UNSW
AUSTRALIA

The University of New South Wales

School of Computer Science and Engineering

CookBook Final Report

(E-Commerce Recommender System)

COMP3900 T3 2021

Computer Science Project

Author	Role
Jenna Chan	Scrum Master
Trina (Ya-Ting) Chang	Back end developer
Maria Cuyutupa Garcia	Back end developer
Louis Limmarga	Front end developer
Wincent Winarko	Front end developer

Table of Contents

Introduction

CookBook is an ecommerce system which sells grocery items filtered by recipe. A user will search or be recommended a recipe, then proceed to add all required ingredients to the cart with a few clicks. This increases the convenience of shopping for groceries as items do not need to be added and searched for individually. The platform also allows users to share, recipes and link ingredients so that other users can purchases them. CookBook allows for customisation of the cart so items can be added, removed or replaced, giving the customer increased flexibility.

CookBook has two novel features: 1. The recommender system and 2. The community functions. CookBook has the functionality to make several types of recommendations – recommendations of recipes based on past purchases, questionnaire results, similar recipes or similar ingredients. The recommendation system encourages users to try new recipes and buy new ingredients and hence increases profits. Community functions such as adding recipes and rating/ commenting on recipes is highly encouraged through the use of CookBook Rewards points. Points are award when another user purchases items one's uploaded recipe. These points can be converted into monetary values for use in a future purchase.

Overview - System Architecture/ design

This section discusses the various layers to the CookBook system architecture. Figure 1 below models the architecture by showing elements of the interface, business, API, database and interface layers as well as their interactions. The subsections below explain each layer in more detail.

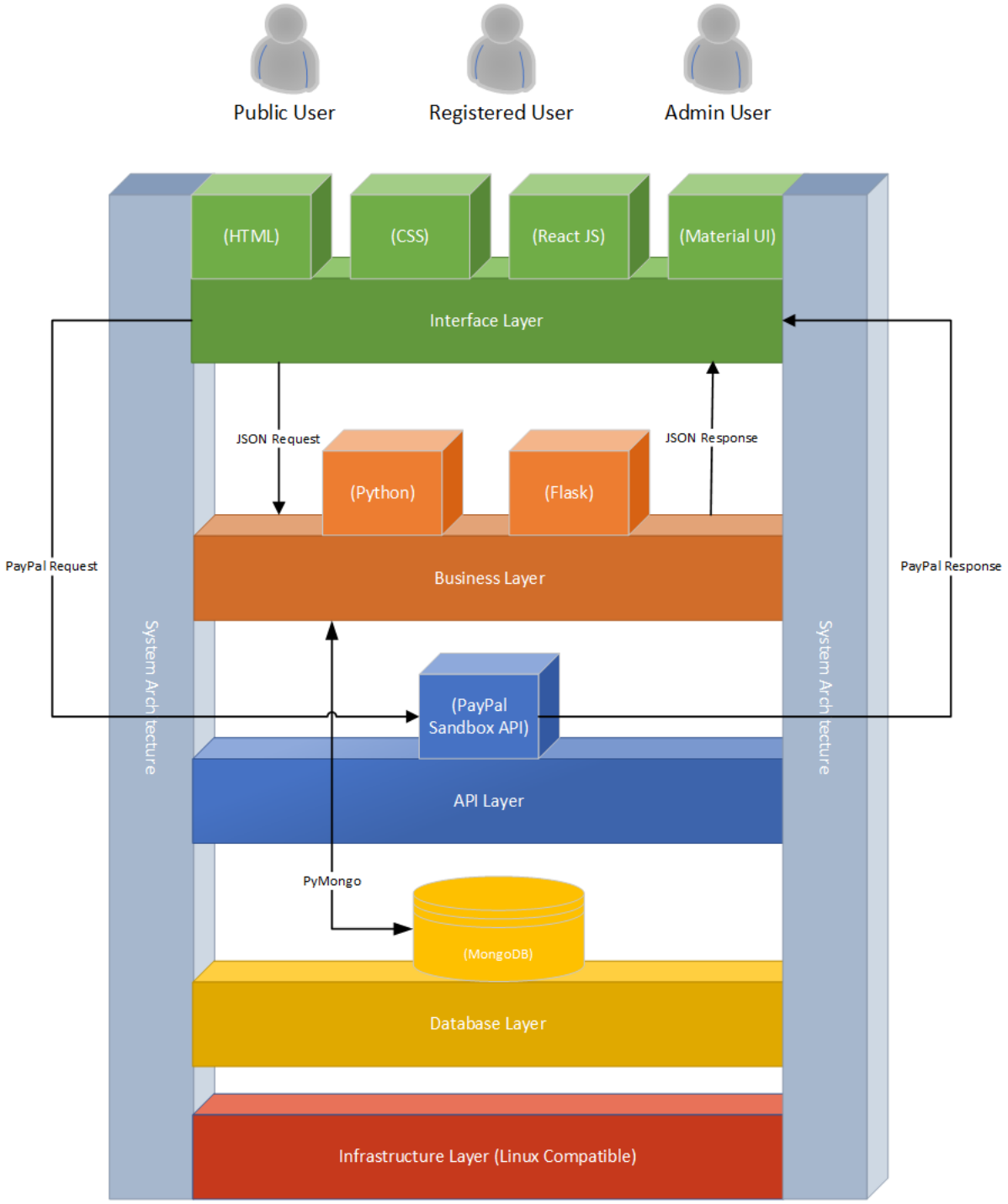

Figure 1: System Architecture

User Types

There are three types of users in the CookBook system: public users, registered users and admins. Public users do not have a CookBook account and are completely limited to recipe viewing. Registered users

are the customers of CookBook and can use the platform to purchase ingredients. Admins have the most privileges including adding products, editing recipes and viewing statistics. The users interact with the system via the interface.

Interface Layer

The interface layer consists of HTML, CSS, ReactJS and MaterialUI. HTML was used to design the structure while CSS was used to style the application. ReactJS was used to develop the user interface and MaterialUI was used for its user interface components. The user interface allows users to interact with the system. The interface layer communicates with the business layer through JSON requests.

Business Layer

Python and Flask belong in the business layer. Python was used develop the back end and communicate with the database. The Flask framework was used to build the server. The business layer responds to the interface layer through JSON responses.

API Layer

The third party PayPal Sandbox API was used to integrate PayPal payment for purchases. Users can choose to use PayPal for safe e-commerce transactions. The PayPal API receives request from and returns responses to the interface layer.

Database Layer

A MongoDB database was used for CookBook. NoSQL was used to design the database and the PyMongo package was used to connect the MongoDB database to the business layer.

Infrastructure Layer

CookBook is designed to be compatible with Linux. It is suitable for use on the UNSW CSE vlab machines.

Functionalities

During the proposal stage, 11 key objectives for CookBook were outlined. This section will describe the functionalities of CookBook and how they address all these objectives and their associated user stories. It will also cover any implementation/ development challenges. The objectives have been split into the seven functionalities in the subsections below. For reference, the objectives and their associated Jira (comp3900-h18b-internship404) user story codes are listed below.

1. Admin can add and edit recipes and product information (Ref: CHI-9, CHI-30, CHI-31)
2. Admin can view and edit purchase order information (Ref: CHI-6, CHI-10)
3. Admin can view product sales and statistics. (Ref: CHI-7)
4. Public users can register an account and sign in. (Ref: CHI-1, CHI-2, CHI-3)
5. Public and registered users can browse the site for recipes. (Ref: CHI-12, CHI-32)
6. Registered users can upload and edit recipes. (Ref: CHI-5, CHI-9, CHI-11)
7. Registered users can add ingredients from recipes to the cart. (Ref: CHI-5, CHI-20, CHI-21, CHI-28)
8. Registered users can purchase items from their cart using online payment and delivery. (Ref: CHI-22, CHI-28, CHI-34)
9. Registered users can view the status of their current and past orders. (Ref: CHI-35, CHI-37)
10. Registered users can engage in the with other users through points, likes and comments. (Ref: CHI-11, CHI-17, CHI-18, CHI-38, CHI-19)
11. Registered users can view relevant suggestions for recipes. (Ref: CHI-13, CHI-14, CHI-39)

Admin Management

Admin management functionality satisfies objectives 1-3. Admins have elevated privileges compared to standard user accounts. Users are assigned admin privileges during registration if their email address domain is '@cookbook.com'. In addition to recipe edit permissions, admins can also add and edit products (ingredients) to the catalogue. Admin accounts can view and update the status of all purchased orders. They can also see statistics of the sales from the catalogue. Table 1 shows the list of completed user stories for admin management.

Table 1: Admin management user stories

Ref	User Story
CHI-9	As an admin, I want to add recipes, so that I can contribute to the catalogue. • Admin can link ingredients to a recipe. • Admin can add steps to a recipe. • Recipe details such as preparation time, cooking time and difficulty can be added.
CHI-30	As an admin, I want to edit recipes, so that I can ensure all recipes are accurate. • Admin can delete a recipe. • Admin can edit ingredients of a recipe. • Admin can modify method of a recipe.
CHI-31	As an admin, I want to add products, so that users can link them to recipes and purchase them. • Admin can edit product details such as name and price. • Admin can remove a product for sale. • Admin can mark a product as sold out.
CHI-6	As an admin, I want to view order details, so that I can prepare orders. • Admin can view order owner, items, time of order, delivery address, current status. • Orders can be sorted by time of order.

CHI-10	As an admin, I want to mark an order as delivered so that the user can track their order. • Admin can mark an ordered as ‘delivered’. • Admin can mark an order as ‘in transit’. • Users can see the updated status.
CHI-7	As an admin, I want to view the sales of the products, so that I can analyse sales. • Admin can see the sales of an individual product.

Implementation Challenges

There were not many challenges regarding implementing the admin functions. This was because many of the user stories addressed similar functions to those for registered users. For example, adding/ editing recipes have the same functionality, but different permissions depending on the user type. All users have a Boolean admin flag, which if true, gave access to the admin specific functions. Other admin user stories involved adding, editing or viewing information from the database.

When implementing the sales information for admin viewing, there was some implantation challenges with displaying the required information. Such as through a graph or table or singular result. For simplicity, it was chosen to allow the admin to search for results for all time, over a certain period or for a ingredient as per the acceptance criteria. No graphs were generated as they would have introduced more confusion.

Another challenge was implementing the order, recipe and product dashboard in account. Each page must be refreshed constantly to load the details. This increases the loading time when the user is changing page in the dashboard. One of the solution is to implement infinite scrolling page so it won't face the problem of refreshing each time the user change the page.

Account Management

Account management satisfies objectives 4. A public user can become a registered user by logging in or creating an account to access additional features of CookBook. Some notable functions available for registered users include: the recommender system which suggests recipes based on a short questionnaire; the ability add ingredients from a recipe to their cart for purchasing; and community functions such as following other users or uploading, commenting, and rating recipes. Each registered user has a public profile page showcasing the recipes which they have uploaded. Table 2 shows the list of completed user stories for admin management.

Table 2: Account management user stories

Ref	**User Story**
CHI-1	As a public user, I want to be able to create an account, so that I can use functions that are only available to registered users. • User needs to set the username. • User needs to set the password. • The username needs to be unique. • The password needs to contain at least 8 characters.
CHI-2	As a public user, I want to be able to log in, so that I can view the website as a registered user. • User needs to login with the correct username and password. • Users can log in by clicking the login button on the top right.
CHI-3	As a registered user, I want to be able to log out, so that I can view the website as a public user. • Users can log out by clicking the logout button on the top right.

Implementation Challenges

Most of the account management user stories involved user log-in and authentication. Authentication was implemented using tokens. One implementation challenge that arose was dealing with users logged into the same account on multiple devices. The main issue was the dealing with different request operations from the logged in device. The resolution was to reject any secondary user attempting to log in, allowing only one device to be logged into an account at a single time.

Recipe Functions

Recipe functions satisfy objectives 5-6. Depending on the user's privileges, functions allow users to view, add and edit recipes on the CookBook database. One of the most important elements was linking ingredients to recipes such that a user could add all necessary ingredients for a meal to the cart in one click. A user may access recipes from various pages such as the home page, through category search, search bar results or profiles. Table 3 shows the list of completed user stories related to recipe functions.

Table 3: Recipe functions user stories

Ref	User Story
CHI-12	As a public user, I want to view uploaded recipes, so that I can learn how to cook a meal. • Can view methods and ingredients. • Can view details such as cooking and preparation time and difficulty. • Public users cannot purchase ingredients. • Public users cannot comment or rate recipes.
CHI-32	As a registered user, I want to search recipes based on tags, so I can refine my search. • Recipes will have associated tags which can be used to filter results. • Users can see results from a search for specific tags.
CHI-5	As a registered user, I want to be able to edit the recipe I uploaded, so that I can ensure all the information is accurate. • Users can delete the recipe by clicking the remove button on the recipe. • Users can edit ingredients of a recipe. • Users can modify the method of a recipe. • Users can edit the detailed information of a recipe such as preparation time, cooking time and difficulty.
CHI-9	As an admin, I want to add recipes, so that I can contribute to the catalogue. • Admin can link ingredients to a recipe. • Admin can add steps to a recipe. • Recipe details such as preparation time, cooking time and difficulty can be added.
CHI-11	As a registered user, I want to share my recipes, so that I can accumulate points. • Recipes must contain ingredients offered in the catalogue. • Recipes must contain details such as preparation time, cooking time and difficulty. • Recipes must contain a procedure.

Implementation Challenges

The method used to implement tags was discussed thoroughly. This was because the use of tags could assist in developing various functionality such as searching for results, filtering for results and recommending similar results.

Another challenge that arose was the implementation of the CookBook rewards points system. A recipe owner receives points when other users purchase ingredients through their recipes. The quantity of the points relates to the cost of items purchased. Since ingredients can be added and removed from the cart, and ingredients can appear in multiple recipes a method to ensure the correct user collected the correct number of points was required. The solution to this challenge was to group items by recipe in the cart and reward the recipe owner with the point value of the items remaining in the specific recipe.

It was identified that allowing users to modify the quantity of each ingredient would provide ambiguity towards the number of 'recipes' purchased. To resolve this, the functionality to view the recommended quantity of each ingredient was added for the user's ease of use.

No challenges arose when implanting add and edit recipes functions as it was directly related to manipulating and updating the database.

