Night-In

Team 19

Project Backlog

Contributors: Jaden Zerbe, Kevin Murcia, Ryan Sand, Samuel Ingram, Adam Kogut

Problem Statement

Becoming an amateur home chef is a skill that many aspire to acquire. Our web-application gives the ability for anyone to quickly go online and create their own personal recipe book filled with custom, and tailored recipes based on their preferences. Preferences can be based on dietary restrictions, access to ingredients, or personal taste. Many online recipe websites contain massive amounts of fluff around the information the user is looking for, and don't allow the user to substitute similar ingredients. Our product will not only be straightforward with instructions and allow the user to save their favorite recipes and provide ratings, but will also allow for the substitution of ingredients based on the previously mentioned preferences of the user. Furthermore, we will provide a way for users to track the ingredients that they have on hand and allow them to make foods with only the ingredients they have or allow them to get a cost estimate for recipes with which they don't have all the ingredients for.

Background Information

Audience

Many people of all ages and backgrounds have an interest in learning how to cook whether it be for themselves or others. The problem lies in where to start once that person decides to go online looking for recipes and cooking instructions. Many websites contain a lot of fluff within their recipes or require an amount of overhead in terms of ingredients and utensils that a beginner may not have. Our web-app will be targeted at these novices up to the experienced home cook with the goal of providing simple, easy to understand recipes in an intuitive manner.

Similar Platforms & Limitations

The most similar platform to ours would be *allrecipes* which is a recipe website. However, like many recipe websites, it provides the recipes and instructions in a social media format with users writing their own public recipes for others to find. Our product will use an api to deliver more simplistic recipes without the ambiguity and fluff that can be involved in users writing their own recipes. Our product may be more simplistic and smaller than other websites, but we're trying to use that to our strengths as it will be more accessible and straight-forward. It also will hopefully include a meal-prep aspect to be able to scale these simplistic recipes into making multiple meals which is a unique component.

Environment

Our environment is going to be largely done using JavaScript. We are going to use React.js framework on the frontend along with Material-UI, Passport.js, and Redux, with Node.js and Express with Mongo on the backend. React will be used to build out the basic structure of our UI while using the components from Material-UI to help build out the actual components. Redux will keep track of our UI data that is fed to us through our Express backend.

As for testing, we will be using an amalgam of Nightwatch.js, Enzyme, and Chai HTTP with Mocha to build out a fully automated test suite. Nightwatch will be used for various integration tests that test the fully assembled app while Enzyme will be used for front end unit testing. Chai HTTP will be used to test the various API endpoints of our server to verify the correct handling and output of requests.

Functional Requirements

Backlog Id	Functional Requirements	Hours	Status
1	As a User, I would like to create an account	2	Planned for sprint 1
2	As a User, I would like to look up recipes	2	Planned for sprint 1
3	As a User, I would like to log into my account that I set up before	1	Planned for sprint 1
4	As a User, I would like to add my own custom recipes to my saved recipes lists	3	
5	As a User, I would like to log out of my account	1	Planned for sprint 1
6	As a User, I would like to create multiple grocery lists	4	
7	As a User, I would like to search by ingredients	3	Planned for sprint 1
8	As a User, I would like to set a list of ingredients that I cannot use	3	Planned for sprint 1
9	As a User, I would like to save searched recipes to my account	2	Planned for sprint 1
10	As a User, I would like to track my ingredients on hand	2	
11	As a User, I would like to write notes on my saved recipes	1	Planned for sprint 1
12	As a User, I would like to edit my saved searched recipes	2	
13	As a User, I would like to see a price breakdown of recipes	2	
14	As a User, I would like to see a price breakdown of recipes based on the current ingredients that I have on hand	2	
15	As a User, I would like to specify my diet (i.e. vegetarian, vegan, etc.) to receive relevant search results and save this to my account	2	

16	As a User, I would like to give ratings to particular recipes	5	
17	As a User, I would like to sort my saved recipes by ratings	5	
18	As a user, I would like to be able to print out recipes	1	Planned for sprint 1
19	As a user, I would like to be able to view random recipes	1	Planned for sprint 1
20	As a user, I would like to view nutrition information for recipes in a graphical format	4	
21	As a user, I would like to find substitutions for ingredients	3	
22	As a User, I would like to be able to delete recipes from my saved recipes	1	Planned for sprint 1

Non-Functional Requirements

- As a user, I would like my get my search results in a timely manner
- As a user, I would like the UI to be intuitive and easy to use
- As a developer, I would like for my application to run fast and with few (or no) noticeable bugs
- As a developer, I want a reasonable number of comments so that I may later understand code I wrote and code that other people wrote
- As a developer, I would like my application to store my users' data securely
- As a developer, I would like my application to run without crashing

Use Cases

Case: Create an account and log out (Functional Requirements: 1, 5)

User Action	**System Response**
1. Navigate to the home page of the app	2. The home page of the app displays
3. Click Create account/Login	4. App redirects to <website>/auth/google
4. Goes through the Google account process	5. App redirects to the homepage but Create account/Login now says Logout
6. Click Logout	7. App refreshes but Logout now says Create account/Login

Case: Login and search for a recipe (Functional Requirements: 2, 3, 15)

User Action	**System Response**
1. Navigate to the home page of the app	2. The home page of the app displays
3. Click Create account/Login	4. App redirects to <website>/auth/google
4. Goes through the login to Google account process	5. App redirects to the homepage but Create account/Login now says Logout
6. Click on the search bar, type a recipe in, and click search	7. The app displays recipes that are similar to what the user put in
8. Click on the advanced options menu near the search bar	9. Select a diet restriction and click search again
10. The app displays recipes that are similar to the user input and loosely adhering to the selected dietary restriction.	

