

# Cooking Papa

Contributors: JD Heckenliable, Ken Endo(TL), Michael Hall, Ryan Shereda

## Summary:

Recipe sharing system that dynamically expands with user interaction. Users can create their own recipes, search desired recipes, and make their own unique cookbook by adding these recipes into a bookmark. Dynamically how? For example, users can select any ingredient from a drop down list for their recipe, if an ingredient does not exist, the user can add a new ingredient to the drop down list so future users can also use it as needed. The main idea for this website is to create a system that dynamically expands with user demands.

## User Stories/Business Requirements:

As a *registered user*, I would like to be able to:

- Create a recipe to share with other people
- Edit or delete a recipe that I created
- Search through existing recipes
- See recipe details
- Bookmark recipes (mini cookbook of favorites or whatever)
- Leave a rating and an optional review on a recipe

As a *Unregistered user*, I would like to be able to:

- Search through existing recipes (like a read only user)
- Create a personal account to access special features of the website

Optional functions:

- Mark recipes that I have tried
- Post images of my recipes
- Post images of other recipes
- Look at images of other recipes
- Follow another user to see whenever they have new recipes
- Be able to navigate to another user's recipes based on one that I found
- Search recipes by ingredients
- Nutritional information under recipe detail

Api Considered:

- Youtube data api
- Serpi api (google image search)
- Spoonacular api (recipe, food, nutrition api)

# Business Requirements

1. **Ability to register and login as a unique user.**
  - a. Unique functionalities for *registered* users.
  - b. Limited functionalities for *unregistered* users.
2. **Ability to create a new recipe.**
  - a. *Registered* users are able to create a recipe with name, origin, cook time, ingredients, and instructions.
  - b. *Registered* user selects any ingredient from a drop down list for their recipe, if an ingredient does not exist, the user can add a new ingredient to the drop down list so future users can also use it as needed.
  - c. Recipes should be persistent.
3. **Ability to edit/delete recipes the user created.**
  - a. *Registered* users may edit/delete a recipe he/she created by adding or removing ingredients/instructions.
4. **Ability to search recipes.**
  - a. *Registered/Unregistered* users may search recipes by name, origin, cook time, ingredients, or ratings.
5. **Ability to view the recipe detail.**
  - a. *Registered/Unregistered* users may click on searched recipes to see detailed instructions of that recipe.
  - b. *Registered/Unregistered* users may see the overall ratings and comments at the bottom of the recipe detail.
  - c. *Registered* users may add a rating and an optional comment to the recipe.
  - d. *Registered* users may have the options to add a recipe to their cookbook.
6. **Ability to access the cookbook.**
  - a. *Registered* users may access its own cookbook to see all the recipes he/she 'liked/bookmarked'.
  - b. *Registered* users may remove a recipe from the cookbook.
  - c. *Registered* users may see a list of all the recipes they've created.

# Use Cases

|                |  |
|----------------|--|
| Use Case       | Create account   |
| Actor          | Unregistered user  |
| Basic Flow     | The user selects the “create account” button. A form opens, with multiple empty blanks(email, username, password) for the user to populate with their information. When submitted, the account details are checked against the database of existing accounts for duplicates. If all necessary values are appropriate, the account is created and the user is prompted to log in; else, the user is informed of the conflicted fields and left to try different values. |
| Preconditions  | None.  |
| Postconditions | User has created an account, and can sign in.  |

|                |  |
|----------------|--|
| Use Case       | Creating a new recipe  |
| Actor          | Registered user  |
| Basic Flow     | The user selects the “Add new recipe” button. A form opens with areas to name the recipe, enter the total cook time, select/add ingredients, select a country of origin, add instructions and notes. When submitted, the form is checked for validity if there are issues the user is notified and asked to fix them(client side validation). If the form is valid the user is shown a finished version and asked if they would like to make any changes. If so they are taken back to the edit page and if not the form is submitted. |
| Preconditions  | The user is logged into an account.  |
| Postconditions | The new recipe will be saved in the DB and made available to all other users.  |

|                |  |
|----------------|--|
| Use Case       | Access Cookbook  |
| Actor          | Registered user  |
| Basic Flow     | User can go to a page to view a list of recipes stored into the cookbook using the current login credential. The list will display the name, origin, cook time, and the rating of each recipe. User can click an item to see its recipe details. |
| Preconditions  | User is logged into an account.  |
| Postconditions | Displays a list of bookmarked.   |

|                |   |
|----------------|---|
| Use Case       | Update/Delete created recipes   |
| Actor          | Registered user   |
| Basic Flow     | User is viewing a recipe they created, and selects either the “edit” or “delete” buttons. On clicking delete, the user is asked for confirmation of deletion. If yes, the recipe is permanently deleted from the database, and the user is returned to their home page; if no, the user is returned to the recipe page. On clicking update, the user is brought to a page similar to the creation page, where they can modify fields as needed. When submitted or cancelled, the user is returned to the recipe page. |
| Preconditions  | <ul style="list-style-type: none"> <li>• User is logged in. <ul style="list-style-type: none"> <li>○ Viewing recipe he/she created.</li> </ul> </li> </ul>  |
| Postconditions | Recipe is edited or deleted based on user input.  |

|                |   |
|----------------|---|
| Use Case       | Search existing recipes   |
| Actor          | Registered/Unregistered user  |
| Basic Flow     | The user selects the “Search recipes” button. A form opens with options to search by recipe name, country of origin, cook/prep time or any combination. |
| Preconditions  | None.   |
| Postconditions | A list of matching recipes is returned to the user  |

|                |  |
|----------------|--|
| Use Case       | View recipe detail   |
| Actor          | Registered/Unregistered user   |
| Basic Flow     | <ul style="list-style-type: none"> <li>● Registered user <ul style="list-style-type: none"> <li>○ User selects a recipe from the search results or their “cookbook” a form opens displaying the created by, ingredients, cook time, instructions/notes, rating/comment.</li> <li>○ User can add ratings and optional comments to the recipe. This will be displayed when the page is refreshed.</li> <li>○ User can select to add the recipe to their cookbook.</li> </ul> </li> <li>● Unregistered user <ul style="list-style-type: none"> <li>○ User selects a recipe from the search results or their “cookbook” a form opens displaying the created by, ingredients, cook time, instructions/notes, rating/comment.</li> </ul> </li> </ul> |
| Preconditions  | <ul style="list-style-type: none"> <li>● User is logged into an account. <ul style="list-style-type: none"> <li>○ User performs a search.</li> <li>○ User checks its own cookbook.</li> </ul> </li> <li>● User is Not Logged into an account(Guest). <ul style="list-style-type: none"> <li>○ User performs a search.</li> </ul> </li> </ul>   |
| Postconditions | Displays the recipe details  |