Team logistics

Team name
- 2Cool 2Cook

Team member names
- Ruisi Jian
- Yueting Liao
- Yuchen Wang
- Yifan Wang
- Xiaoyu Zhang

Weekly team meeting schedule in EDT
- Mon/Thu 9:00 AM - 10:50 AM

A summary of how you will run each meeting and where minutes will be recorded and stored (and searchable)
- Summarize and evaluate what we did in the last meeting and conclude our inadequacy.
- Each member will be reporting their progress of tasks and potential problems from the last meeting
- Discuss the progress made so far and evaluate for potential improvements.
- Provide some constructive thinking and set a list of tasks for the next meeting.
- Assign appropriate works for each team member.
- Use Google Doc to record the minutes from the meeting (One Doc for all meetings)
- https://docs.google.com/document/d/1mJW8UWMUnrlh8FBEFf_GY9MLaYU6fGM3n-BjViskeRc/edit?usp=sharing

Nooks room we typically meet in
- K

Project's URL
- https://github.com/YuetingLiao/SDD-Team.git

Project summary

An executive summary of your project that (in one paragraph) describes your project and why your project is important

This project presents a web-based solution to help a wide population of cooking greenhands to manage their daily menu and minimize their cooking time. The idea of the project came from the Covid-19 epidemic, a time when restaurants and diners are closed or have a limited suppliance. A considerable number of beginners inevitably have to cook themselves. They prefer to have some tool to help them plan out a weekly menu and rearrange the procedures to maximize concurrency of steps so that the cooking time could be minimized. In order to assist users to cook in a more convenient and efficient way, this project will be developing a website that allows users to search for a recipe, manage recipes into menus, and generate optimized cooking procedures to minimize cooking time.

A list of project stakeholders and their specific interests in the project

- Cooking novice: make their first step easy and build their confidence in cooking
- People who have little time to cook: plan cooking procedures in maximized concurrency to help prepare a good meal in a short time
- People who are indecisive: provide related recipes based on interested keyword input
- People who are plan-driven: record the daily menu ahead of time
- Recipe authors who create recipes and share recipes
- Chefs who want to learn new recipes: expand their professional skills

A brief statement on the feasibility of your project both in general and within the 10-12 weeks you will work on this project this semester

        The feasibility of the project is examined through technological considerations, the existing marketplace, and schedule & time.

Technological considerations:

- Insufficient experience in HTML and javascript
- Database management
- Recipe collection
- Remote Server
- Service may need to be on several platforms

Existing marketplace:

- Customer Stickiness
- Competition among similar service in the market
- Similar service on different platforms in the market

Schedule and timeline:

- Collaboration across time zone
- Obligations besides this project
- Work may not spread out evenly throughout the 12 weeks

The scope of your project, which should indicate what you are not planning to build (also include other systems that your system would/could interact with)

- Service on a smartphone platform
- A database of recipes from all countries/regions
- Chatting or comments for recipes on the website
- Deployment of the service on a remote server
- Manual check of recipe upload (assuming all uploads are legal)

A brief statement on project risks that identifies at least three potential risks of your project

- Risk of dietary allergy
- Risk of putting people in danger if they might forget to turn off the stove when they failed to handle concurrency
- Users might upload inappropriate data into the recipe database
- Copyright risks of the recipes

Behavioral requirements

Using a top-down approach, include at least 12 fully dressed use cases for your project. Start each use case on a new page and remember to avoid user interface details.

Use Case Title: Search for a Recipe

Primary Actor: User

Stakeholders and Interests: User would like to find a recipe they want to cook

Preconditions: N/A

Main Success Scenario:
1. User navigates to the search bar
2. User types in the keywords of a recipe
3. User get a list of potential recipes through keywords; if displaying nothing, return to step 2
4. User chooses one of the recipes from the displayed list
5. User gets a detailed view of the chosen recipe

Extensions:
2a. If the user leaves the search bar fields blank, then show blank.
2b. If the user types in unidentified keywords, then show blank.

