Institiúid Teicneolaíochta Cheatharlach

INSTITUTE of TECHNOLOGY CARLOW

At the Heart of South Leinster

4th Year Final Project - Functional Specification

BSc (Honours) Software Development

Name: Alex Matthews
Student ID: C00208942
Supervisor: Dr. Oisin Cawley

Contents

1. Abstract 3
2. Introduction 4
3. Problem Statement 5
4. Use Case Diagram 6
5. Brief Use Cases 7
5.1. Add Diary Entry 7
5.2. Edit Diary Entry 7
5.3. Delete Diary Entry 7
5.4. Add Recipe 7
5.5. Edit Recipe 7
5.6. Delete Recipe 8
5.7. Share Recipe 8
5.8. Search Recipes 8
6. Functionality Description 9
7. UI Mock-ups 10
7.1. Food Diary 10
7.2. Recipes 13
8. FURPS+ 21
Functionality 21
Usability 21
Reliability 21
Performance 21
Supportability 21
+ 21

1. Abstract

The purpose of this document is to provide an account of the functionality of a food diary and recipes application for teenagers with coeliac disease. The document will outline the main functionalities of the system and the reasoning behind picking each functionality. Some of the material documented may not be included in the final application, as the it will be created using an agile development methodology.

2. Introduction

The application being developed is a food diary and recipe book which is aimed at teenagers with coeliac disease. The application is a stand-alone module which will be integrated with other modules which are being developed by multiple other students. At the end of the project, each of the modules will be put together to create a full application for teenagers with coeliac disease. The aim of the module being developed for this project is to increase the diet compliance rate among teenagers with coeliac disease. This document will outline all of the functionalities that should be included in the final application. The document will include use case diagrams, brief and detailed use cases, wireframes and a description of the user interface, scenarios, and functional and non-functional requirements.

The use cases will specify the main functionality of the system and the detailed use cases will drill down into each use case by outlining the steps that must be taken for the use case and the alternative steps. The wireframe will give a clear idea of how the final application will look. With future iterations, the appearance of the application could change over time as requirements become clearer. The document will also outline some typical scenarios for using the application. A scenario is a situation a person might be in where the application can help them out. The functional requirements will outline what the system will do, and the non-functional requirements will state any other requirements not outlined in the functional requirements e.g. speed of a certain function.

3. Problem Statement

As stated in the need finding report for the project, in Europe, 1% - 2% of people have coeliac disease and the incidence of coeliac disease is increasing year-on-year. When patients don't comply with their diet, it causes them to have minimal short-term symptoms, but regular non-compliance can cause severe long term-term symptoms such as fragile bones, autoimmune diseases, infertility problems, poor nutrient absorption, and malignant tumours in the digestive system. This application will aim to provide functionality which will help increase the compliance rate among teenagers and therefore reduce the number of people with coeliac disease contracting any of the aforementioned health conditions.

4. Use Case Diagram

The purpose of this use case diagram is to visualise the main functionalities of the system at a high level. The functionalities outlined here are subject to change due to the agile approach to the development of the application.

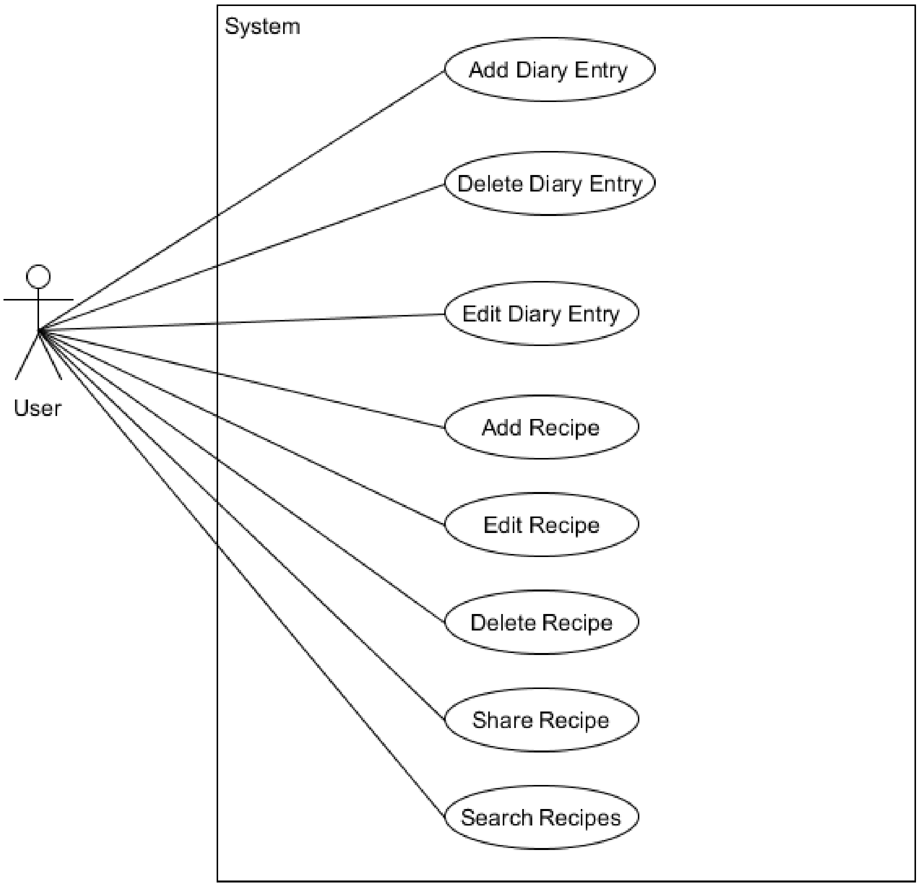


Figure 1 – Use Case Diagram

5. Brief Use Cases

5.1. Add Diary Entry

Name: Add Diary Entry

Actors: User

Description: This use case begins when the user wishes to add an entry to their food diary. The user first selects the day they want to add an entry for and selects the option for adding a new entry. The system prompts the user for the details of the entry including, photo, title, and description. The user enters the relevant information and selects the option to save the entry. The system saves the entry on the day selected by the user.

5.2. Edit Diary Entry

Name: Edit Diary Entry

Actors: User

Description: This use case begins when the user wishes to edit an entry in their food diary. The user first selects the day they want to edit an entry for. They then open the context menu and select the edit option. The user then edits the information presented to them, in any way they see fit. If the user proceeds, the entry will be updated with the new information.

5.3. Delete Diary Entry

Name: Delete Diary Entry

Actors: User

Description: This use case begins when the user wishes to delete an entry from their food diary. The user first selects the day they want to delete an entry from. They open the context menu and select the delete option. The system will ask the user are they sure they want to delete the entry. If the user proceeds, the entry will be deleted from the diary.

