify their account ownership. The User verifies their ownership (See Supplementary Requirement WC-4 for ownership verification for account removal). The system validates the ownership. The system requests a confirmation from the User for the account removal. The User confirms the request. The system removes the account and shows a message regarding a successful operation. The system directs the User to the visitor interface.

1. **Create Recipe:**

A User shall have an account on the system and be logged in. The User wants to create a recipe. The User enters recipe information (See Supplementary Requirement WC-5 for recipe information). The system validates the recipe information and fetches the current prices of the ingredients from the marketplace, calculates and displays the total cost of the recipe. The User approves the content and submits the recipe. The system reflects the changes to the application's user interface.

1. **Edit Recipe:**

A User who wants to edit their recipe shall have an account on the system, be logged in, and have at least one recipe on their account. The User selects a recipe and starts to edit its information (See Supplementary Requirement WC-5 for recipe information). The system validates the recipe information and fetches the current prices of the ingredients from the marketplace, calculates and displays the total cost of the recipe. The User approves the content and submits the updated recipe. The system reflects the changes to the application's user interface.

1. **Delete Recipe:**

A User who wants to delete their recipe shall have an account on the system, be logged in, and have at least one recipe on their account. The User selects a recipe and requests to delete it. The system asks for a confirmation about the deletion request. The User confirms their request. The system deletes the recipe and reflects the changes to the application's User interface.

1. **Search Recipe:**

A User who wants to search for a recipe shall have an account on the system and be logged in. The User requests to search for a recipe. The system presents filtering options (See Supplementary Requirement WC-7 for filtering options). The User applies filtering options and submits their request. The system displays recipes that match the filters.

1. **Rate Recipe:**

A User who wants to rate a recipe shall have an account on the system and be logged in. The system should contain at least one recipe. The User requests to rate a recipe. The system displays the rating options (See Supplementary Requirement WC-8 for rating options). The User gives rates to the recipe (See Supplementary Requirement WC-9 for the rating range). The system takes the average of these ratings due to their categories and shows the average ratings on the posted recipe.

1. **Comment Recipe:**

A User who wants to comment on a recipe shall have an account on the system and be logged in. The system shall contain at least one recipe. The User requests to comment on a recipe. The system displays an area where the User can enter their comment. The User comments on a recipe. The System validates and saves the comment, then shows it on the recipe post.

1. **Follow User:**

A User who wants to follow another User shall have an account on the system and be logged in. The system shall contain at least two Users. The User requests to follow another User. The system notifies the corresponding User. If this User accepts this request, the system updates the Followers of the target User and the Followings of the initial User. The System notifies the initial User about the request's status if it is accepted.

1. **Report Misuse:**

A User who wants to Report Misuse shall have an account on the system and be logged in. The system shall contain at least one entity to report (See Supplementary Requirement WC-10 for the entities). The User reports the entity. The System saves this request and notifies the Admin. After the examination, the Admin takes action (See Supplementary Requirement WC-11 for the supported actions). The system notifies the User about the result of their request.

1. **Update to Verified Account:**

A User who wants to become an Influencer should have an account on the system and reach a certain number of followers (See Supplementary Requirement WC-12 for the certain number of followers). While logged in, the User requests the Admin to update their account. The system checks the follower number of the User and directs the request to the Admin if it is sufficient enough. The Admin updates the User account to a verified one by giving the User a “Blue Tick”, after checking their account. The system shows the “Blue Tick” on the User’s Profile Page.

Supplementary Requirements:

[SpReq: WC-1]: The account information contains username, email, password, and password confirmation.

[SpReq: WC-2]: The identity verification is done by the system sending an email to the Visitor where the Visitor can follow the steps regarding verification.

[SpReq: WC-3]: The profile information contains name, email address, username, password, avatar, and biography.

[SpReq: WC-4]: The ownership verification for the account removal is done via the User entering their password.

[SpReq: WC-5]: The recipe information contains the recipe name, description, image, dietary type (See Supplementary Requirement WC-6 for existing dietary types), ingredients, and the amounts of the ingredients.

[SpReq: WC-6]: The existing dietary types are regular, vegan, vegetarian, gluten-free, and keto.

[SpReq: WC-7]: The filtering options are name, dietary type (See Supplementary Requirement WC-6 for dietary types), cost range, rate, and ingredients.

[SpReq: WC-8]: The rating options are taste and preparation of ease.

[SpReq: WC-9]: The rating range is between 1-5 stars.

[SpReq: WC-10]: The entity might be another User, post, or comment.

[SpReq: WC-11]: The supported action might be deleting the User, the post, the comment, or rejecting the initial request.

[SpReq: WC-12]: The follower number should be at least 10,000.