Order/ Purchase/ Payment Functions

Purchase and payment functions map to objectives 7-9. These functions refer to the process of adding items to the cart, editing items in the cart, proceeding to checkout and payment. As users are purchasing ingredients through the recipe interface, the option to edit or remove items in the cart. During checkout a calculation of CookBook reward points to be added to recipe owners is also performed. At the payment stage, users have the option to use purchase using PayPal or the equivalent monetary value of their CookBook reward points. PayPal was chosen as a safe way to complete online transactions. Table 4 shows the related user stories and acceptance criteria for order/ purchase/ payment functions

Table 4: Order/ purchase/ payment function user stories

Ref	**User Story**
CHI-5	As a registered user, I want to be able to edit the recipe I uploaded, so that I can ensure all the information is accurate. • Users can delete the recipe by clicking the remove button on the recipe. • Users can edit ingredients of a recipe. • Users can modify the method of a recipe. • Users can edit the detailed information of a recipe such as preparation time, cooking time and difficulty.
CHI-20	As a registered user, I want to add recipe ingredients to my cart, so I can purchase them online. • Items can be individually added to the cart. • Set of items can be added to cart via recipes. • Items can be removed from carts.
CHI-21	As a registered user, I want to modify the ingredients from a recipe before adding to cart, so that I can make customisations. • Items can be removed from recipes. • The quantity can be changed from the recipe. • The brand of the product can be changed from the recipe.
CHI-28	As a registered user, I want to view my cart, so I can see what items I have added. • User can click on cart button on top right to see current items in cart. • User can see all items added and the quantity.
CHI-22	As a registered user, I want to check out my cart, so I can make my purchases. • Users can see a review of their cart. • Users can proceed to payment and delivery.
CHI-34	As a registered user, I want to input my payment of choice so that I can confirm my purchase.

	• Users can input their PayPal details. • Users can see confirmation of their payment. • Users can choose to use points. • User receives an email confirmation.
CHI-35	As a registered user, I want to be able to view my order history, so that I can see what I have ordered in the past. • Users can view the date of the past orders. • Users can view the items bought in each past order. • Users can view the cost of the past orders. • Users can view the pickup date of the past orders.
CHI-37	As a registered user, I want to be able to see if my order has been delivered, so that I can pick up my groceries. • If the order has already been delivered, "delivered" will be shown on the order management page. • If the order has not yet been delivered, "not yet delivered" will be shown on the order management page. • The pickup location will be shown on the order management page.

PayPal Sandbox (3rd Party Functionality)

Payment functionality was completed using PayPal SandBox (PayPal, 2021). It was initially challenging to implement this functionally due to lack of knowledge and research. This was because it was originally understood that the PayPal Sandbox API was connected to the backend, however, after discussions and reading of the PayPal Sandbox API documentation, the issue was corrected. Later, the PayPal function was successfully implemented on the front end. Since PayPl Sandbox is a virtual testing environment, PayPal functionality could be tested without real money.

Implementation Challenges

Another challenge was fulfilling the response requirement. The acceptance criteria for a successful purchase was an email confirmation. Much research alongside trial and error was needed to make the email format presentable. HTML was used to format the email template.

Finally due to time restrictions implementation of accurate shipping/ delivery details was considered. To reduce the scope of the assignment, default timing and dispatch messages templates were used.

Community Functions

Community functions map to novel objective 10. The community functions include the ability to view other user profiles, follow other users, comment, and rate recipes. The sharing of recipes to the community is encouraged through the CookBook rewards points system. Table 5 shows the user stories related to the community functions.

Table 5: Community functions user stories

Ref	User Story
CHI-17	As a registered user, I want to be able to view profiles of other users, so that I can see what recipes they have uploaded. • Registered users can view the profile of other users by clicking them. • The recipes the user has uploaded will be shown on the profile. • The recipes will be shown according to the time it is uploaded from the newest to the oldest. • The number of the following users will be shown on the profile.

CHI-18	As a registered user, I want to be able to follow other users, so that I can see their new recipes when they upload them. • Registered users can follow other users by clicking the follow button in the user's profile. • Users can check which users they are following in the following user section. • The recipes from the following users will be shown in the designated section.
CHI-38	As a registered user, I want to be able to comment on the recipes, so that I can share my opinions about the recipes. • Registered users can comment on the recipes in the comment section. • Users can view the comments from other users. • The comment is ordered from the oldest to the newest.
CHI-19	As a registered user, I want to be able to rate recipes, so that I can improve the recipes rating. • Registered users can rate recipes by commenting under the recipe in the recipe page. • Every user can see the rating of the recipes.

Implementation Challenges

Ratings were implemented with a 5-star scale. This allows for users to effectively rate the recipe in addition to commenting. The average rating could then be used to calculate the average rating, which was further used for the recommendation functions. Ratings and profile implementations did not have significant challenges.

Recommendation Functions

Recommendation functions map to objective 11. There are several recommendation methods that have been implemented in CookBook. The first is recommendations of recipes based on the answer to a short questionnaire. This type was designed based on tags associated to the recipes. The second type of recommendation is for ingredients. Similar ingredients can be recommended as replacements or substitutions. Table 6 shows the user stories related to recommendation functions.

Table 6: Recommendation functions user stories

Ref	**User Story**
CHI-13	As a registered user, I want to be able to get recommendations from the system according to the questions asked, so that I can get suggestions based on my preference. • The recommendation will ask the user several questions and give recommendations based on the answer of the user. • The question set acts as a filter.
CHI-14	As a registered user, I want to get recommendations based on my account history, so that I can get suggestions for recipes. • The recommendations are shown on the homepage. • The recommendations are shown at the bottom of the recipe pages. • The recommendation algorithm considers order history as one of the factors.
CHI-39	As a registered user, I want to view recommendations for similar ingredients, so that I can easily replace ones I do not want. • The recommendation algorithm considers different brands of the same ingredient. • User is given different replacement options • Recommendations are shown once a user edits an ingredient

Implementation Challenges

As the recommendation systems were a novel feature, this section contains more detailed information regarding its implementation and the algorithms used.

Recipe recommendations based on history

Recommendations based on history has the most complex algorithm. It considers a user's past purchases as well as the ratings of each recipe. It will search and rank the labels associated with recipes from past purchases and assign each a rank dependant on number of appearances. This is stored in a dictionary.

As an example, a user has purchased 3 recipes. The link related to these recipes are ["Chicken", "Dinner", "Australian"], ["Chicken", "Lunch", "Australian"], ["Chicken", "Dinner", "Asian"]. The points calculated for each tag would be [{"Chicken": 3}, {"Dinner": 2}, {"Australian": 2}, {"Lunch": 1}, {"Asian": 1}].

Then, for each recipe in the database, points are calculated based on the above dictionary and their average rating. The formula used is:

$$Recipe\ points = \sum matching\ TagPoints * Avg\ RatingPoints * 0.4$$

The figure 0.4 comes from 1/2.5 (2.5 is the middle rating of 5 stars). This remove the bias of recipes with more labels (and hence higher sum of label points) so that recipes with a rating below 2.5 are recommended less even if they have higher label points.

For example, a 4 star recipe A with tags ["Chicken", "Lunch", and "Australian"] would have recipe points:

$$Recipe\ A\ pts = (Chicken\ tag\ pts + \ lunch\ tag\ pts + Australian\ tag\ pts) * 4 * 0.4$$

$$= (3 + 1 + 2) * 4 * 0.4$$

$$= 9.6$$

Whereas a 3 star recipe B with tags ["Chicken", "breakfast", "Asian"] would have recipe points:

$$Recipe\ B\ pts = (Chicken\ tag\ pts + \ breakfast\ tag\ pts + Asian\ tag\ pts) * 3 * 0.4$$

$$= (3 + 0 + 1) * 3 * 0.4$$

$$= 4.8$$

An example showing the how the tag bias is removed can be done with a 1 star recipe C with the tags ["Chicken", "Dinner", "Australian"]. The recipe has high number of tag points but a low rating. The '0.4' will remove this bias. The points for the recipe will be:

$$Recipe\ C\ pts = (Chicken\ tag\ pts + \ dinner\ tag\ pts + Australian\ tag\ pts) * 1 * 0.4$$

$$= (3 + 2 + 2) * 1 * 0.4$$

$$= 2.8$$

The first recipe A would be highly recommended followed by recipe C and Recipe B.

The recipes are then displayed in order of the points calculated, showing the recipes with increased similarity and higher rankings first.

Recipe recommendations based on questionnaire

The CookBook platform can also provide recommendations based on a short questionnaire filled out by the user. In the questionnaire, the user must select relevant categories for which they enjoy or want to receive recommendations. The recommendation system assigns all recipes recipe points as with the recommendations by history. It then filters the recipes with the highest points. The average of the max recipe points and min recipe points is calculated, and all recipes which have a higher score than the average are displayed. Of these, the recipes are ranked from highest to lowest points.

Similar ingredient recommendation

Ingredient recommendation also uses labels to determine a suitable replacement. The user can swap ingredients after the recipe has been added to the cart. The algorithm looks for ingredients which contain the same flags as the given ingredient and suggests replacements with the same tags. Since all ingredients are uploaded with tag information, similar products can be recommended.

Similar Recipe Recommendation

Similar recipe recommendations are given at the bottom of recipe pages. They show recipes which may be like the current recipe. The algorithm used to determine a recipes' similarity is like recipe recommendations by questionnaire. All recipes are given points using the recipe points formula mentioned above, and an ordered (by points) list is returned.

Additional Implementation Challenges

Further implementation challenges include working with the responsiveness of the website. This was due to different screen sizes. The solution was to utilise relative lengths (vw and vh) for most of the components instead of pixel. Additionally a standard a screen size was recommended.

Third-Party Services

The table below shows a summary of third-party systems used. These systems were utilised to provide a better experience for the end user.

Table 7: Third Party Services

Third Party Application	Description	Implementation/ Justification of use	Licensing impacts	Reference
Paypal Sandbox	Virtual testing environment which simulates the live PayPal production environment.	Paypal allows for secure transactions for users. Processes behave the same way as they would in	As it is just a simulation for testing transaction, no licensing impacts.	(PayPal, 2021)
MongoDB	A cloud document database	MonogDB was chosen as the schema was more flexible compared to other databases. Faster performance compared to other relational databases. Supports JSON.	Community version is free and has an open database.	(MongoDB, 2021)

Libraries

The Table 8 below outlines additional libraries used during development. The second column identifies the reason why each library was chosen and used.

Table 8: Additional Libraries Used

Library/ Package	Reason for use	Reference
ReactJS	Building the cookbook user interface.	(Facebook Inc, 2021)
Material-UI	React components for user interface development.	(MaterialUI, 2021)
Axios	Supports HTTP requests from node.js	(npm, 2021)
Flask	Framework used to build the CookBook	(Pallets, 2010)
Flask_cors	A Flask extension for handling Cross Origin Resource Sharing (CORS), making cross-origin AJAX possible.	(Flask, 2021)
PyJWT	Encoding and decoding JSON web tokens	(Python, 2021)
PyMongo	Connecting to MongoDB to the backend (python)	(Mongo DB, 2021)
dsnPython	Connect to cloud so PyMongo can be used as a python MongoDB library	(Python, 2021)
pretty_html_table	Making the formatting of tables for the email generation more visually appealing.	(Python, 2021)

User documentation

Project Links

GitHub Repo: https://github.com/unsw-cse-comp3900-9900-21T3/capstone-project-3900-h18b-internship404

Jira Page: https://comp3900-h18b-internship404.atlassian.net/jira/software/projects/CHI/boards/1

Directory Structure

- backend/
  - Contains all backend code using Flask framework
  - Contains connection to MongoDB
- frontend/cookbook
  - Contains all frontend code using NodeJS
  - frontend/src/components/ contains all reactJS components
  - subdirectories categorise each page's components
    - frontend/src/pages/ contains all pages for CookBook frontend
- Diaries/
  - Contains individual diaries from all students in the group

Initial Build/ Configuration Instructions

Notes: Make sure you have installed node.js (recommended version: 12.16.2) and python3 (Python 3.6, pip >= 19.0) beforehand

Ideal screen size should be 1920x 1080

1. Git clone or download this repo
```
$ git clone git@github.com:unsw-cse-comp3900-9900-21T3/capstone-project-3900-h18b-internship404.git CookBook
```

2. Change directory to CookBook's backend
```
$ cd CookBook/backend
```

3. Pip install requirement.txt to install required python libraries
```
$ pip3 install -r requirement.txt
```

4. Run the backend server (Flask application will run on localhost:5000)
```
$ python3 server.py
```

5. Open new terminal and Change directory to CookBook's frontend/cookbook
```
$ cd CookBook/frontend/cookbook
```

6. Run npm install to install required libraries
```
$ npm install
```

7. Run npm start to run the frontend server (React application will run on localhost:3000)
```
$ npm start
```

8. Open http://localhost:3000 with your favourite brower and enjoy our service

General Run Instructions

Notes: make sure you are on root directory

1. Change directory to backend directory
```
$ cd CookBook/backend
```

2. Run the backend server (Flask application will run on localhost:5000)
```
$ python3 server.py
```

3. Open new terminal and Change directory to CookBook's frontend/cookbook
```
$ cd CookBook/frontend/cookbook
```

4. Run npm start to run the frontend server (React application will run on localhost:3000)
```
$ npm start
```

5. Open http://localhost:3000 with your favourite brower and enjoy our service

User Manual – Public User

Home Page

When first launching CookBook, the user will be able to view something similar to Figure 2. This is the public user home page. The user must register or log-in to access the full functionality of CookBook.