5.4. Add Recipe

Name: Add Recipe

Actors: User

Description: This use case begins when the user wishes to add a new recipe. The user first navigates to the recipe section. The user then chooses the type of meal they want to add a recipe for. They then select the option for adding a new recipe. The system will prompt the user for information such as title, cooking time, ingredients, method, and a photo. When the user has entered the relevant information and proceed, the system saves the recipe in the appropriate place.

5.5. Edit Recipe

Name: Edit Recipe

Actors: User

Description: This use case begins when the user wishes to edit an existing recipe. The user navigates to the recipe they want to edit. They then select the option for editing the recipe. When the user updates the recipe information and proceeds, the system will update the recipe record.

5.6. Delete Recipe

Name: Delete Recipe

Actors: User

Description: This use case begins when the user wishes to delete an existing recipe. The user navigates to the recipe they want to delete. They then select the option for deleting the recipe. The system will ask the user are they sure they want to delete the recipe. If the user proceeds, the recipe will be deleted.

5.7. Share Recipe

Name: Share Recipe

Actors: User

Description: This use case begins when the user wishes to share an existing recipe. The user navigates to the recipe they want to share. They then select the option for sharing the recipe. The system will ask the user where they want to share it to. The user selects the appropriate option and proceeds to share the recipe to the application community or to an external application.

5.8. Search Recipes

Name: Search Recipes

Actors: User

Description: This use case begins when the user wishes to search a list of recipes. The user first navigates to the list they want to search. The list could be the users breakfast recipes or community breakfast etc. They then begin to type what they want to search for in the search box. The list will filter as the user types into the box. The user can search the filtered list for the recipe they were looking for.

5.9. Save/Favourite Recipes

Name: Save/Favourite Recipes

Actors: User

Description: This use case begins when the user wishes to save a community recipe. The user first navigates to the recipe they want to save/favourite. The user opens the menu and selects the save/favourite button. The recipe is saved.

6. Functionality Description

The main purpose of this application is to help increase the diet compliance rate among teenagers with coeliac. The application will try to accomplish this by providing users with functionality to log every meal they eat. The goal is to cause teenagers to be more likely to comply with their diet as there will be a visible record of everything they ate, assuming users create an entry for each meal and snack they consume. This is not always feasible due to a multitude of reasons. The driving force to encourage users to create an entry for all meals will be to integrate the food diary with the functionality for checking for checking the results of the Gluten Immunogenic Peptides (GIP) test. The GIP test determines if a patient has recently ingested gluten. The functionality for reading the GIP test is being created by another member of the team.

The goal is to integrate the food diary with GIP test reader so that the user can see a correlation between what they ate and whether there was gluten in their system, or not. If the integration of theses two functions is feasible then the food diary will ask the user to make a prediction of their expected test result. If the user takes the test and the prediction is the same as the test results, they will increase their hit rate. The hit rate can be plotted on a graph against the actual results to give the user a visual representation of how often their prediction is right. The result of the test could be represented on the calendar using different colours. This would give the user a quick overview of how their diet is going. This functionality could be complemented with a reward system. If the user makes a streak of making a right prediction for a certain number of days or weeks, they could be rewarded with badges or achievements. As teenagers are accustomed to having reward systems in application, this seems like the most natural way to engage the users of the application.

The application will also provide a facility for storing, searching, and sharing recipes. This part of the application will allow users to store recipes in a personal recipe list. It will allow users to share their personal recipes with the rest of the community on the application. Users can also save community recipes to their personal recipe list, so they can view them later. The goal of this functionality is to provide the community with a central repository for gluten free recipes. This will give users more possible meals to choose from. The more choice available to coeliac patients, the more chance that they will comply to their diet. Without a large choice of meals to choose from, users are more likely to grow tired of eating the same meals every day and in turn will be more likely to break compliance.

7. UI Mock-ups

Provisional mock-ups for what the application will look like can be seen below. As this project involves multiple developers working on different modules, the user interface on each module might have to be changed in order to create the final application. There are some design students working on the user interface as well. In the future, it might be possible to use their design for the application. As the application is being developed using an agile approach, these screens are subject to change. This section is divided into two sub-sections; food diary and recipes.

7.1. Food Diary

This section of the application is for tracking the user's diet on a day-to-day basis. There are two ideas for the landing screen of this part of the application. The first idea, which can be seen in *Figure 2,* is the original idea for this application. The original idea was to have a timeline which showed the meals on each day or week. It was originally planned that the timeline could be zoomed in and out on to show different views. After some research, it seems this could be a hard feature to implement due to scarce documentation online and the skill ceiling of the developer. This is something that will be prototyped in the future to see if it is feasible. This would be a very useful feature as it would allow users to see their progress over time fast and in an easy to read format. This might increase the chance that people will continue to use the application. The alternative to the timeline is to have a simple list view for each meal on a particular date. This is the most natural alternative as most mobile applications are made up of lists. The list view can be seen in *Figure 3. Figure 4* shows the calendar that is shown when the "Pick Date" button is tapped.