Minimal Guarantees: N/A

Success Guarantees: Users able to get all recipes

Use case references by which you reference another use case by underlining the given task: N/A

Use Case Title: User Creates a Login Account

Primary Actor: User

Stakeholders and Interests: User would like to upload their favorite recipes

Preconditions: N/A

Main Success Scenario:
1. User navigates to "create account" button
2. Fill in the personal account information
   a. Username (required)
   b. Date of Birth
   c. Password (required)
   d. Confirmed password (required)
   e. Email address (required)
   f. Allergies
   g. Dietary preferences
3. User automatically logs into the new account

Extensions:
2a. If User leaves a required blank or empty, then shows errors
2b. If the username has already existed, then shows errors
2c. If the confirmed password does not match the password, then shows errors

Minimal Guarantees: N/A

Success Guarantees: Users own a new account

Use case references by which you reference another use case by underlining the given task:
N/A

Use Case Title: User Uploads a New Recipe

Primary Actor: User

Stakeholders and Interests: User uploads a new recipe to the database. Everyone can search and see the new recipe

Preconditions: User must be authenticated and logged in

Main Success Scenario:
4. User clicks "upload a new recipe" button
5. User enters the new recipe name (required)
6. User enters all the necessary ingredients (required)
7. User enters all steps of recipes (required)
8. User uploads an image of the recipe
9. User uploads the new recipe to the database successfully.

Extensions:
2a. If the user leaves a blank, then shows errors
3a. If the user leaves a blank, then shows errors
4a. If the user leaves a blank, then shows errors

Minimal Guarantees: User uploads the recipe to the database

Success Guarantees: User's recipe is valid after verified. Everyone can see the new recipe

Use case references by which you reference another use case by underlining the given task
Staff <u>verifies that new recipe is valid</u>

Use Case Title: User Creates a Daily Menu

Primary Actor: User

Stakeholders and Interests: User would like to plan out what they cook for meals on a daily basis

Preconditions: User must be authenticated and logged in

Main Success Scenario:
     10. User navigates to today's menu
     11. User select which meal (breakfast, lunch, dinner) to add
     12. User search a recipe for that meal
     13. User add recipes to that meal on the daily menu
     14. User clicks save button to finish create a menu

Extensions:
3a. If the user couldn't get a recipe from add, user can choose to upload a new recipe

Minimal Guarantees: N/A

Success Guarantees: A new daily menu is created

Use case references by which you reference another use case by underlining the given task
The users search for a recipe
The users upload a new recipe (optional)

Use Case Title: Apply Filters by Allergy

Primary Actor: User

Stakeholders and Interests: User would like eliminate recipes based their dietary allergy

Preconditions: Log into the account; Enter recipe information in the search bar

Main Success Scenario:
15. User login to the account
16. User updates the their dietary allergy in their profile
17. User navigates to the search bar
18. User enters the keywords for a recipe
19. Allergy filter will be applied to the recipes returned
20. Recipes with the keywords will be displayed if it does not contain allergy ingredient

Extensions:
2a. If users already have their dietary allergy entered in their profile, skip this step.
4a. If the user leaves the search bar fields blank, then show blank.
4b. If the user types in unidentified keywords, then show blank.

Minimal Guarantees: Normal search result will be displayed if the filter failed in an unexpected manner

Success Guarantees: A list of recipes with no allergic ingredients will be displayed after the filter.