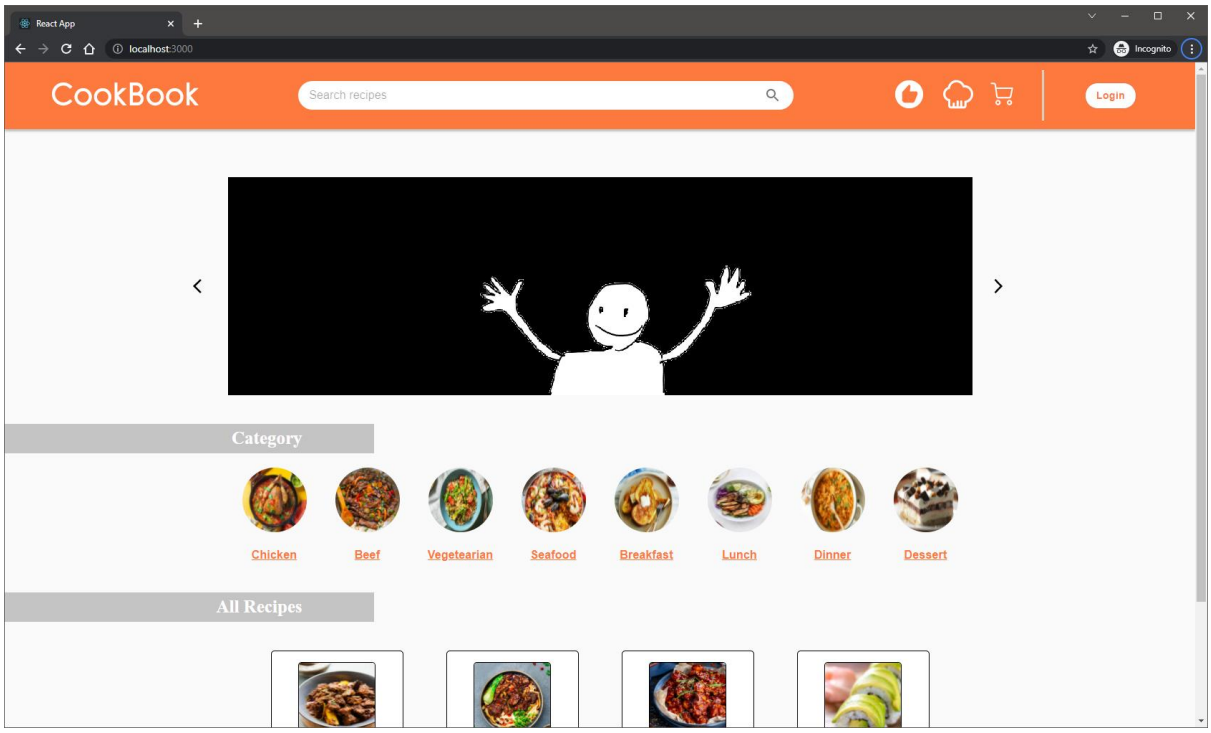

Figure 2: Public User Home Page

Account Registration

To register, the user must click the ‘Login’ button in the top right corner of Figure 2. Unregistered users will need to click ‘Register’ (Figure 3), and fill in the required information in Figure 4. Once registered, proceed to the next section of the manual (User Manual – Registered User).

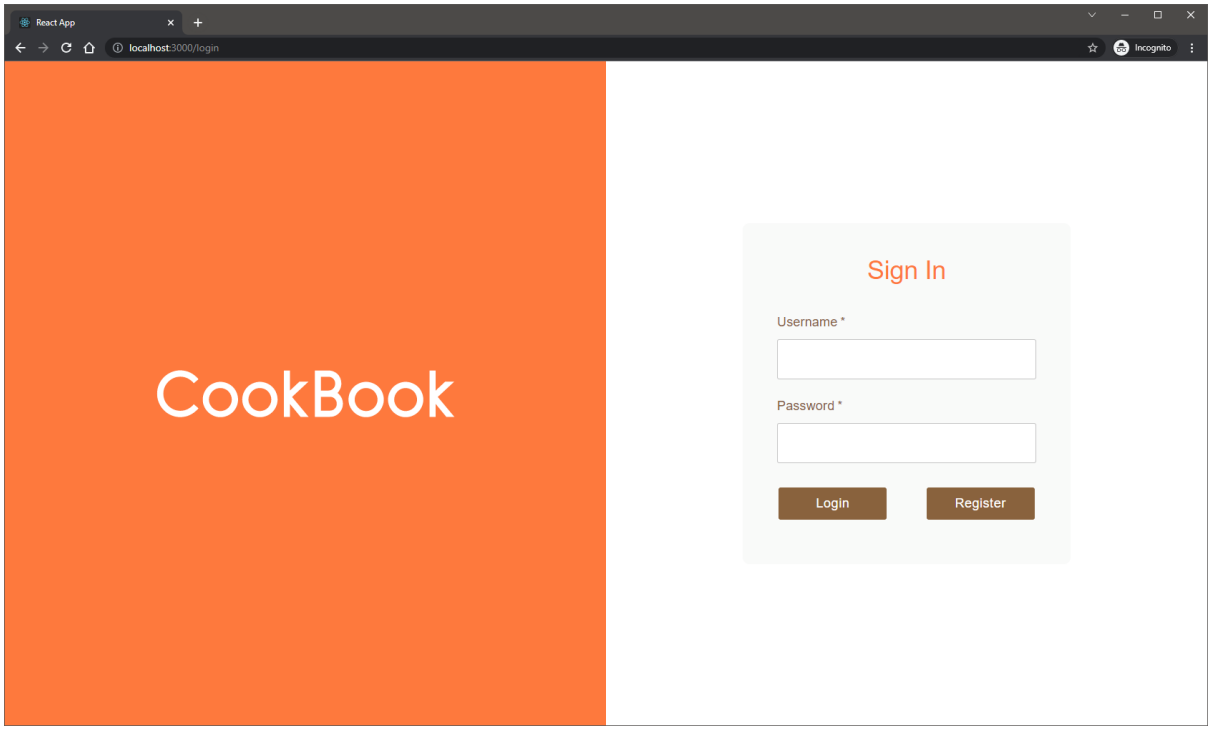

Figure 3: Sign In/ Registration Page

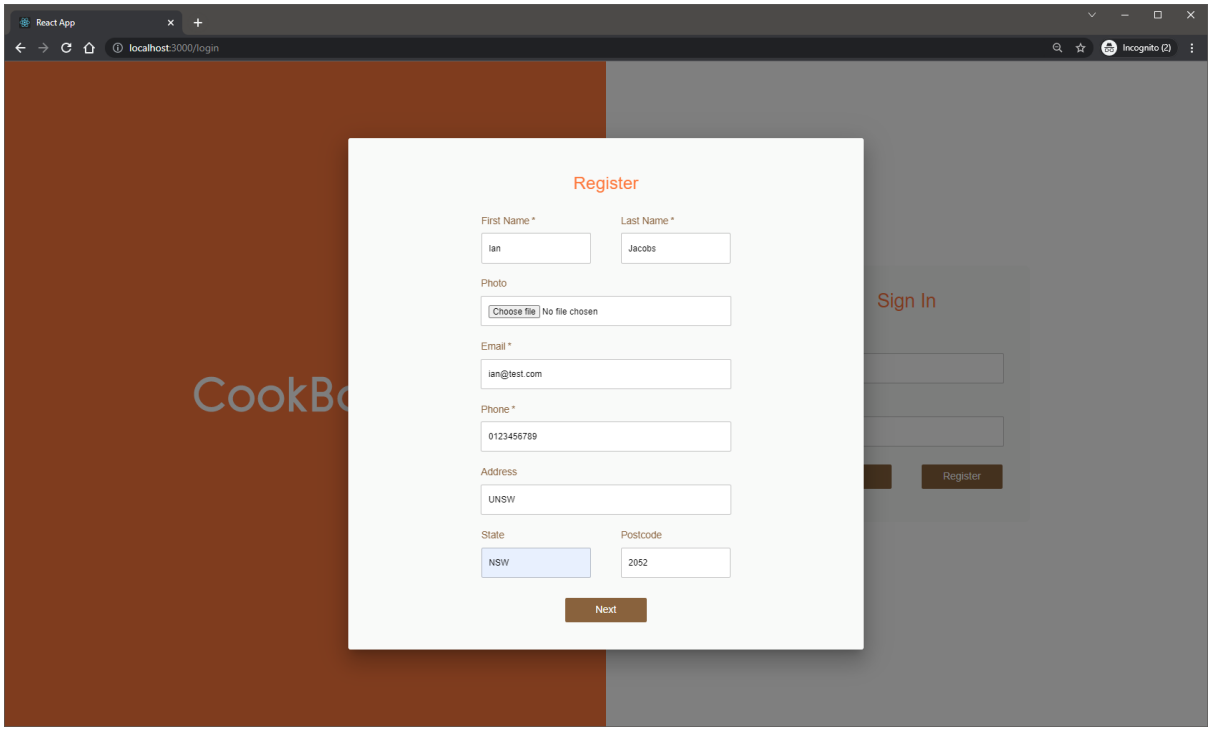

Figure 4: Registration Page

User Manual – Registered User

Once logged in, the user will be directed to the home page (as in Figure 2), however, the log in button will show the user uploaded profile picture/ or default image.

Viewing Recipes

Clicking on a recipe will bring up recipe details such as ingredients, steps, cook/ prep time, rating, etc… An example is shown in Figure 5.

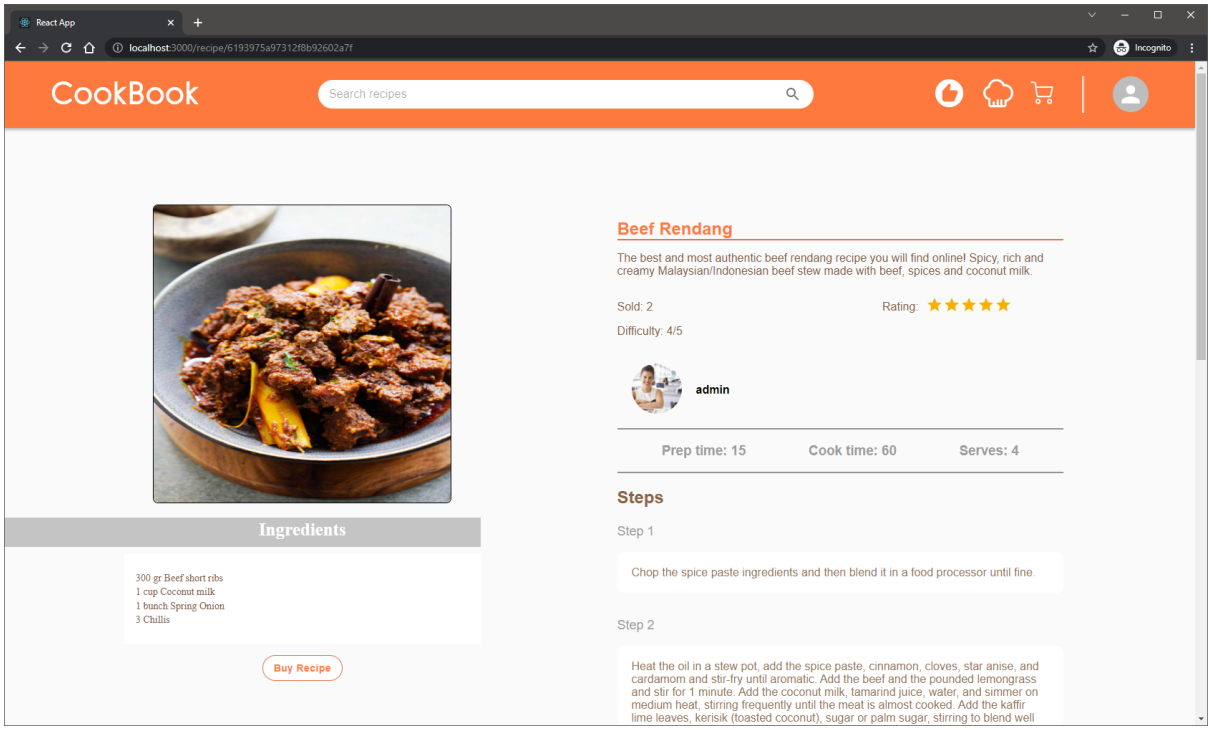

Figure 5: Recipe View

Adding Recipes to Cart

By clicking ‘Buy Recipe’ below the listed ingredients on the left of Figure 5, a user will be able to add all required ingredients to their cart (Figure 6). Here, the user can also edit the quantity or swap out ingredients. Figure 7 shows an example of swapping beef short rips to minced beef. The ingredients are recommended using the recommender algorithm.