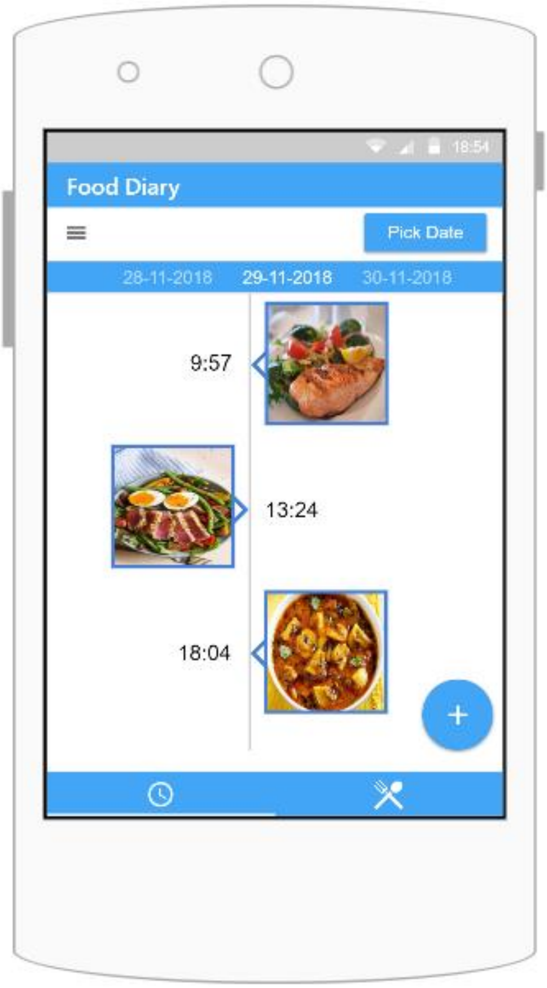


Figure 2 – Timeline Landing Screen

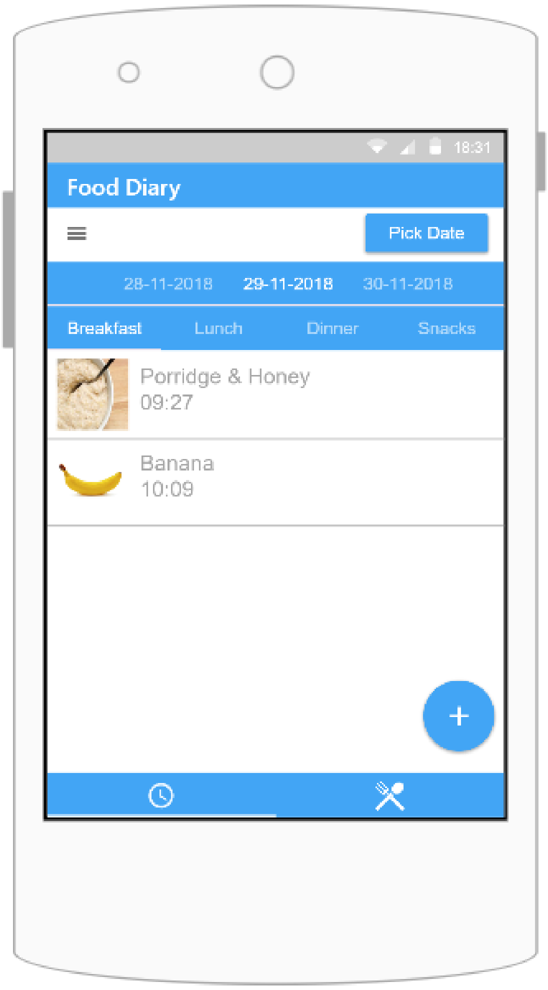


Figure 3 – List Landing Screen

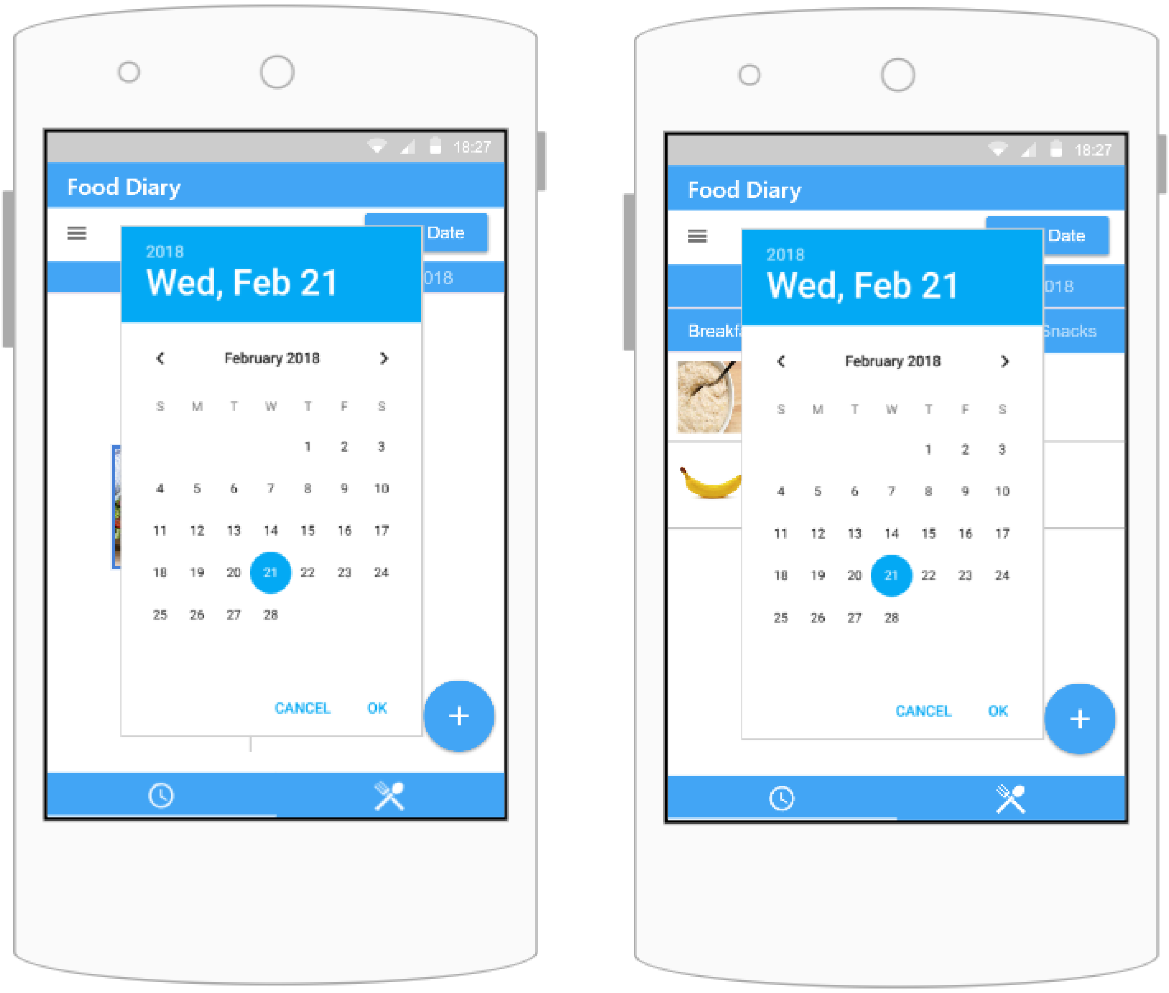


Figure 4 – Choose Food Diary Date

The screen in *Figure 5* shows the add item screen. This screen allows the user to log a meal or a snack. They can press the photo button to take a photo of their meal. This photo can be seen on the timeline/list. They can also enter the title of the entry and a description. The title is a mandatory field but the description is optional, so it's up to the user whether they want to add one or not. *Figure 6* shows the pop-up menu when the user long presses on a diary entry. They are presented with an option to edit. This option allows the user to make changes to a previously entered diary entry. They can change the photo, title or description. They also have the option to delete and entry from the diary entirely. They can also cancel the pop-up menu.