Use case references by which you reference another use case by underlining the given task
The users create an account and log in
The users search for a recipe

Use Case Title: User feedback

Primary Actor: User

Stakeholders and Interests: User would like to give some feedback on their experience with this website

Preconditions: Log into the account

Main Success Scenario:
1. User login to the account
2. User navigates to the feedback page
3. User check from a list of frequent issues (required)
4. User types in more detailed comments in textbox

Extensions:
3a. If the user did not select any of the check from the list, the submit button is disabled

Minimal Guarantees: N/A

Success Guarantees: A success submission page will be displayed to the user

Use case references by which you reference another use case by underlining the given task
The users <u>create an account and log in</u>

Use Case Title: Favourite Recipes

Primary Actor: User

Stakeholders and Interests: User would like to save recipes that they like for future reference

Preconditions: Log into the account

Main Success Scenario:
1. User login to the account
2. User navigates to the search bar
3. User enters the keywords for a recipe
4. Recipes with the keywords will be displayed
5. User would select the "add to favourite" button

Extensions:
3a. If the user leaves the search bar fields blank, then show blank.
3b. If the user types in unidentified keywords, then show blank.
5a. If the recipe is already added to favourite, click again to remove from favourite

Minimal Guarantees: N/A

Success Guarantees: Recipe saved to their list of favourite recipes

Use case references by which you reference another use case by underlining the given task
The users create an account and log in
The users search for a recipe

User Case: Verify Validity of Uploaded Recipes

Primary Actor: Staff

Stakeholders and Interests: Staff would like to make sure every recipe waiting to be added to the database is feasible

Preconditions: Staff log into the account

Main Success Scenario:
1. Read through the list of recipes uploaded by user that are waiting to be added to database
2. Update the recipe into the database if this recipe looks reasonable
3. Send notification to the users that their recipe successfully went through

Extensions:
2a. If the recipe looks bad, staff would notify users that their recipe is invalid

Minimal Guarantees: N/A

Success Guarantees: The recipe will be either passed verification or not.

Use case references by which you reference another use case by underlining the given task
N/A

User Case: Rate the Recipe

Primary Actor: User

Stakeholders and Interests: User would like to rate each recipe

Preconditions: Log into the account

Main Success Scenario:
1. User login to the account
2. User navigates to the search bar
3. User enters the keywords for a recipe
4. User navigates to the recipe
5. User click rate the recipe button and give a score

Extensions:
3a. If the user leaves the search bar fields blank, then show blank.
3b. If the user types in unidentified keywords, then show blank.
5a. If user has already rated, rating button disabled

Minimal Guarantees: N/A

Success Guarantees: Recipe score is updated with the rating input

Use case references by which you reference another use case by underlining the given task
The users create an account and log in
The users search for a recipe

User Case: Sort the Search Result by Rating

Primary Actor: User

Stakeholders and Interests: User would like to see recipes with higher rating up front

Preconditions:

Main Success Scenario:
1. User navigates to the search bar
2. User enters the keywords for a recipe
3. Recipes with the keywords will be displayed in order of sorting

Extensions:
2a. If the user leaves the search bar fields blank, then show blank.
2b. If the user types in unidentified keywords, then show blank.

Minimal Guarantees: N/A

Success Guarantees: Recipes will automatically be ordered based on the rating from highest to lowest

Use case references by which you reference another use case by underlining the given task
The users <u>search for a recipe</u>
The users <u>rate the recipe</u>

User Case: Ranking List for Top 10 Recipes for Each Week

Primary Actor: User

Stakeholders and Interests: User would like to see the best rating recipes

Preconditions: N/A

Main Success Scenario:
1. User navigates to the ranking page
2. Ranking is updated based on current recipe rating at the beginning of each week

Extensions: N/A

Minimal Guarantees: N/A

Success Guarantees: Display top 10 recipes weekly.

Use case references by which you reference another use case by underlining the given task
The users <u>rate the recipe</u>

User Case: Search for Nearby Farmers Market for to buy Ingredients

Primary Actor: User

Stakeholders and Interests: User would like to see nearby farmer market to buy ingredients

Preconditions: Log into the account

Main Success Scenario:
1. User login to the account
2. User navigates to search farmers market page
3. User enters information in the text box
4. The farmers market result will be displayed

Extensions:
3a. If the user leaves all text box fields blank, then show all.
3b. If the user types in unidentified keywords, then show blank.

Minimal Guarantees: N/A

Success Guarantees: All the farmers market will be displayed depend on user input

Use case references by which you reference another use case by underlining the given task
The users create an account and log in