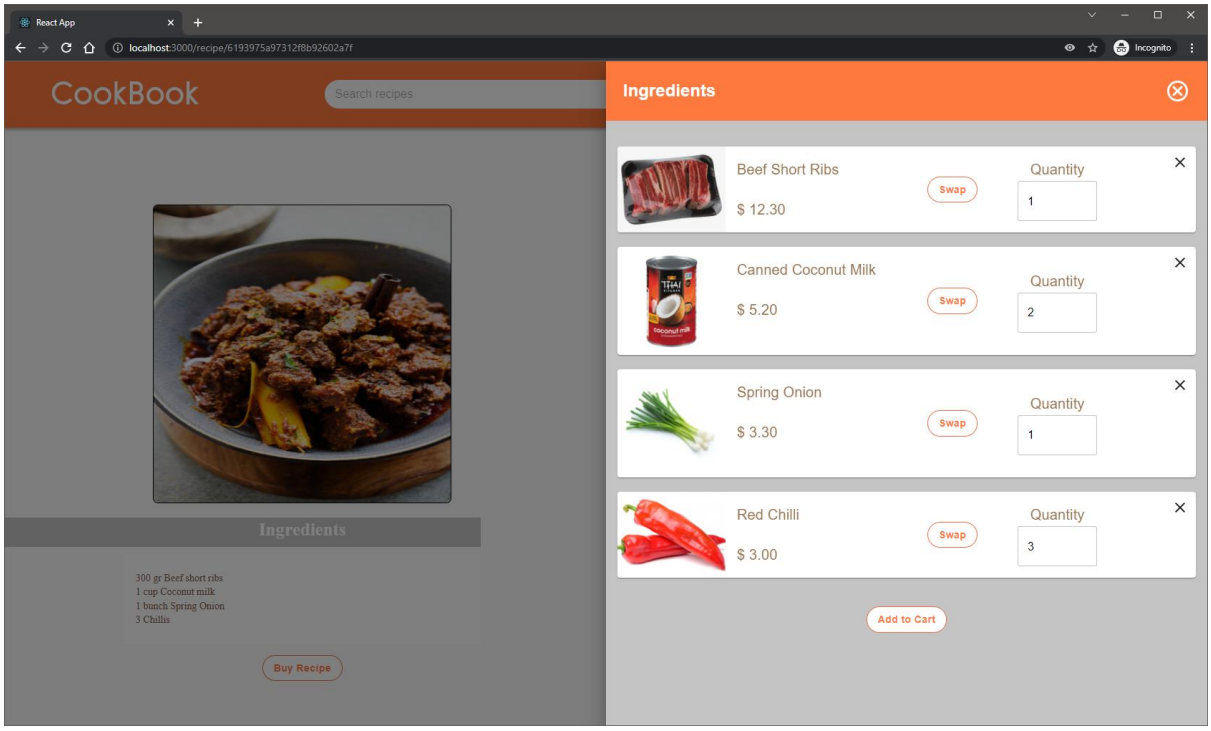

Figure 6: Add Ingredients to Cart

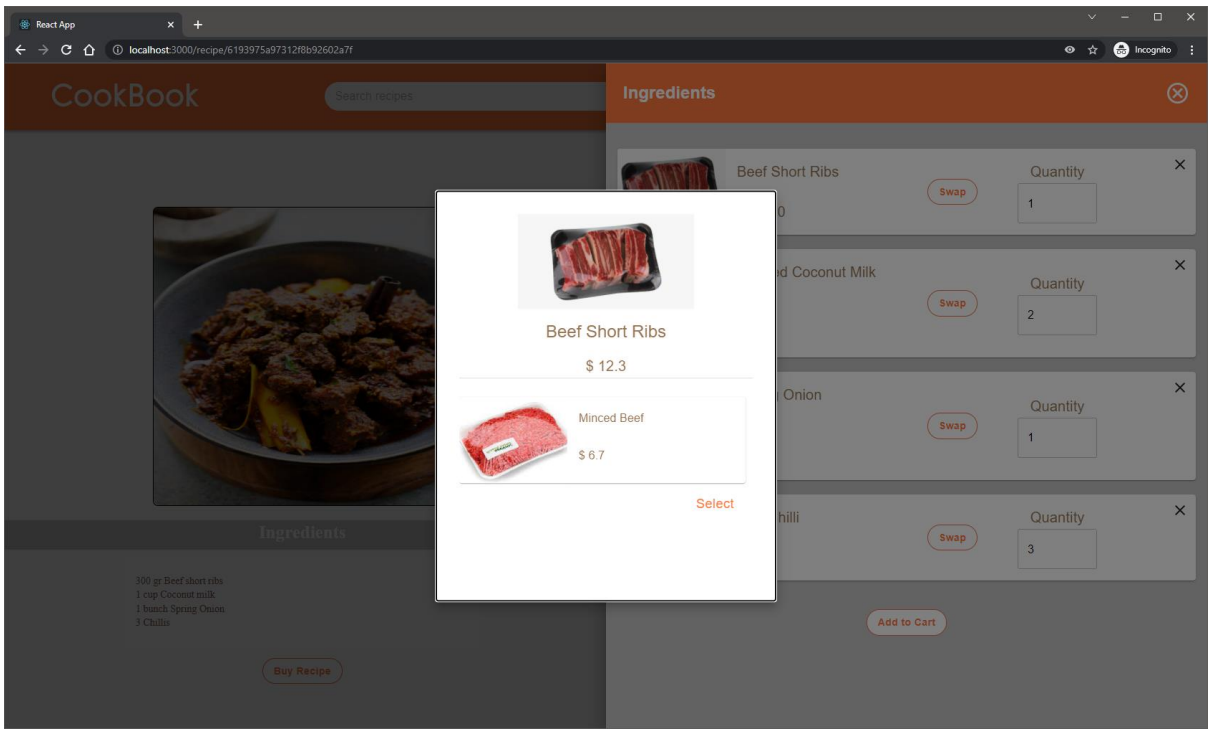

Figure 7: Swap Ingredient Recommender

Purchasing the cart

The contents of the cart can be viewed by clicking the cart symbol on the top right of the menu bar. The quantity of the items added to the cart can also be edited again. The cart items will be filtered per recipe for the user's reference. The user can click on 'Proceed to Checkout' (Figure 8) to make the payment (Figure 9) using PayPal or Rewards Cash. If paying using PayPal, Figure 10 will appear, and prompt the user for PayPal information. An email confirmation of the order will be sent to the user (Figure 11).

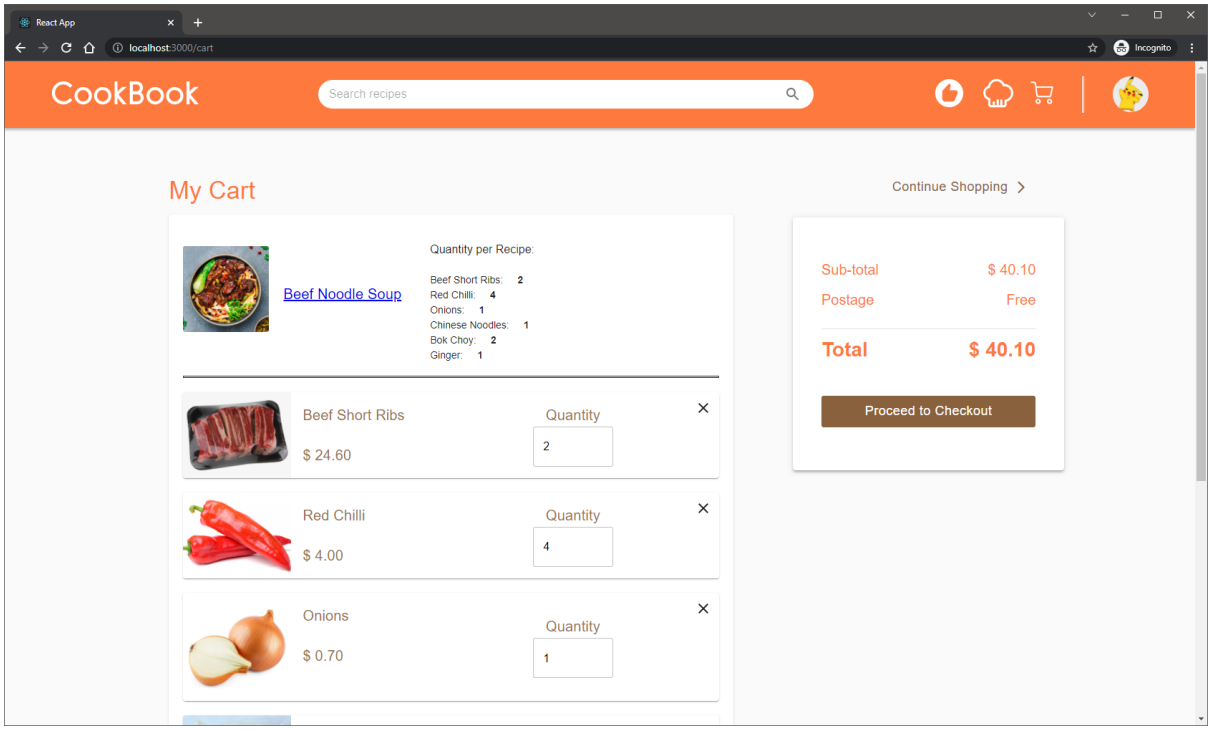

Figure 8: My Cart

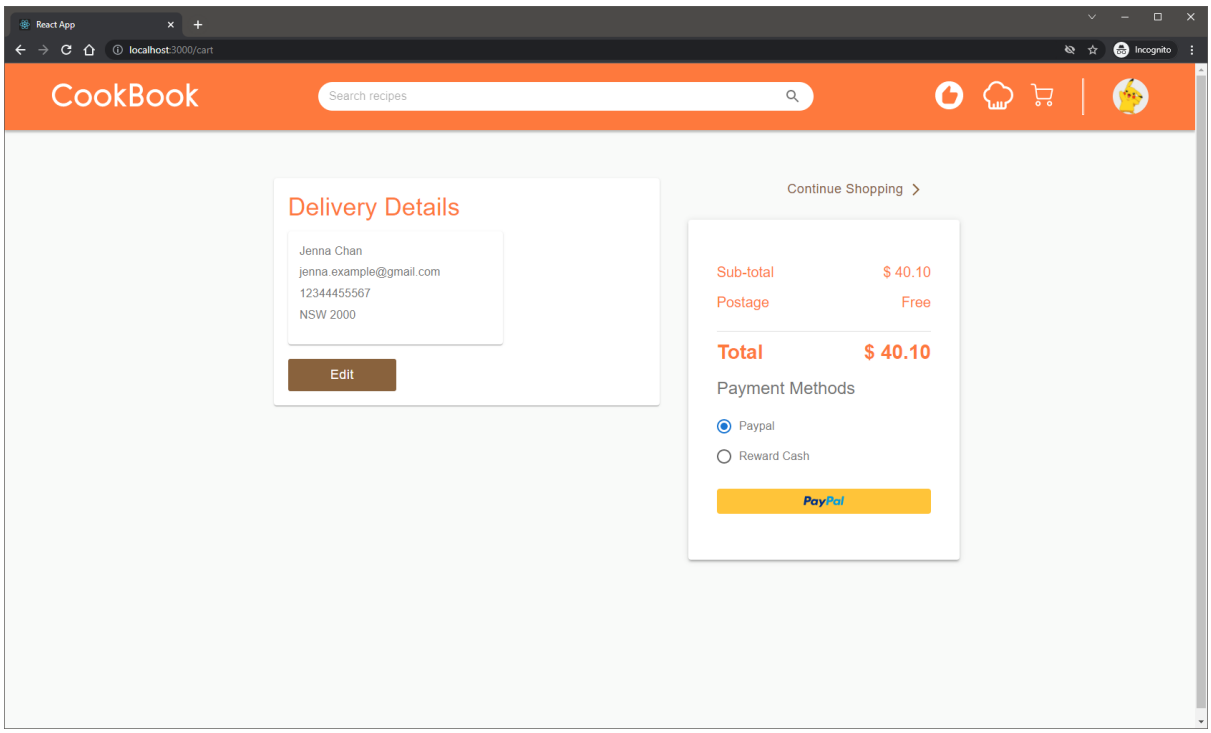

Figure 9: Cart Payment

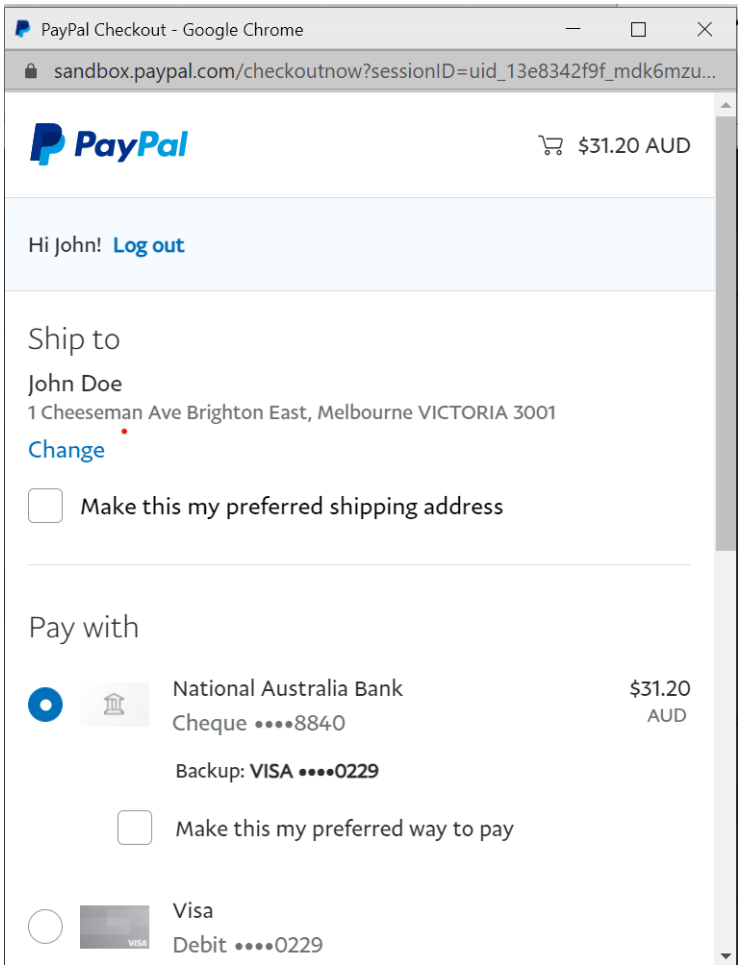

Figure 10: PayPal

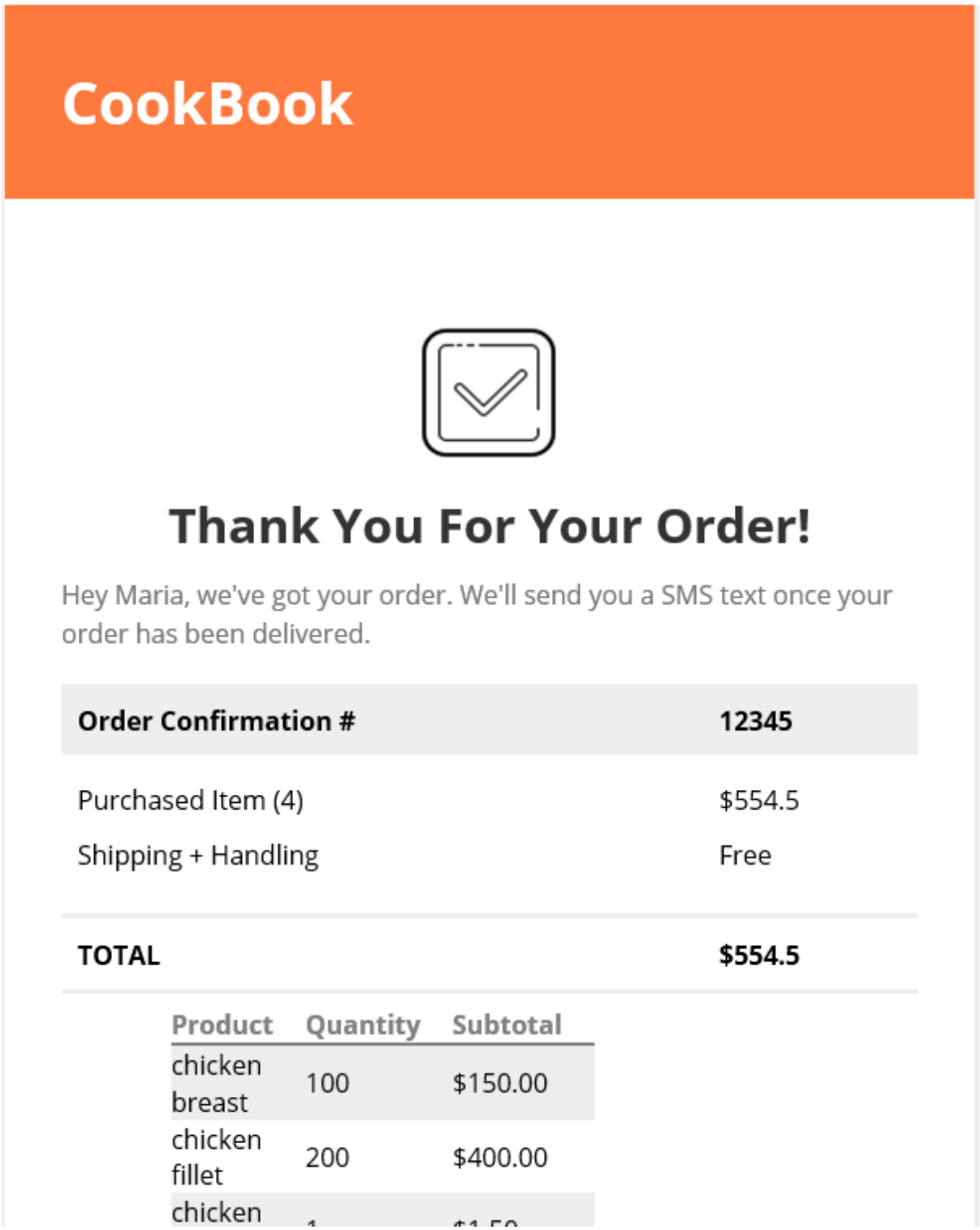

Figure 11: Email Confirmation

Commenting/ rating recipes

Further down on the recipe page shows reviews made by users relating to the recipe (Figure 12). The user can also add their own comments using the 'Add Comment' button.