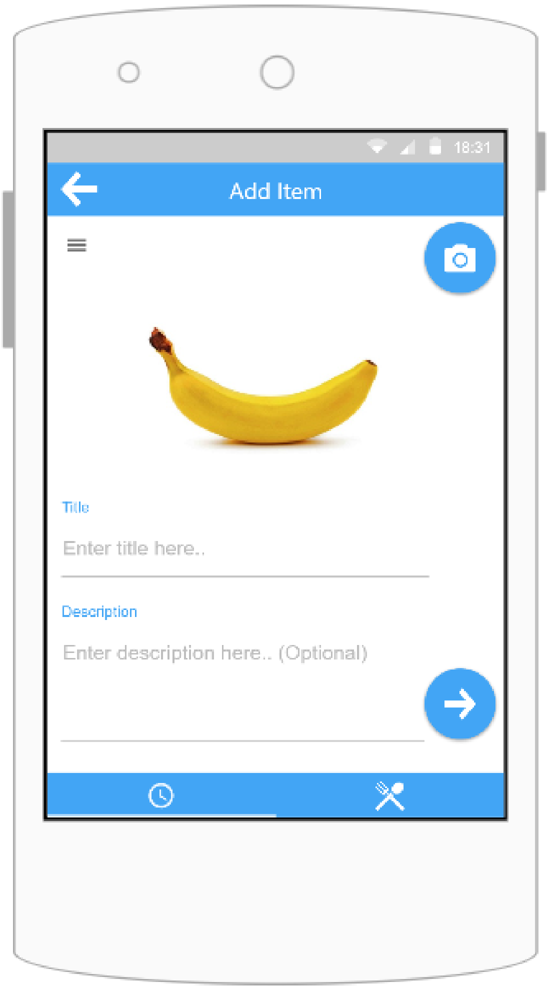


Figure 5 – Add Diary Entry

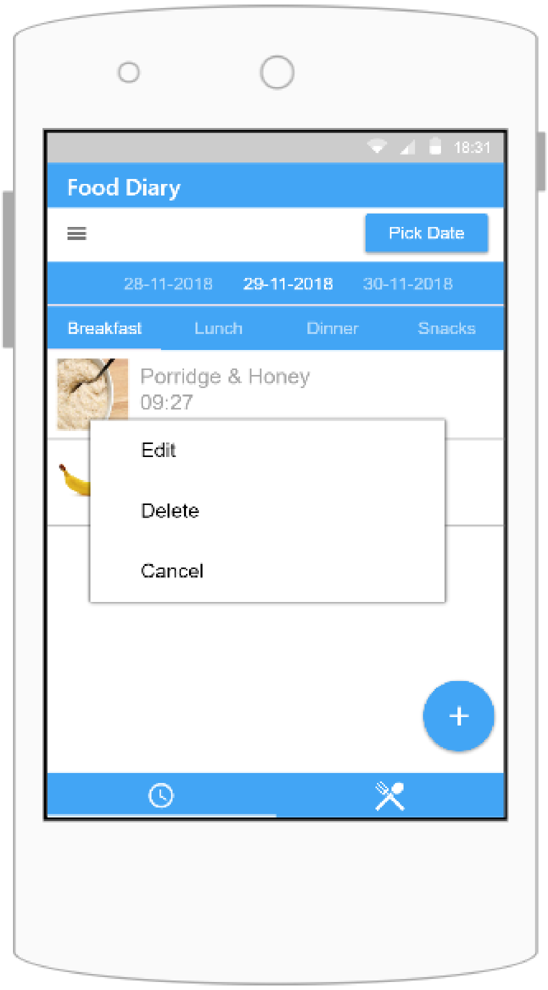


Figure 6 – Food Diary Pop-up Menu

7.2. Recipes

This part of the application is to allow users to keep track of recipes, share them with the community, and have access to recipes shared by other users. When the user first presses the recipes tab (fork and spoon symbol), they are brought to their recipes. Recipes are divided into meals, identically to the way the food diary is laid out. This is to allow users to more easily search for meals. In the final application, there could also be an option included which will allow users to see lists of recipes based on cuisine type or cooking time. Whether this feature is implemented is dependent on time constraints etc. *Figure 7* shows the "My Recipes" section for both breakfast and dinner. The search box is at the top of the list and allows users to quickly filter by keyword. As the user types, the list will get smaller and smaller until the user finds what they are looking for.

