Cookable

By:

Copyright Team 5 CSC 648-848 Spring 2019

Trevor Sampson (Team Lead, Front & Back-end) tsampson1@mail.sfsu.edu

Thanh Dip (Back-end Lead Developer, Database & Back-end)

Pyae Naing (Back-end Developer, Github Master)

Tony Rodriguez (Front-end Lead Developer, UI)

Shivam Rai (Front-end Developer, UI)

Surabhi Chavan (Front-end Developer, UI)

Milestone 1

3/14/19

1. Executive Summary

Team 5 is designing a web client application called “Cookable” which will be the number one place for accessing food recipes on the web. This service will allow users to search for recipes based on ingredients that they will provide on the website. The users will also have the option of registering an account in order to access features that are only available to registered users. One main feature is that the registered user now has the option to save and store multiple ingredients that they currently own in their kitchen. This allows the website to suggest recipes that correspond to ingredients that the user has available to them. These features also include the option to “favorite/like” recipes that you would like to save for future use, as well as leaving reviews on recipes that users have made themselves. Registered users will also be able to submit their own recipes to the website for others to use. As more people join and use Cookable, more fun and exciting recipes will become available for people to use and enjoy.

2. Use Cases

Casual User:

- A casual user, Emily, wants to cook something at home using the specific ingredient. She can log into Cookable.com and receive recipe recommendations based on the ingredient she has provided.

Registered User:

- Emily decides to register for a Cookable.com account. She now has the option to input and save all of the ingredients she owns in her kitchen to the website, and receive hundreds of recommendations based on the ingredients she provides.
- A registered user, Brandon, is a skilled cook. He notices that on Cookable.com, one of his favorite recipes is missing from the website. He is given the option to add the recipe to the website so that other users will get to enjoy his recipe as well.
- A registered user shall be able to add to a shopping list when they are missing an ingredient to a recipe.

Restaurant Owner:

- Restaurants list there menus which can be advertised to users depending on the recipe. I.e a user who gets an apple pie recipe can be served an ad for a restaurant that sells apple pies.

Food Product Manufacturer:

- Food product manufacturers can provide recipes highlighting their products, and pay to have these recipes displayed to users through the homepage recommended recipes tab. I.E Del Monte foods can offer recipes for their products like canned corn which can be used with fried rice.

Meal Kit Services:

- Meal kit services like Hello Fresh and Blue Apron can also list their recipe and users who search results outputs recipes similar to what they offer can be served ads that lets them know they can get those ingredients delivered to them.

3. Data items and Entities

Casual User - Casual visitor who browses the website shall be able to view and search for recipes.
Registered User - Users who are registered can view and browse like a casual user but also has access to add recipes.
Admin - User who has elevated permissions, able to edit and delete recipes.
Privacy Policy - Shall denote policies about user's data and why, what, and where the data is stored.
Profile - User customization, preferences, and other data shall be saved here.
Recipes - Recipes will be stored in a table with the name of the food as well as the ingredients to make the food.
Recommendations Tab - This is where popular/recommended recipes will be displayed.
Recipe Instructions - Instructions for the recipes step by step.
Ingredients - A list of ingredients and other names that can be added.
Ingredients Available - Ingredients the user has to use to cook any recipe.
Ingredient Type - Groups ingredients into certain types like vegetables, meats, dairy, dairy alternatives, etc.
Likes - Counts of likes for each recipe respectively.
Favorite - Users can mark recipes and save them as favorites.
Comments - Each recipe can be commented on.
Food type - The type of food, i.e Chinese, Mongolian, American, Pies, Cookies, etc. A recipe can be multiple food types.
Meal Type - When the meal is usually eaten. I.e Breakfast, brunch, lunch, dinner.
General Type - A flexible type that can describe a recipe. I.e Healthy, Diet, Gluten Free, and anything that is not already accounted for.
Nutrition - The nutrition of the recipe if available things like calories or vitamins.

4. Functional Requirement

For users:

1. New users shall be able to register.
2. Registered users shall be able to log in.
3. Users shall be allowed to casually browse the site without having to .
4. A privacy policy shall be displayed to new users upon registration, and it shall be available under user settings.
5. Registered users shall be given the option to upload their photo on their profile.
6. Registered users shall be given the option to set default preferences such as their price range, store based on locations and favorite recipes.
7. A free text search box shall be displayed to all users which shall accept search terms regarding recipes, a button would be placed on this search bar to open ingredient search.
8. The default display in the free text search box shall be “Enter recipes”, with a button to switch to ingredients
9. Search parameters shall be price/store location of ingredients or cooking time/cuisine for recipes.
10. All (registered and casual) users shall be allowed to filter search results by type of prices ‘low to high’, ‘high to low ‘and/or user.
11. All users shall be able to sort search results by best match, user ratings, price range and/or the number of reviews.
12. All users shall be able to view a list of stores for an ingredient to buy with store rating and distance from pin code.
13. Upon clicking a recipe, users shall be able to view details including the cuisine, cooking time and process, number of people it serves with measurements of ingredients taken.
14. Registered users shall be able to cancel orders.
15. Casual users would still be able to browse and search for recipes, but they will not be able to maintain pantry or buy ingredients.
16. Upon clicking an ingredient, users shall be able to view basic details including the country of origin, different names associated with it (okra for ladyfinger and so on) and availability at nearest stores.
17. All users shall be able to add and store recipes to favorites.
18. All users shall be able to get maintain pantry with ingredients stocked with them. For each recipe, the pantry will have a user-based deduction of items and once the pantry is getting depleted of any ingredient, the user would be getting notifications about restocking.