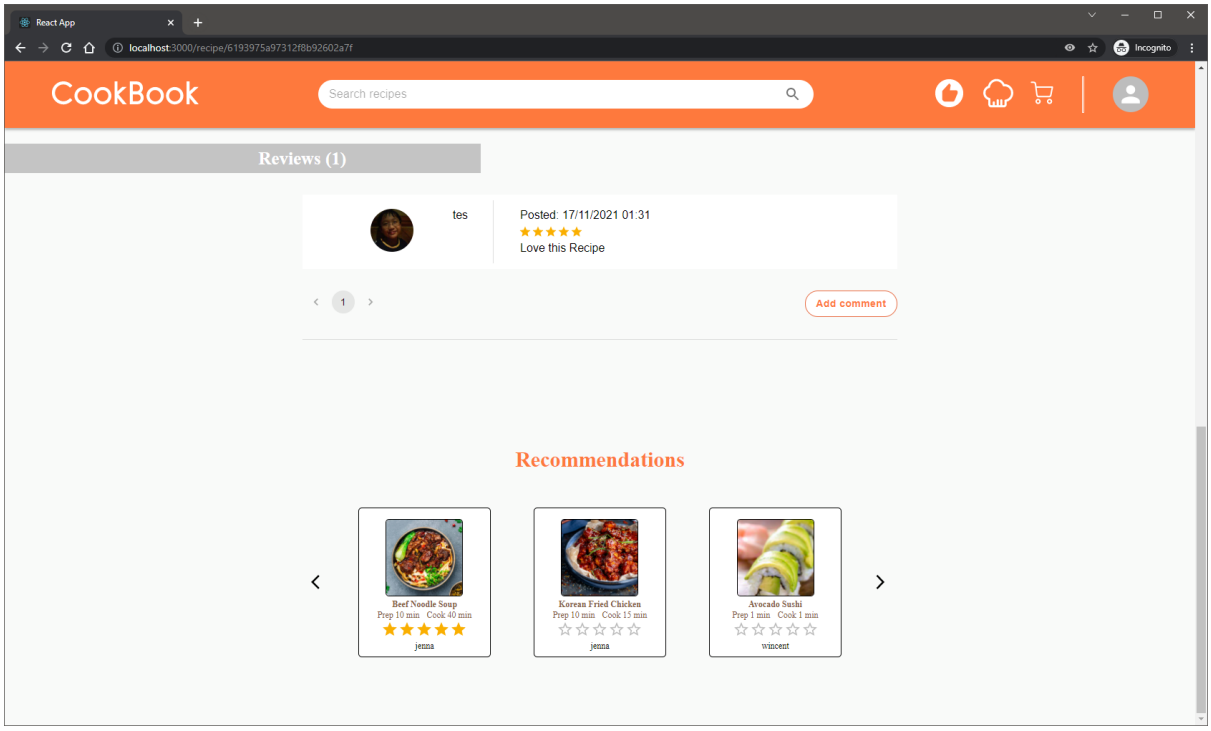

Figure 12: Comments/ Recommendations

Reward points

Users are awarded points when other users purchase from their recipes. They can view their points (if any) by selecting the profile picture in the top right corner. This is shown in Figure 13.

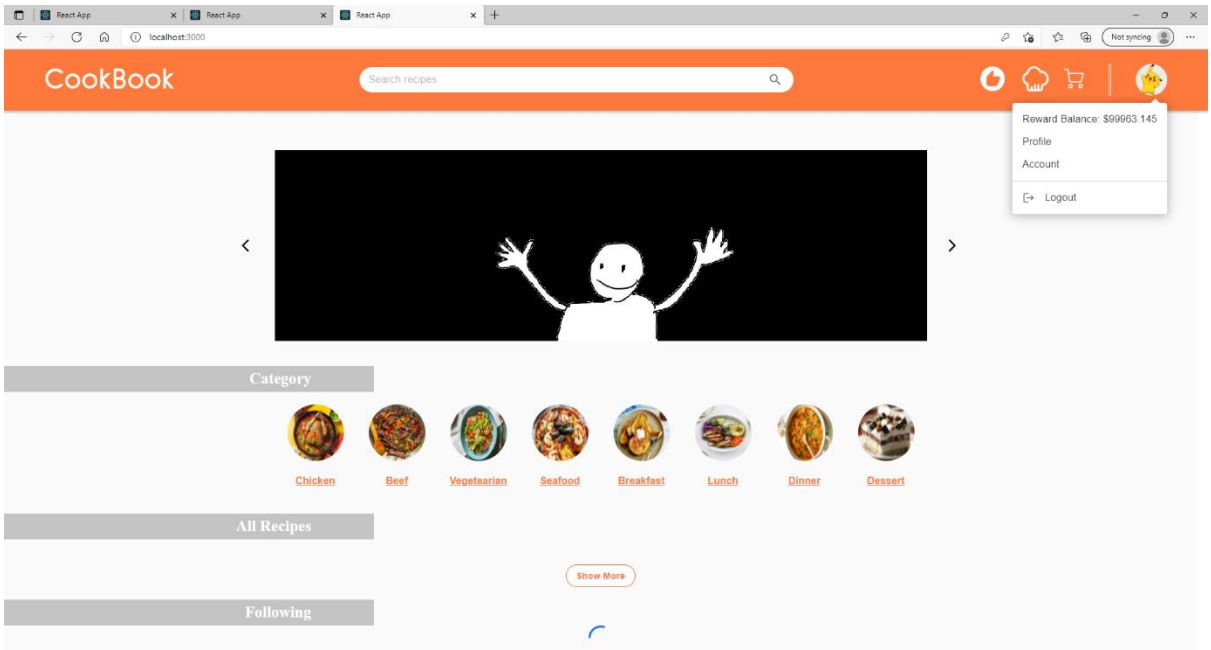

Figure 13: Reward Balance

Search Functions

The search bar at the top can also be used to find recipes by key. Figure 14 is an example of the search result ‘Chicken’. The user may also click on the categories on the front page. This will filter results similarly to the search bar.

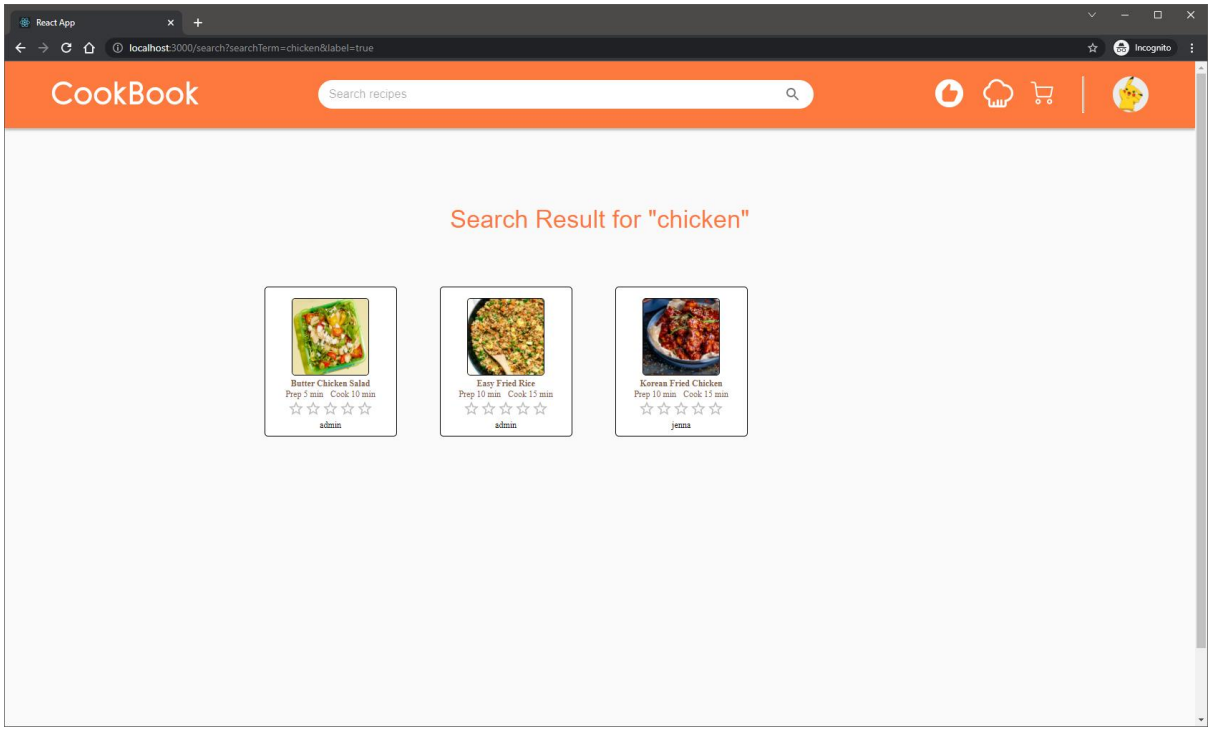

Figure 14: Search for Recipe

Viewing Recommended Recipes

Recipes are recommended in various locations across the site. Figure 12 show recommendations based on similarity to the current recipe. Recommended recipes based on history can also be seen on the home page. By clicking the thumb icon in the top right menu bar, the user is able to answer questions to receive recipe recommendations (Figure 15).

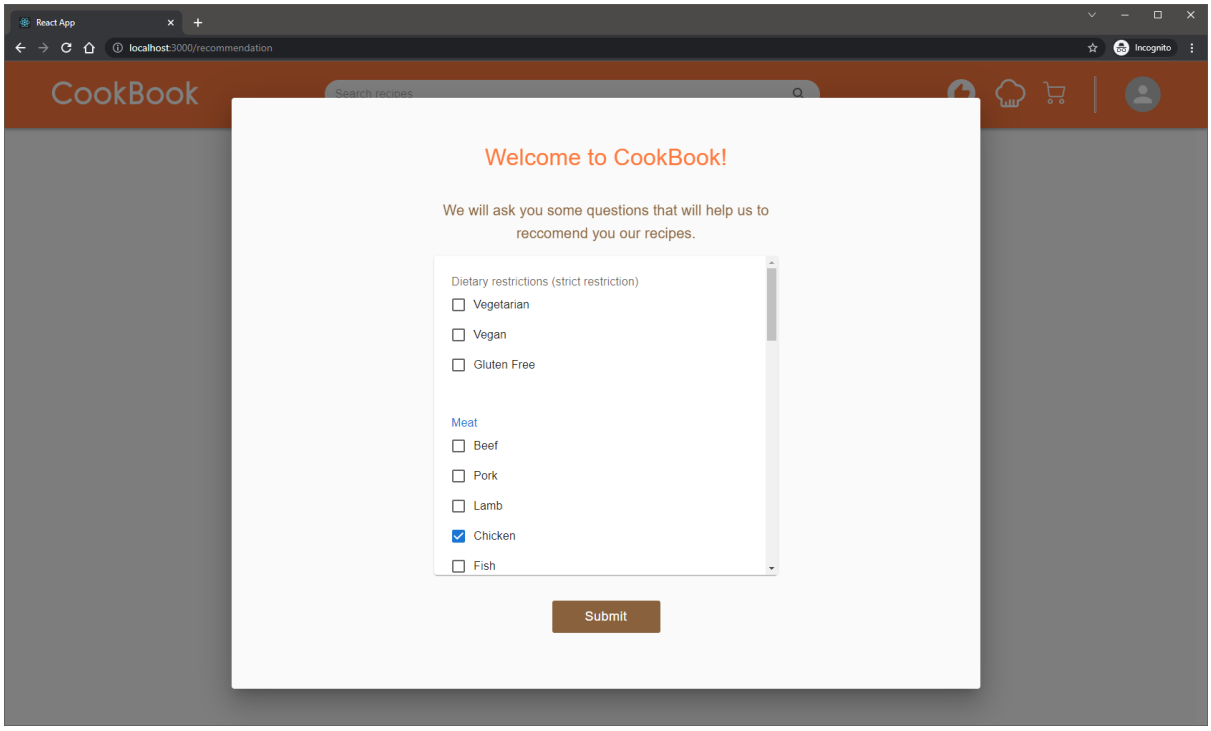

Figure 15: Recommender Questions

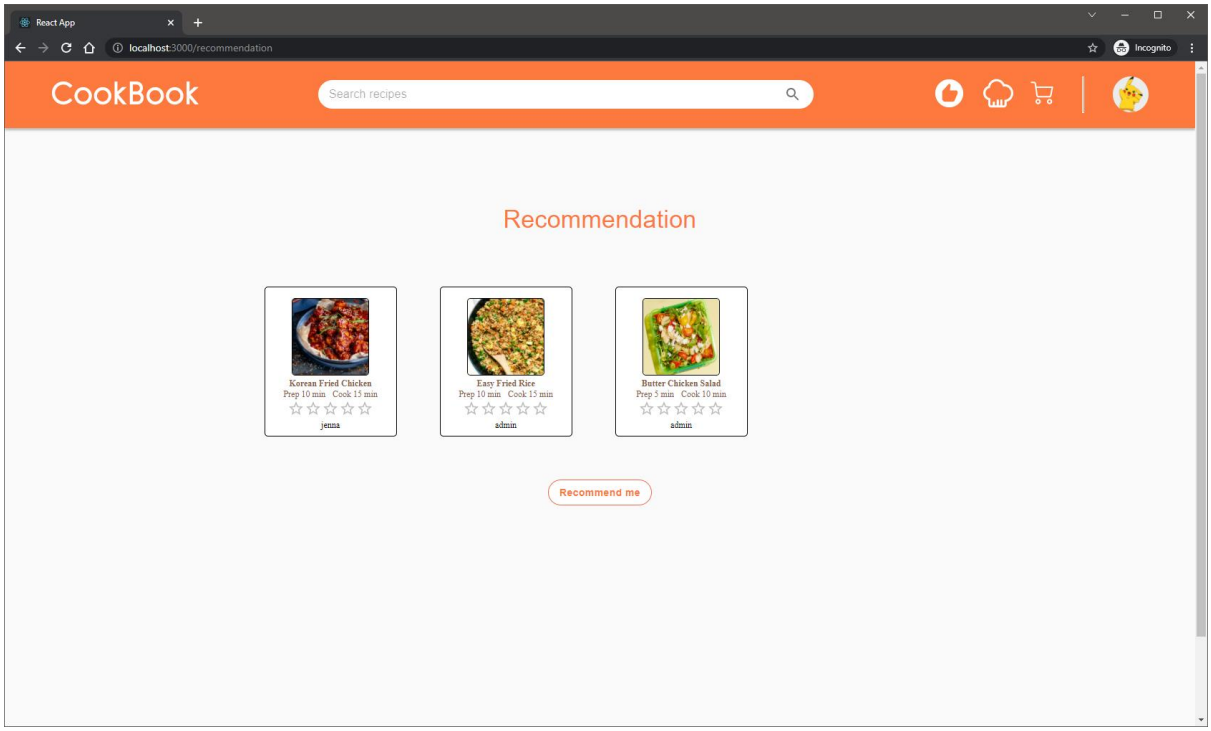

Figure 16: Recommended Recipes

Uploading Recipes

A user can share/ upload their own recipes to the CookBook database. This is done by pressing the chef hat icon on the top right of the menu bar. Figure 17 shows the fields which need to be filled in. Figure 18 shows how ingredients can be linked the recipe, with customised quantity. Recipe steps will also need to be added before saving.