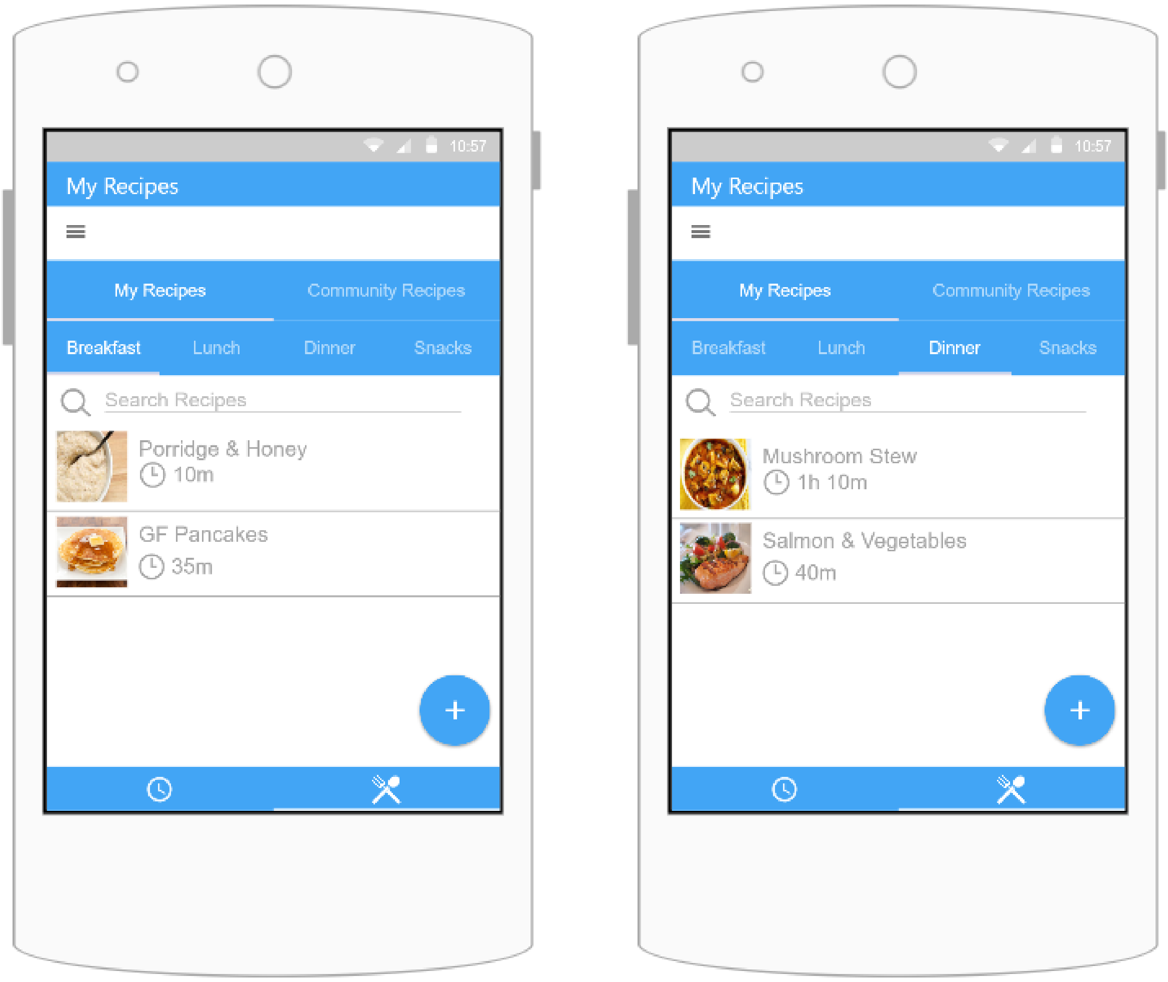


Figure 7 – My Recipes - Breakfast & Dinner

Figure 8 shows the pop-up menu when the user long presses on one of the recipes in “My Recipes”. The first option is to view the recipe. The view recipe screens will be discussed later in this section.There is an option to edit the recipe which allows the user to make changes to the photo, title, ingredients, cooking time, and cooking method. The screen for adding and editing a recipe will be shown further in this section. The next option allows them to delete a recipe. The fourth option is to share the recipe. The recipe can be shared within the community of the application or they can choose to share it on an external application like Facebook, WhatsApp, or e-mail. The share menu can be seen in *Figure 9*, although there is a high chance it will look completely different in the final application.

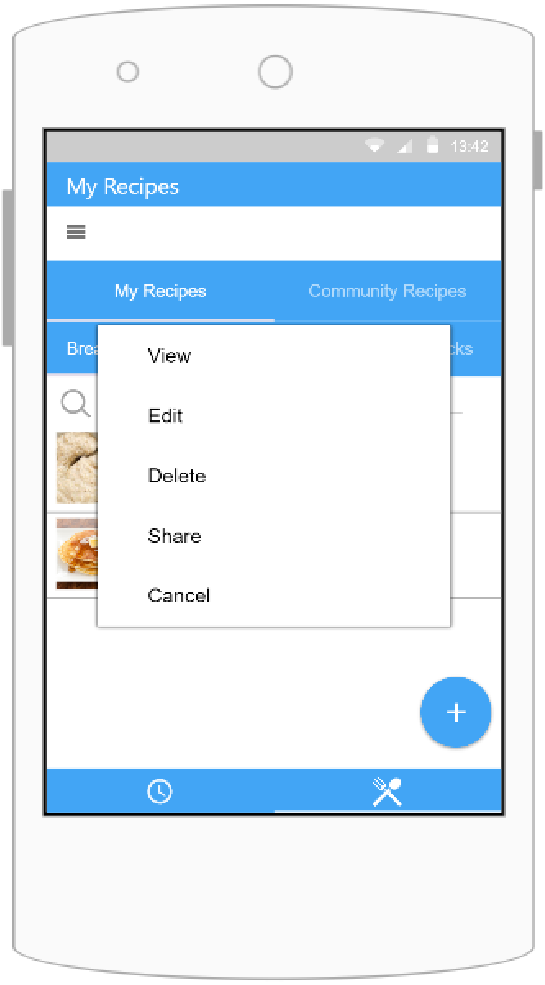


Figure 8 – “My Recipes” Pop-up Menu

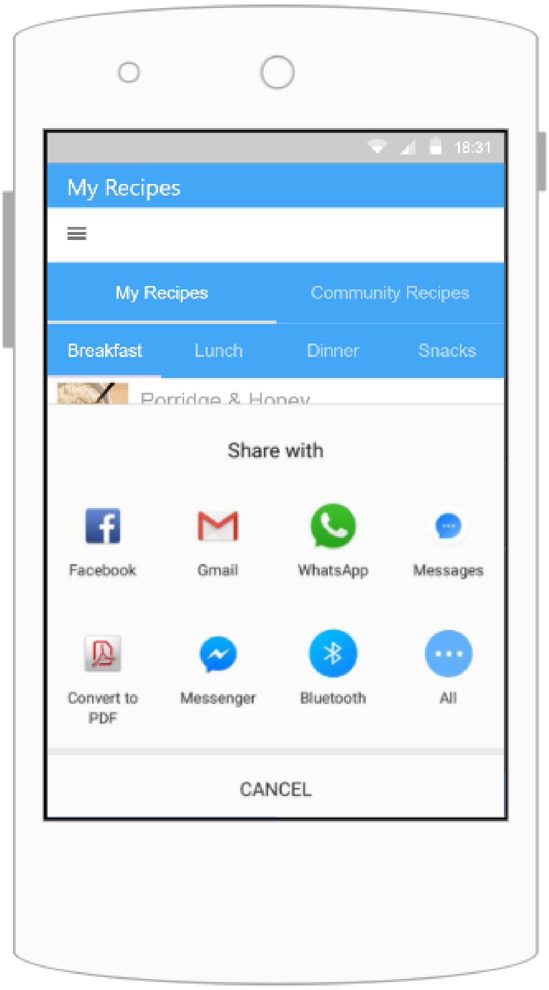


Figure 9 – “My Recipes” Share Option

Figure 10 show the community recipes section. This looks identical to the my recipes section but it consists of all of the recipes that users have shared with the community. The screen also shows the number of favorites a recipe has. This is shown with the heart and number beside the heart. When a user favorites a community recipe, the recipe is added to “My Recipes”. Users can also add recipes straight to the community. The recipe is also automatically added to “My Recipes”. Figure 11 shows the pop-up menu which is displayed when the user long presses a community recipe. The view option allows the user to view the recipe. The screens for viewing a recipe will be shown later in this section. The next option allows the user to share to an external application. The “Save To My Recipes” option saves the selected recipe to the users “My Recipes”.