Case: Add custom recipes (Functional Requirements: 4)

User Action	**System Response**
1. Login to app	2. The home page of the app displays
3. Click saved recipes button	4. App redirects to show the recipes that the user has saved
4. Click add recipe	5. A modal pops up with areas to put in the recipe name, ingredients, directions, nutrition, and possibly more
6. Insert all information, then click submit	7. App refreshes but now the new recipe is in

	the list of saved recipes

Case: Building a Grocery List (Functional Requirements: 10, 6)

User Action	**System Response**
1. Navigate to the Ingredients Tracker	2. The Ingredients Tracker system displays
3. Enter ingredients that I already have (i.e. Chicken, cheese, bread)	4. Ingredients display on list after being entered
4. Navigate to the Search section, search for a recipe, and select it (i.e. Chicken Casserole)	5. A list of recipes will be displayed followed by a popup modal upon selection
6. Click on the “Export Grocery List” button	7. A list is displayed containing the needed ingredients sans the ingredients already listed as on hand in the Ingredients Tracker
8. Print the list using the browser’s print functionality	

Case: Finding and Saving a Relevant Recipe (Functional Requirements: 7, 8, 9)

User Action	**System Response**
1. In the profile, click on the button to edit excluded ingredients	2. A modal pops up with options to configure ingredients
3. Input ingredients that the user does not want (i.e. they are allergic)	4. The list reflects the changes and they are saved in the system
5. Navigate to the Search page	6. The Search page is displayed with no results
7. Search for a recipe by inputting ingredients into the search bar	8. The site displays a list of recipes that do not contain the ingredients previously specified as excluded ingredients and that are still relevant to the ingredients input into the search
9. After finding a nice recipe, click on it	10. A modal displays the recipe and its details
11. Click on the “Save” button	12. A confirmation notification pops up showing that the recipe has been saved
13. Navigate to the saved recipes page	13. The recipe that was saved is now displayed in this section

Case: Editing and saving notes on saved searched recipes (Functional Requirements: 11, 12)

User Action	System Response
In the user’s list of saved recipes, click on a recipe	A modal displays the recipe and its details
Click on edit button	An editing submenu appears
Click on Add Note button	A small pop-up text box appears
Write something and click Add	The text will appear in a notes section of the recipe
Highlight some text and begin typing	The text will change to whatever is typed
Click on Save Changes	A confirmation notification pops up showing that the recipe has been saved

Case: See a price breakdown of a recipe and then include current ingredients (Functional Requirements: 13, 14)

User Action	System Response
Click on a recipe	A modal displays the recipe and its details
Click on price breakdown button	A list of the ingredients will appear along with the price of each ingredient and a total
Click on Include my ingredients button	The list and total will change to not include ingredients in the user’s list of on hand ingredients
Click on close button	The list will collapse

Case: Add and sort by recipe ratings (Functional Requirements: 16,17)

User Action	System Response
1. Login to app	2. The home page of the app displays
3. Click saved recipes button	4. App redirects to show the recipes that the user has saved
4. Click on rate recipe button	5. Modal appears and user enters a rating from 0-5 and click submit
6. Modal disappears and rating is shown if not already present	7. Click on sort by rating descending button
8. Saved recipes are listed in descending order by rating	

Case: Print a recipe (Functional Requirements: 18)

User Action	**System Response**
1. Login to app	2. The home page of the app displays
3. Click saved recipes button	4. App redirects to show the recipes that the user has saved
4. Click on a recipe	5. Recipe details appear on screen and user clicks print
6. Print preview modal appears, the user selects their printer, and clicks print	7. The modal closes and the recipe prints to the user’s printer

Case: View random recipes (Functional Requirements: 19)

User Action	**System Response**
1. Login to app	2. The home page of the app displays
3. Click random recipes button	4. A list of random recipes appears

Case: View nutritional information (Functional Requirements: 20)

User Action	**System Response**
1. Login to app	2. The home page of the app displays
3. Click saved recipes button or search for recipes	4. App redirects to show the recipes that the user has saved or searched
5. Click on recipe	6. Click on view nutrition facts
7. Nutrition facts appear on screen in graphical format	

Case: Substitute ingredients (Functional Requirements: 21)

User Action	**System Response**
1. Login to app	2. The home page of the app displays
3. Click saved recipes button or search for recipes	4. App redirects to show the recipes that the user has saved or searched
5. Click on recipe	6. Click on arrows next to ingredients
7. Ingredient changes to similar ingredient	

Case: View nutritional information (Functional Requirements: 22)

User Action	**System Response**
1. Login to app	2. The home page of the app displays
3. Click saved recipes button	4. App redirects to show the recipes that the user has saved
5. Click on trash can/ delete icon next to recipe	6. “Are you sure” window pops up
7. Click “Yes”	8. Recipe is removed from saved list