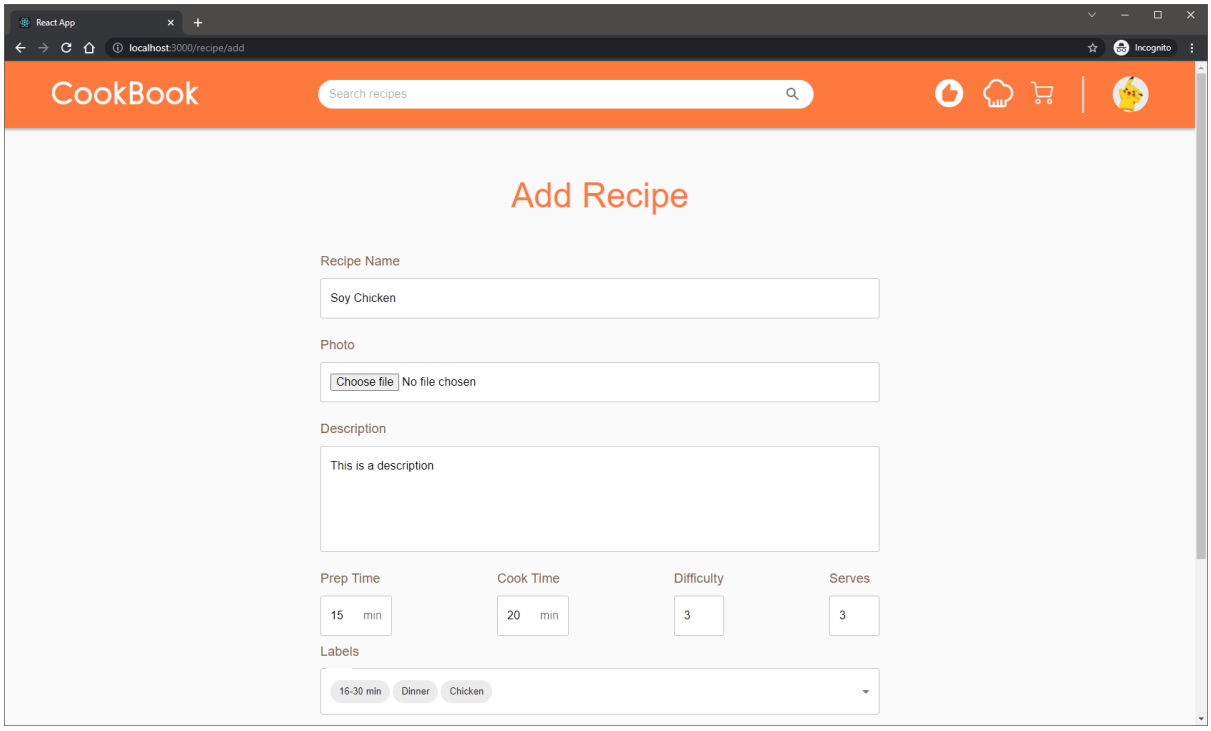

Figure 17: Add Recipe

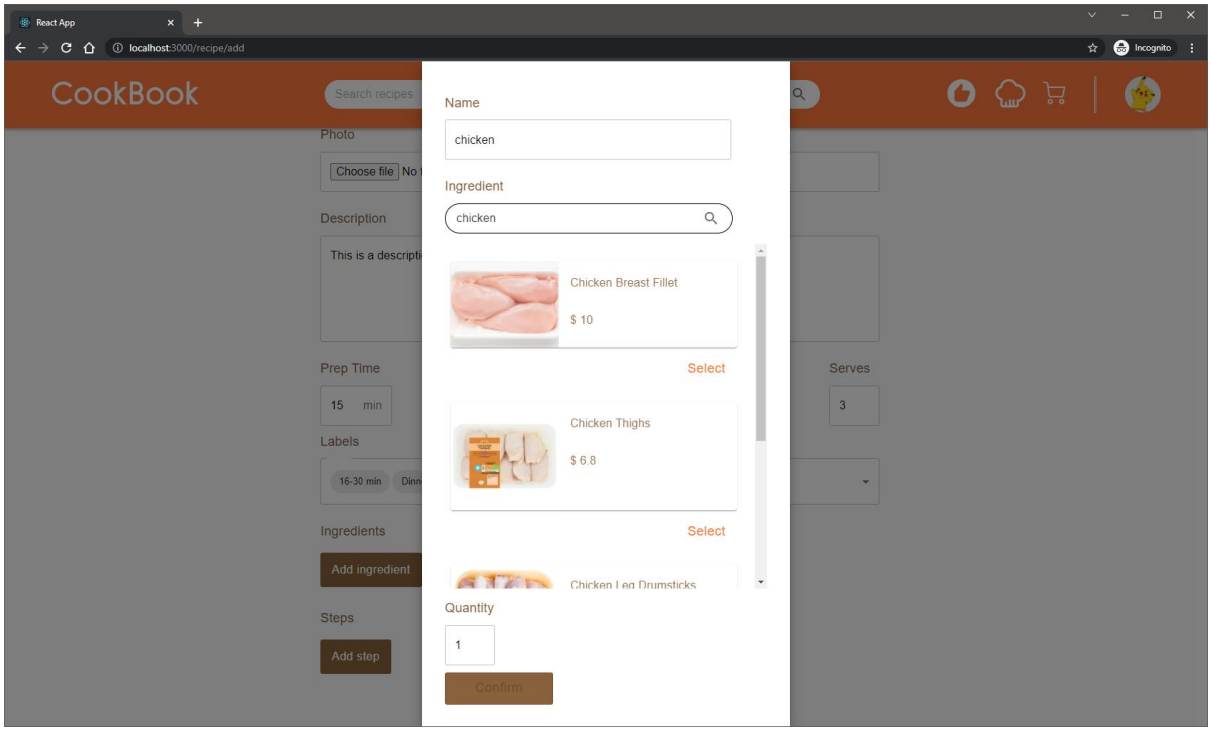

Figure 18: Add Ingredient to Recipe

Editing/ Deleting Recipes

A registered user may edit and delete their own recipes. This can be done on the recipes dashboard found by clicking on the profile image on the top right, followed by account. Selecting ‘Recipes’ on the

left menu shows all uploaded recipes by the logged in user (Figure 19). Each recipe can be edited (Figure 20) or deleted.

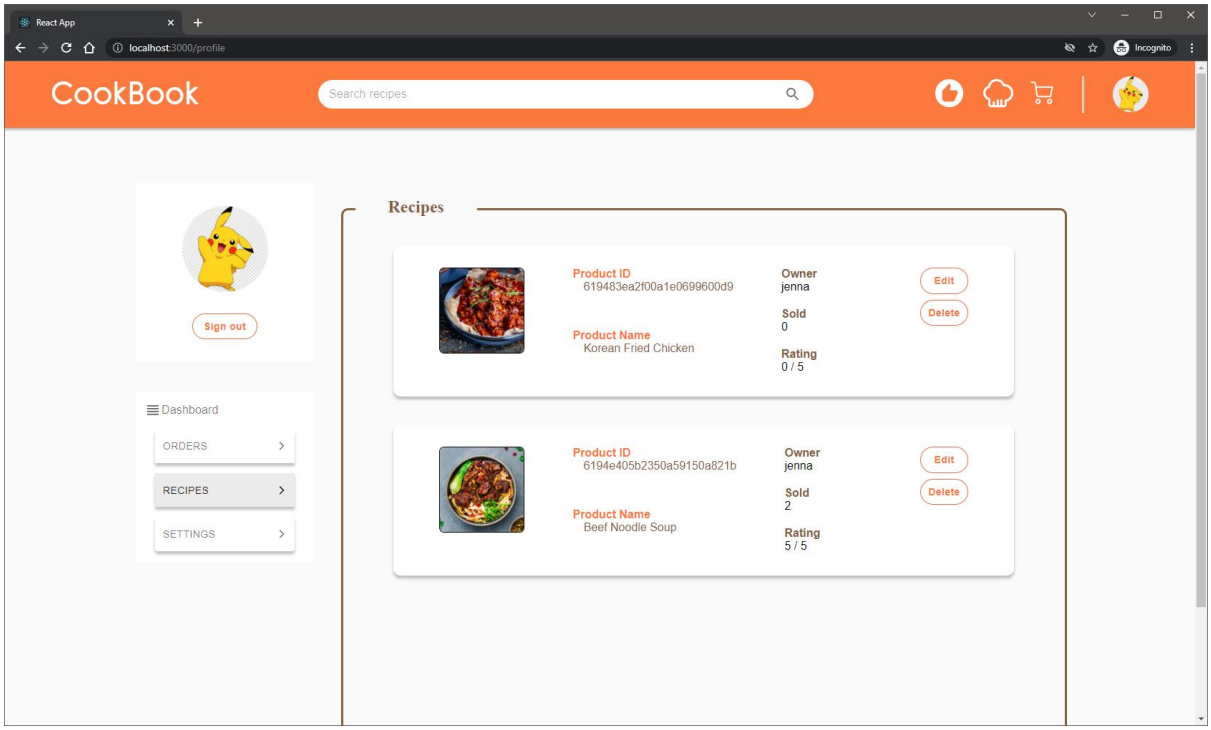

Figure 19: Recipe Dashboard

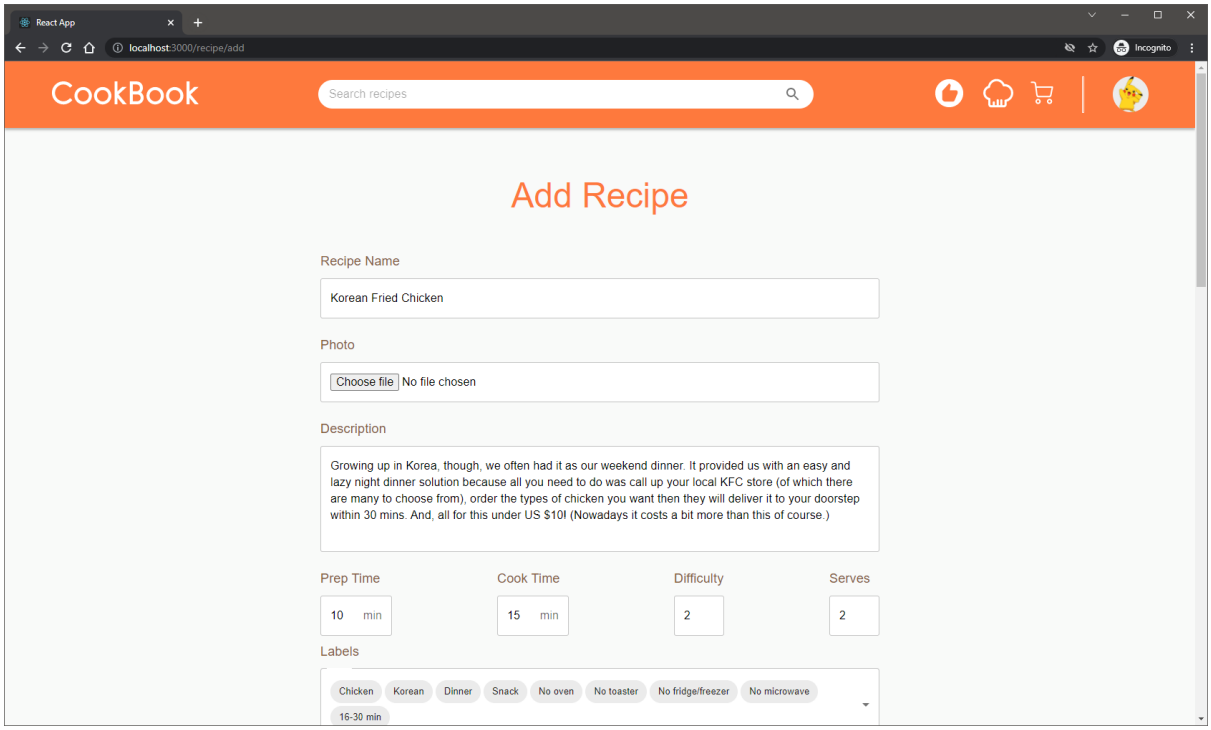

Figure 20: Edit Recipe

Profile Pages

The profile's page can be found by clicking on the profile picture in the top right corner and then clicking 'Profile'. This will show the logged in user's profile, their recommended recipes and other information. Users can also follow other user accounts to follow what recipes they have uploaded (Figure 22).