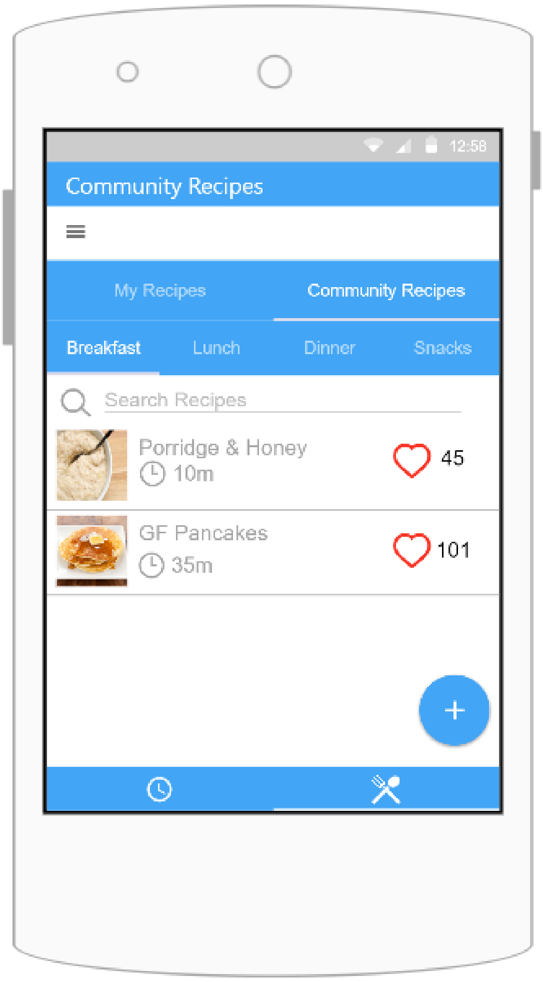


Figure 10 – Community Recipes

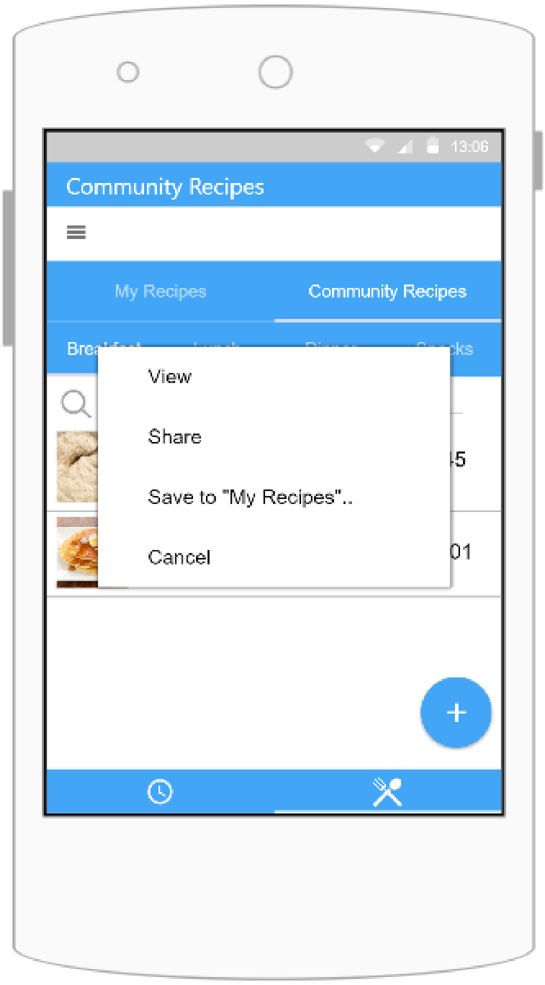
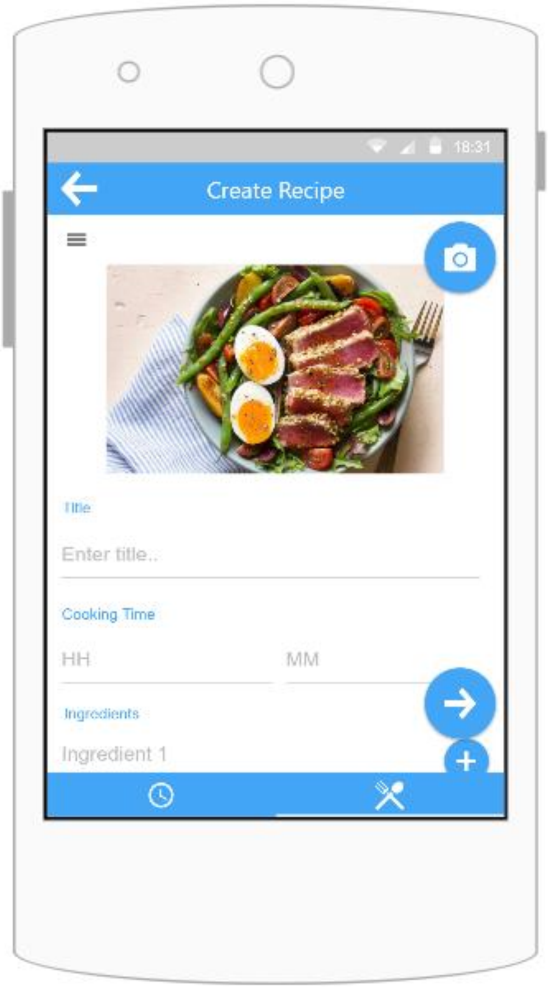


Figure 11 – Community Recipes Pop-up

When the user presses the plus button on the bottom right corner of either the community recipes or my recipes, they are brought to the add recipe screen. The first thing the user can add is a photo for the recipe. Next, they can add the title of the recipe and the cooking time. The user can then add the ingredients of the recipe. They can add a new ingredient by pressing the small plus button beside each ingredient. When they press the button, a new input field is displayed. This is the same system for adding each step of the recipe. When the user has filled out all fields and presses the next button on the bottom right of the screen, the recipe is saved. *Figures 12, 13, & 14* show this screen.



18:31

Create Recipe

Title

Enter title..