19. There will be a sidebar to search from ingredients (need to review search feature if this needs to be added), you can type, and it will return you suggestions or you can find them ordered categorically in dairy, vegetables, fruits etc.
20. Store number and details would be featured if a user clicks on a store after searching for an ingredient. This will include store ratings.
21. An 'add to favorites' option shall be displayed so that registered users can save recipes or ingredients to this list.
22. The application shall keep track of registered user history including searches and orders.
23. Registered users shall be able to write reviews for outlets.
24. A help section shall be displayed to all users with details regarding how the application can be used.

5. Non-Functional Requirement

Security:

1. Login or creating account shall be a requirement for all user who wishes to save the information
2. The username will not be available if another user has already created the account with the same name
3. There will not be any restrictions and limits for a password.
4. The user will be stayed logged in unless he/she refresh the page.

Performance:

1. The load time should not take more than 1 min.
2. The search shall be executed in the background.
3. Query shall be executed in the background.

Availability:

1. The server shall be up for as long as admin wish it to be.
2. The server will be down for no more than 1 day if there is any maintenance.
3. The server shall not crash for any client-side reasons.

Recovery:

1. If there is any event where server crash, recover time shall not take longer than 1 day.
2. In the case of server failure, there should be through revision for the server log

Data and privacy:

1. Data will be backed up every week.

2. The administrator can view and look at any account data, including username and password.
3. There will not be any image uploads for the profile picture.

Conformance with Coding Standards:

1. The coding style will be professional and easy to read.
2. Only working code will be pushed or submitted.
3. All the code will be tested before submitting.
4. Any error will be stored in the log and be reviewed.
5. Any error shall be handled in a way that does not affect the functionality of the site.
6. This site shall be tested and debugged thoroughly before final submission.

Compatibility:

1. The site shall be compatible with the Safari browser.
2. The site shall be compatible with the IE explorer.
3. The site shall be compatible with the Firefox browser.
4. The site shall be compatible with the Chrome browser.
5. The site shall be compatible with most Mobile devices.

Product's Standard

1. The application will look casual and user-friendly.
2. The site will be simple to navigate and easy to be used by anyone.
3. The site will provide related or recommended products base on the current product.
4. The site will look the same across all different browser.

6. Competitive Analysis

Recipe Based Sites:	Supercook	Allrecipes	Epicurious	Cookable
Recommendations Tab (Recipes)	-	+	-	+
Recipe Instructions	+	+	+	+
Ingredients available to cook	+	-	-	+

Food Type	+	+	+	+
Feature to add personal recipes	-	-	-	+
Liked/ Saved / Favorites	-	+	-	+

Supercook offers an extensive variety of recipes on their site. With many features that allow easy access to these recipes by either searching for foods with a certain set of ingredients or even finding recipes to prepare with ingredients the user may suggest. However, Supercook lacks in features such as saving/favoriting recipes, adding personal recipes to their application, and navigating through what is their recommended recipe portion on their site. Allrecipes excels in what Supercook lacks by offering a variety of different recipes as well as links to recommended recipes. They also offer an option to save and personalize your homepage with recipes you have favorited. However, Allrecipes much like its competitor lacks the ability to add personal recipes to the application which is what we aim to solve. Finally, Epicurious was a site that had more articles and blogs than actual recipes. Our application will try to include all the features listed above along with the feature of allowing registered users to add personal recipes to our database.

7. High-level system architecture and technologies used:

Server Host: AWS t2.micro 1vCPU 1 GB RAM
Operating System: Ubuntu 16.04 Server
Web Server: NGINX 1.15.8
Server-Side language: Javascript
Additional Technologies:
Front-end library: React, Material-UI
Web Framework: Node.js, Express.js

1. Application shall be developed using a pre-approved set of Software development tools.
2. Application Front-end user interface will be developed using the React javascript framework.
3. Application will have an intuitive interface that consists of individually rendered and updated components in React.
4. Application's Front-end will also be developed using the React library called material.ui
5. Application's Back-end will be handled using javascript libraries node.js and express.js.
6. Application's data will be stored in a database using MySQL.

7. Application shall communicate between front and back-end using GET and POST calls.
8. Application port routing will be handled using NGINX version 1.15.8
9. Application shall use a Ubuntu 16.04 server and will be hosted and deployed on an AWS t2.micro EC2 instance containing 1vCPU and 1GB RAM.
10. Application shall be viewable in standard web browsers, and shall render correctly on all major browsers: Firefox, Safari, Chrome, Internet Explorer
11. User privacy shall be protected and all privacy policies will be communicated to users.

8. Team

Trevor Sampson: Team Lead, Back-end/Front-end Developer
Thanh Dip: Back-end Lead, Database Manager
Tony Rodriguez: Front-end Lead
Pyae Naing: Back-end Developer, Github Master
Shivam Rai: Front-end Developer
Surabhi Chavan: Front-end Developer

9. Checklist

- Team found a time slot to meet outside of class
  (ON TRACK)
- Github Master chosen
  (DONE)
- Team decided and agreed together on using the listed SW tools and deployment server
  (DONE)
- Team ready and able to use the chosen back and front end frameworks and those who need to learn are working on learning and practicing
  (ON TRACK)
- Team lead ensured that all team members read the final M1 and agree/ understand it before submission
  (DONE)
- Github organized as discussed in class (e.g. master branch, development branch, folder for milestone documents etc.)
  (DONE)