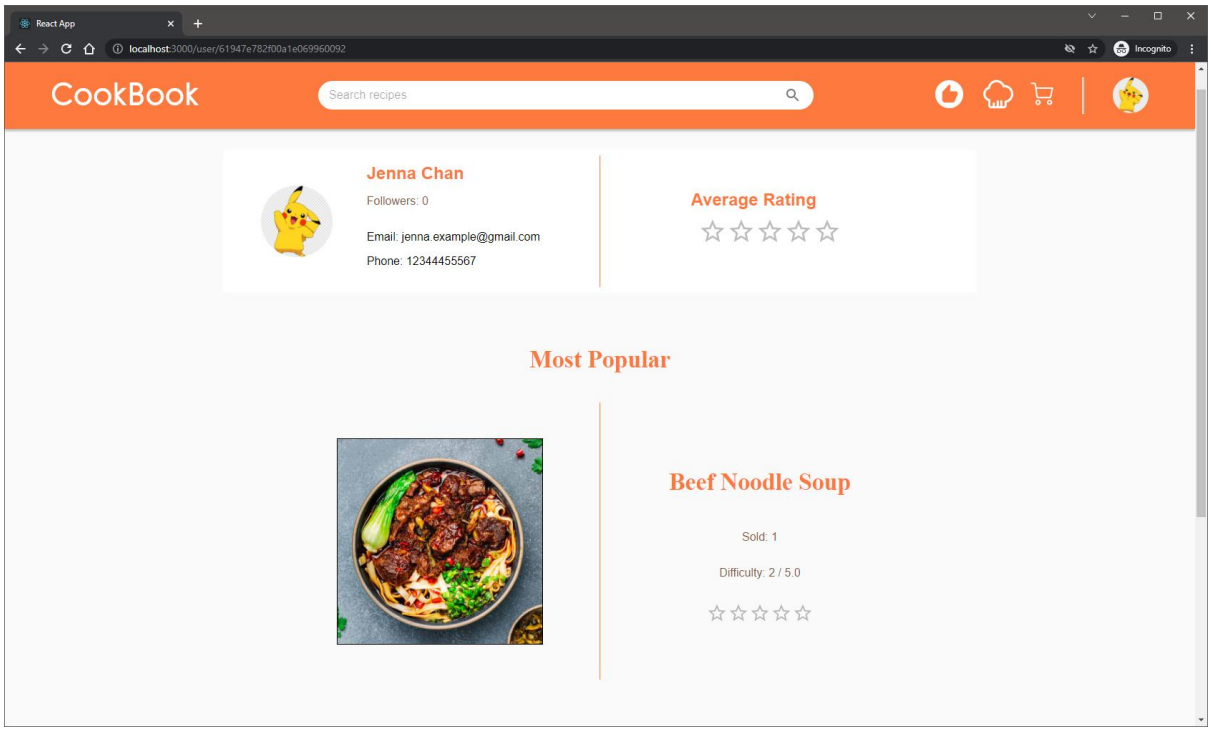

Figure 21: Profile View

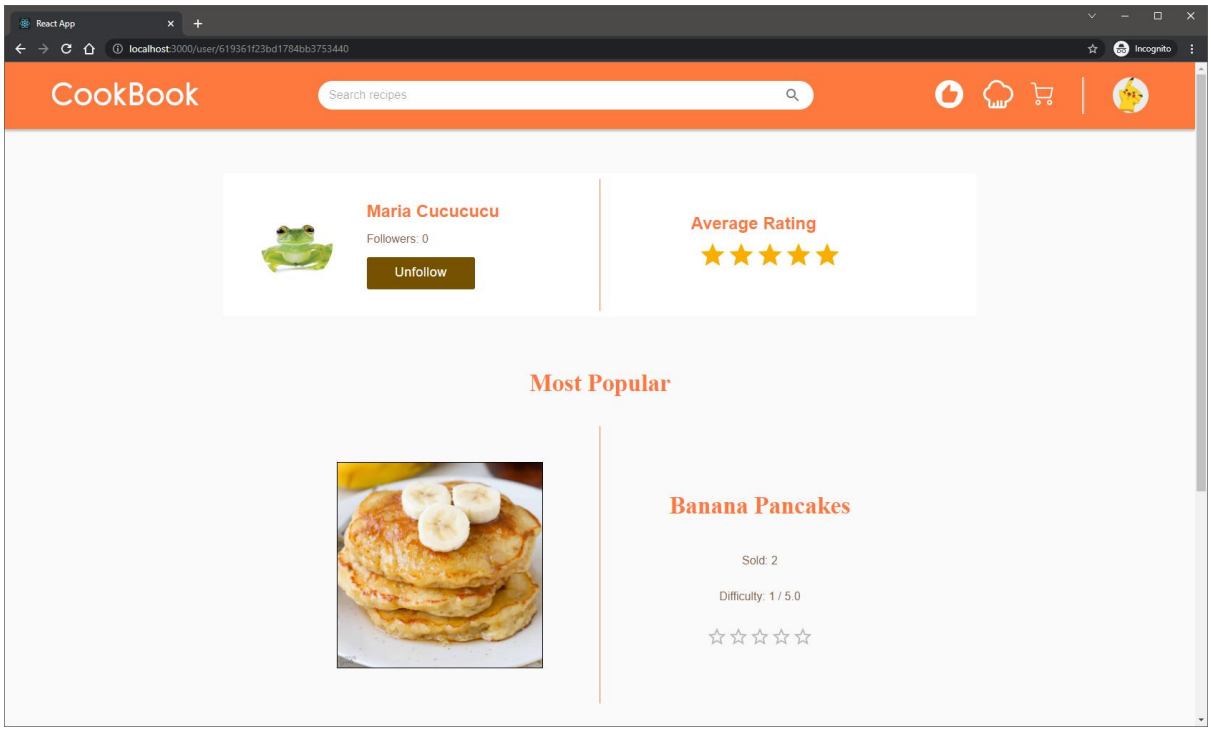

Figure 22: Follow Profile

Account Dashboard

From the account dashboard, details of current and previous orders can be shown. These views can be changed using the tabs at the top: ‘processing’, ‘dispatched’ or ‘delivered’ as seem in Figure 23.

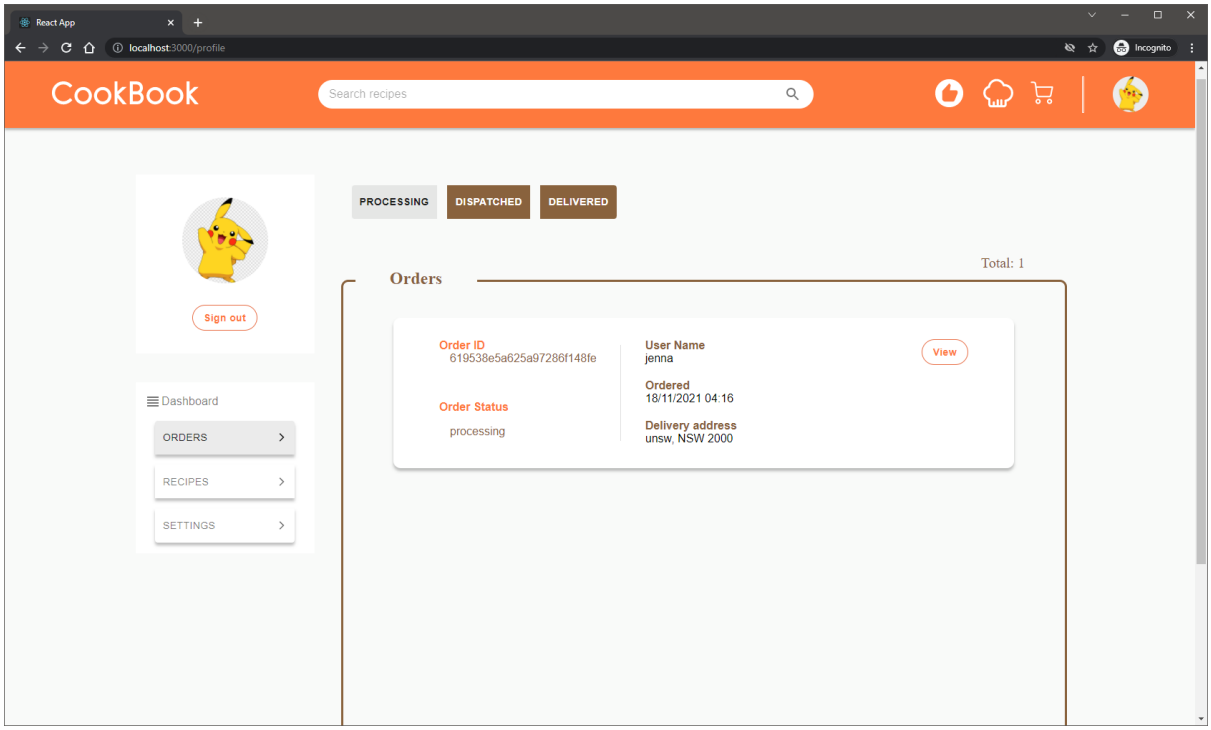

Figure 23: User orders

Profile Settings

Profile settings can be found by clicking on the settings dashboard in account (after clicking profile picture). A user can change their name, profile image, and other personal details (Figure 24).

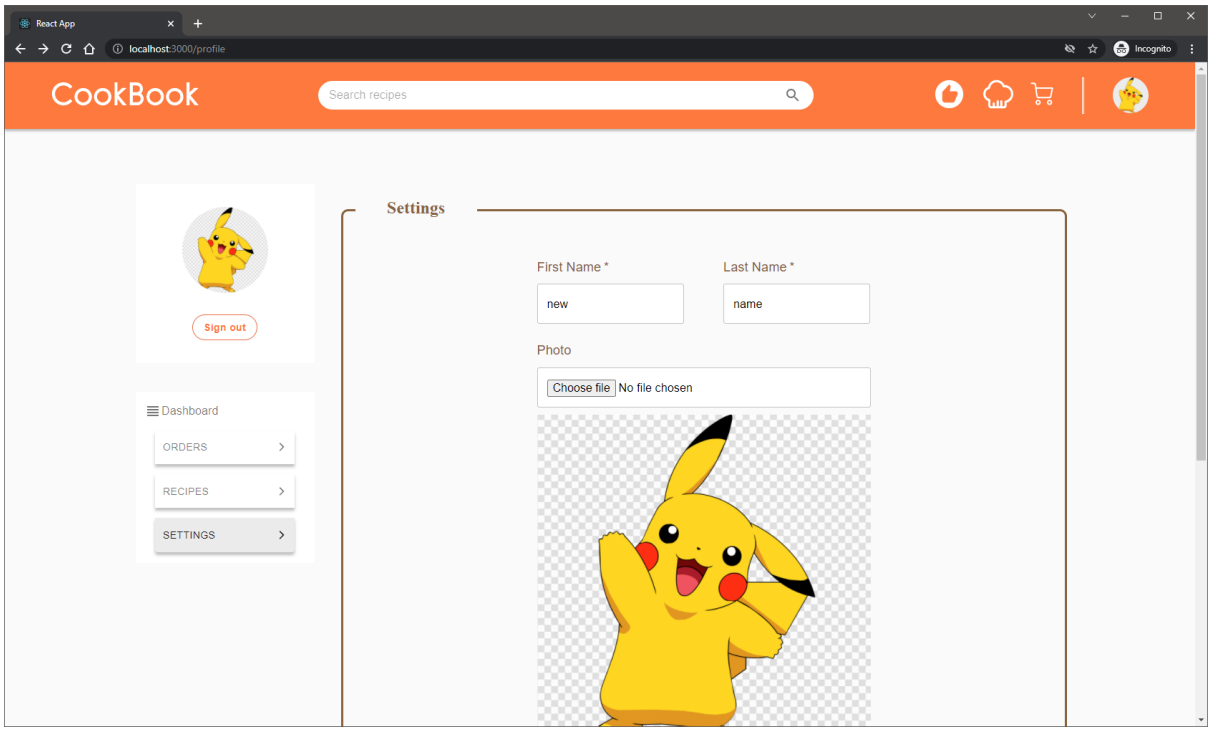

Figure 24: Profile Settings

Logging out

A registered user can log out by pressing on the profile picture in the right corner, and pressing log out (Figure 25).

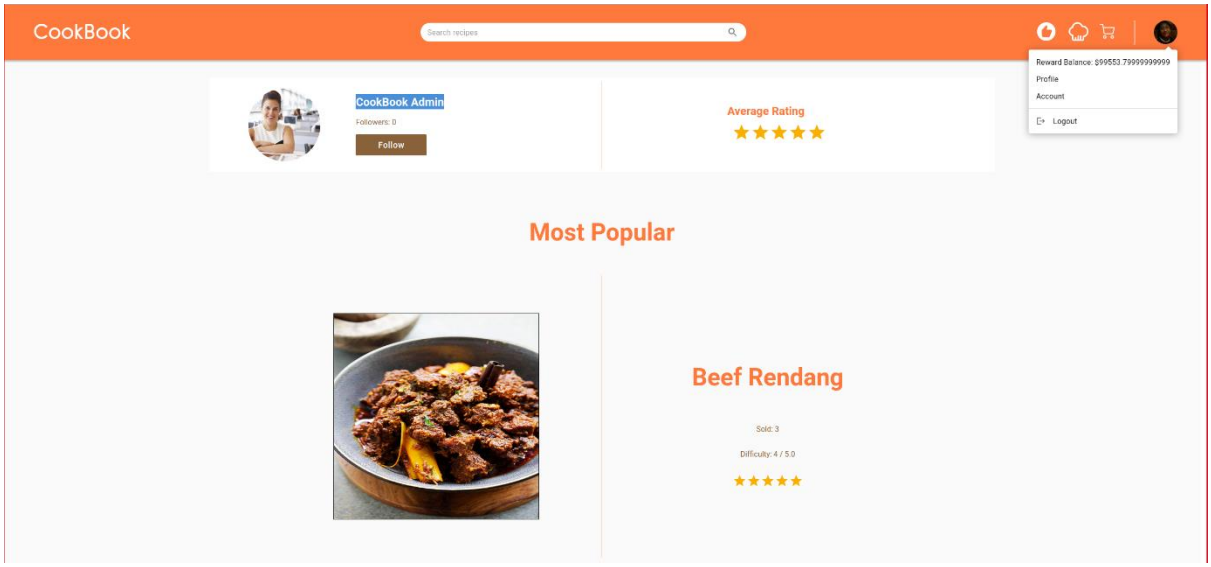

Figure 25: Logout

User Manual – Admin Accounts

The below functions show Admin only features. Admins can also utilise registered user functionality

Adding Product

Admins can add products to the database by clicking on their profile image in the top right followed by ‘Add product’. The admin will then add all relevant details and clicking ‘add product’ to confirm ().