Cooking Time

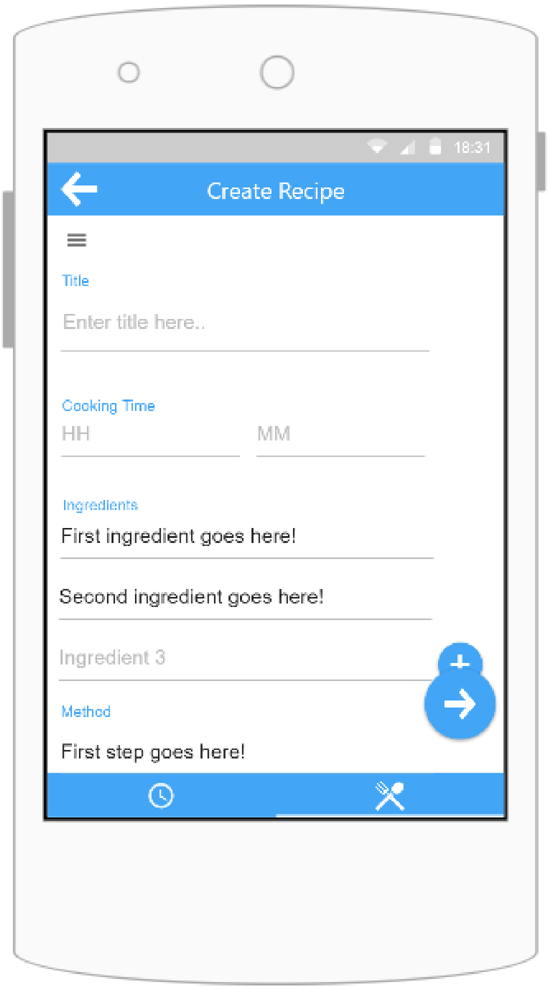
HH

MM

Ingredients

Ingredient 1

Figure 12 – Create Recipe – Part 1



18:31

Create Recipe

Title

Enter title here..

Cooking Time

HH

MM

Ingredients

First ingredient goes here!

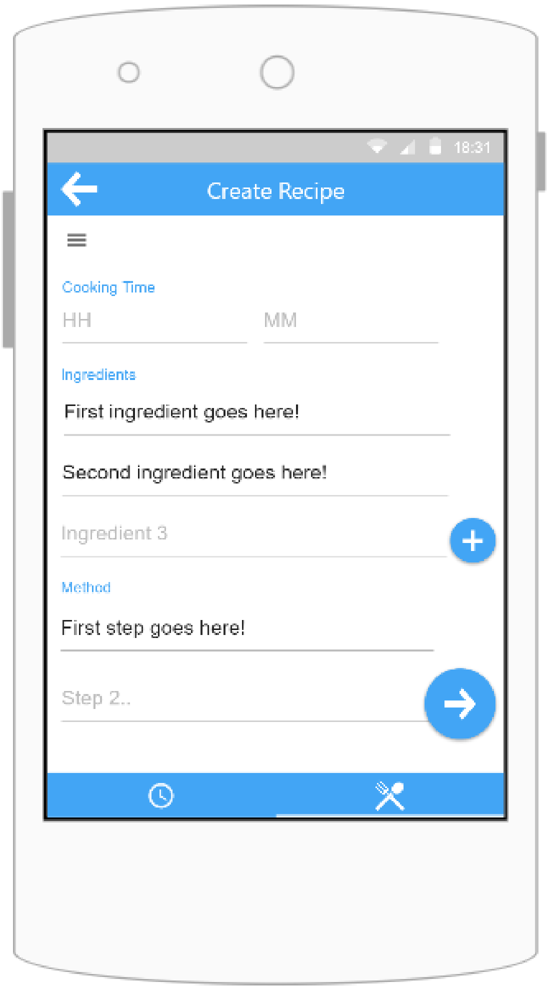
Second ingredient goes here!

Ingredient 3

Method

First step goes here!

Figure 13 – Create Recipe – Part 2



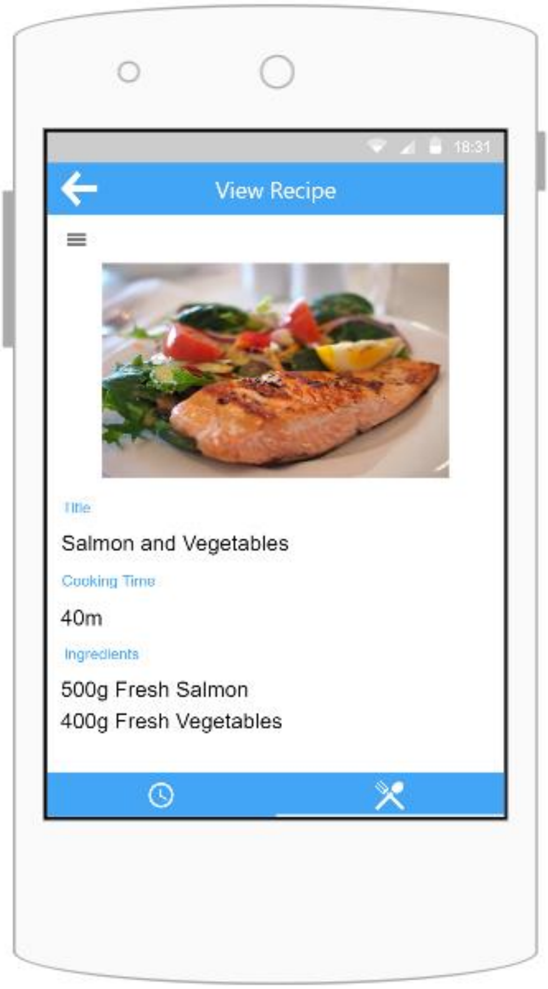
First ingredient goes here!

Second ingredient goes here!

First step goes here!

Figure 14 – Create Recipe – Part 3

Figure 15 shows the screens for viewing a recipe. This is a simple screen which presents the recipe information in a readable format to the user. *Figure 16* shows what happens to the view recipe screen when the screen orientation changes to landscape mode. Initially the first step of the recipe is shown and the user can press the next button to go the next step. This is a useful feature for when people are cooking and they don’t want to scroll through the method.