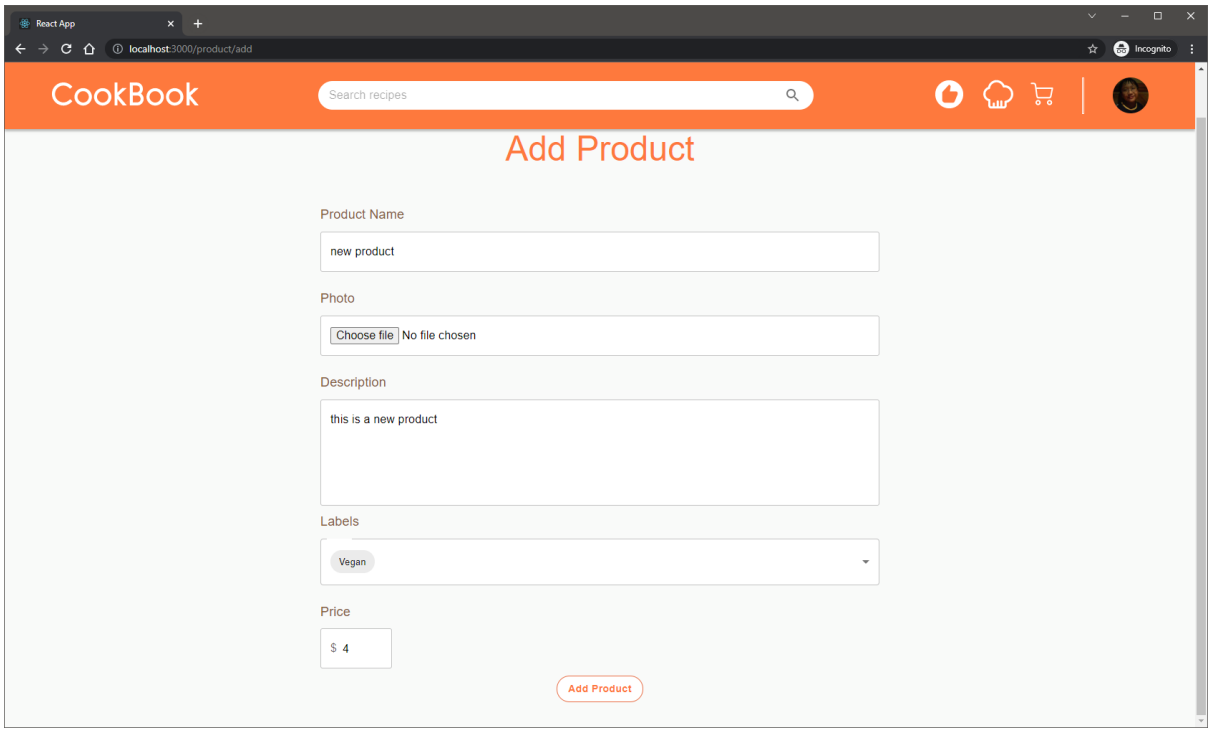

Figure 26: Admin Add Product

Editing Products

Admins can edit products by clicking on their profile image, ‘Account’ and then using the ‘Products’ dashboard on the left menu. Each item can be edited individually by clicking on the associated button (Figure 27).

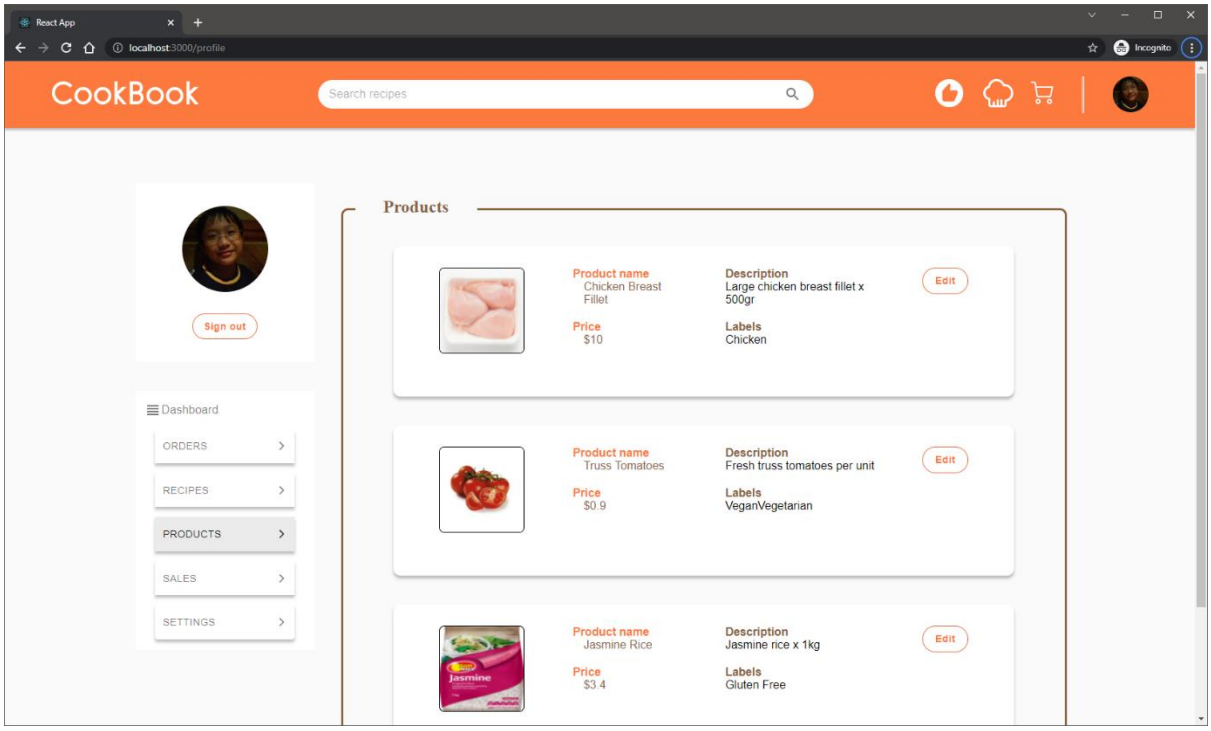

Figure 27: Edit Product

Updating Orders

Admins can manage and update orders made by all customers. They can find the page in Figure 28 by clicking on the profile picture, Account, and then the ‘Orders’ dashboard on the left. Admins can change the order status by using the drop down menu for each order.

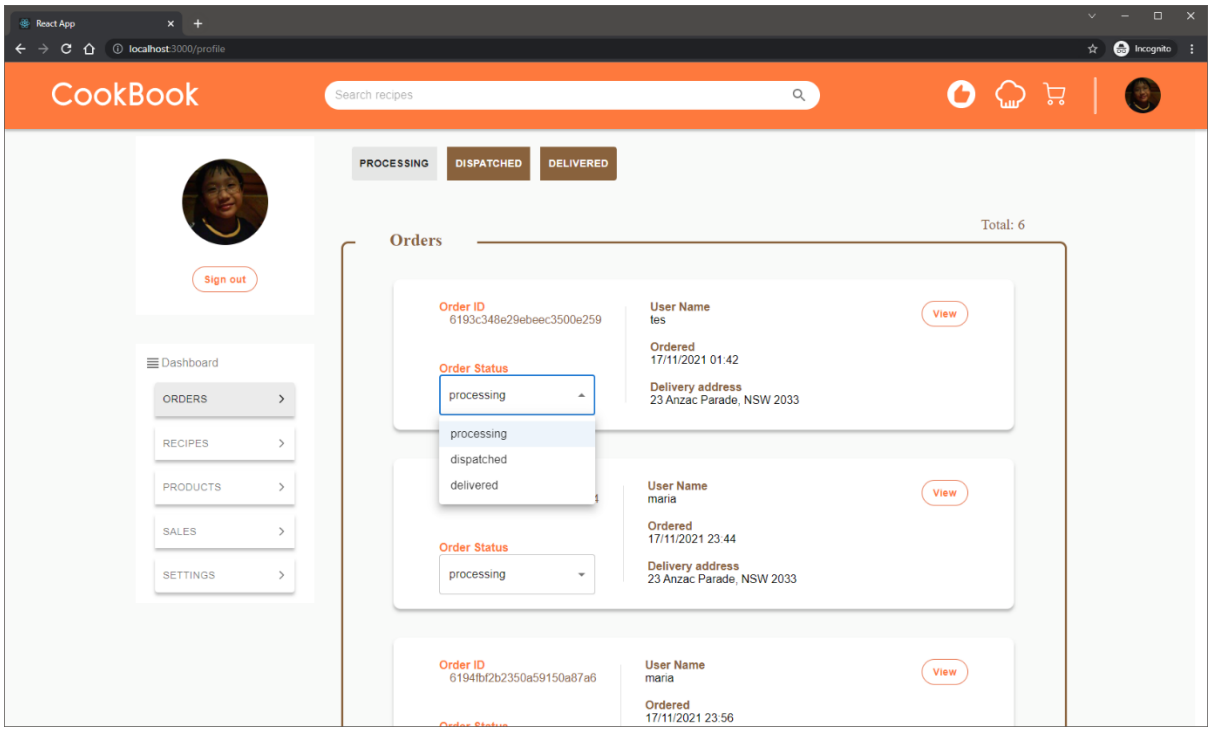

Figure 28: Manage Orders

Viewing Sales Reports

Admins also have the functionality to view sales information for products in the database. This can be found by clicking on user profile image, ‘account’ and using the sales dashboard on the left menu. If required, date parameters can be filled or sales by product can also be searched (Figure 29).

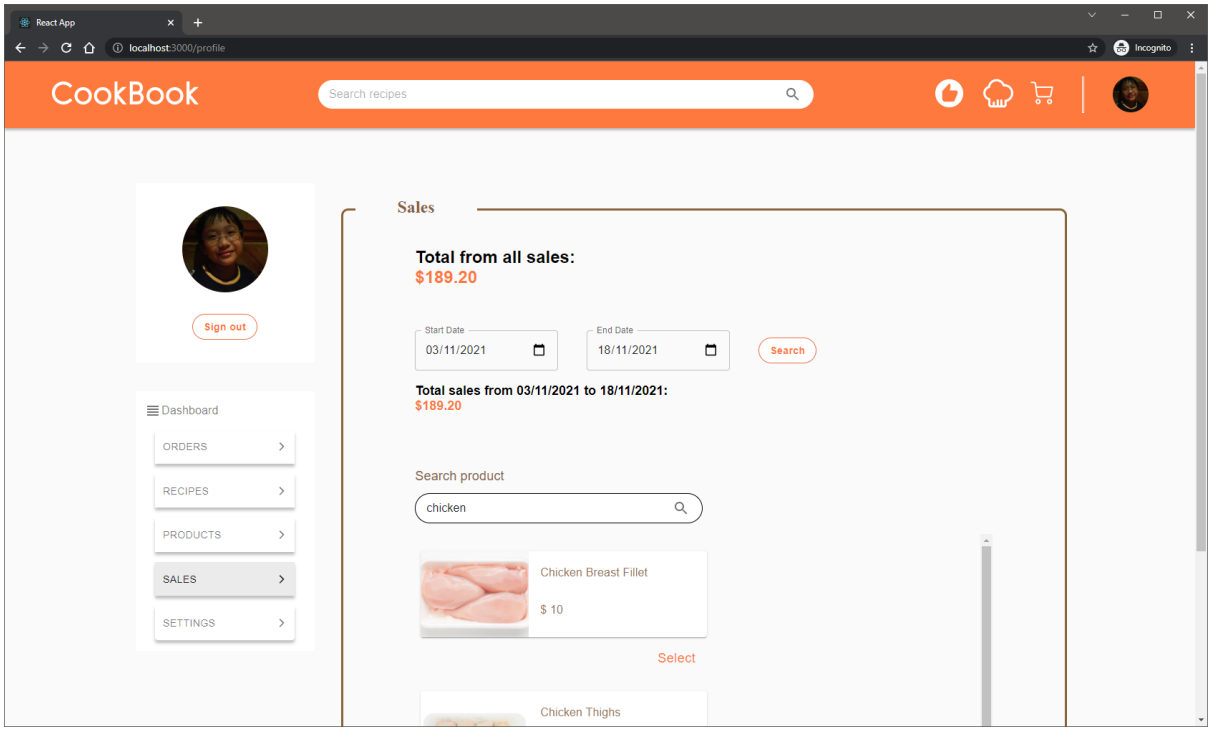

Figure 29: Sales Reports

References

Atlassian, 2021. *COMP3900-H18B-Internship404.* [Online]
Available at: https://comp3900-h18b-internship404.atlassian.net/jira/software/projects/CHI/boards/1/backlog?selectedIssue=CHI-14
[Accessed 03 10 2021].

EcommerceDB, 2021. *The eCommerce market in Australia.* [Online]
Available at: https://ecommercedb.com/en/markets/au/all
[Accessed 20 09 2021].

Facebook Inc, 2021. *React.* [Online]
Available at: https://reactjs.org/
[Accessed 17 11 2021].

Flask, 2021. *Flask-CORS.* [Online]
Available at: https://flask-cors.readthedocs.io/en/latest/
[Accessed 18 11 2021].

Hello Fresh, 2021. *Hello Fresh.* [Online]
Available at: https://www.hellofresh.com.au/
[Accessed 23 09 2021].

Ibis World, 2021. *Online Grocery Sales in Australia - Market Research Report.* [Online]
Available at: https://www.ibisworld.com/au/industry/online-grocery-sales/5527/
[Accessed 20 09 2021].

MaterialUI, 2021. *Material UI.* [Online]
Available at: https://mui.com/
[Accessed 18 11 2021].

Mongo DB, 2021. *PyMongo.* [Online]
Available at: https://pymongo.readthedocs.io/en/stable/
[Accessed 18 11 2021].

MongoDB, 2021. *MongoDB.* [Online]
Available at: https://www.mongodb.com/
[Accessed 17 11 2021].

npm, 2021. *Axios.* [Online]
Available at: https://www.npmjs.com/package/axios
[Accessed 18 11 2021].

Pallets, 2010. *Flask.* [Online]
Available at: https://flask.palletsprojects.com/en/2.0.x/
[Accessed 18 11 2021].

PayPal, 2021. *SandBox Accounts.* [Online]
[Accessed 12 11].

Python, 2021. *dnspython2.1.0.* [Online]
Available at: https://pypi.org/project/dnspython/
[Accessed 18 11 2021].

Python, 2021. *pretty_html_table.* [Online]
Available at: https://pypi.org/project/pretty-html-table/
[Accessed 18 11 2021].

Python, 2021. *PyJWT.* [Online]
Available at: https://pyjwt.readthedocs.io/en/stable/
[Accessed 18 11 2021].

Woolworths, 2021. *Recipes.* [Online]
Available at: https://www.woolworths.com.au/shop/recipes),
[Accessed 23 09 2021].