Title

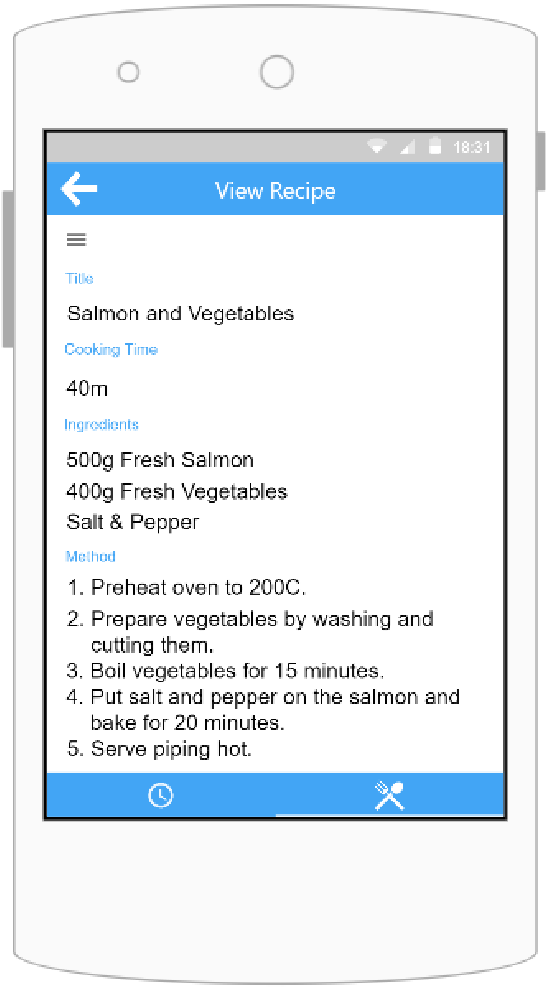
Salmon and Vegetables

Cooking Time

40m

Ingredients

500g Fresh Salmon
400g Fresh Vegetables



Title

Salmon and Vegetables

Cooking Time

40m

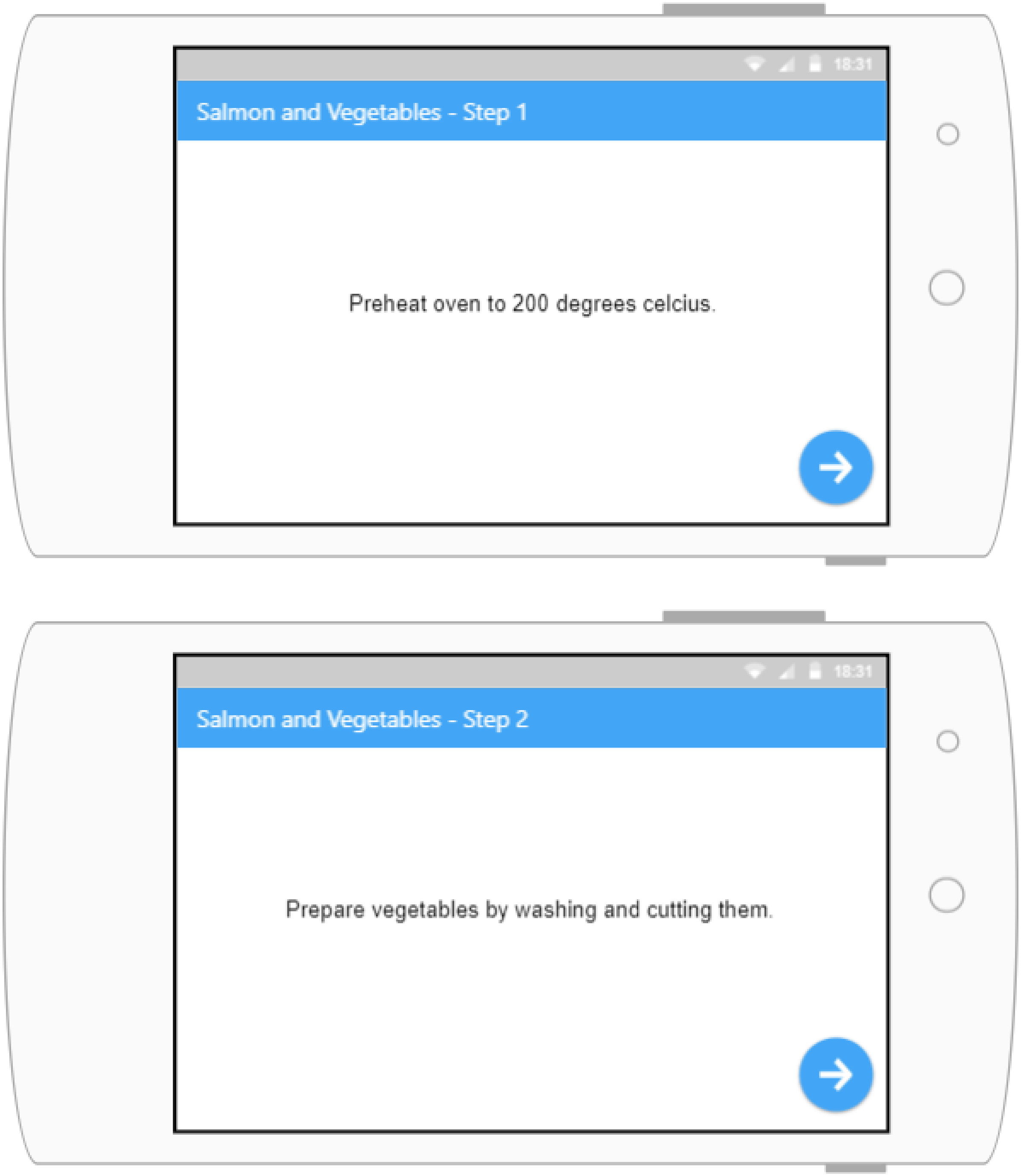
Ingredients

500g Fresh Salmon
400g Fresh Vegetables
Salt & Pepper

Method

1. Preheat oven to 200C.
2. Prepare vegetables by washing and cutting them.
3. Boil vegetables for 15 minutes.
4. Put salt and pepper on the salmon and bake for 20 minutes.
5. Serve piping hot.

Figure 15 – View Recipe



Salmon and Vegetables - Step 1

Preheat oven to 200 degrees celcius.

Salmon and Vegetables - Step 2

Prepare vegetables by washing and cutting them.

Figure 16 – View Recipe – Landscape Orientation

8. FURPS+

Functionality

- The application must always be connected to the internet in order to access the database and pull down the relevant data.
- The application must be secure as to keep user's personal data safe.

Usability

- A user must be able to add a diary entry in less than 5 clicks.
- A user must be able to edit a diary entry in less than 5 clicks.
- A user must be able to delete a diary entry in 2 clicks.
- A user must be able to add a new recipe in less than 7 clicks.
- A user must be able to delete a recipe in 2 clicks.
- A user must be able to share a recipe in 2 clicks.
- A user must be able to save a recipe in less than 5 clicks.
- A user must be able to search for a recipe in 2 clicks.

Reliability

- The system must have 99% up-time.
- The system must add a diary entry and a recipe 99% of the time.

Performance

- The application will load in less than 3 seconds on a fast internet connection.
- Each request should take no longer than 3 seconds to finish.
- Most pages will load in less than 1 second.

Supportability

- The system should support the English language.

+

- The system contains some sensitive data so should adhere to GDPR.
- The passwords of the system should be salted and hashed using an MD5 algorithm.
- A user's private recipes should be kept secure.