

Recipe Generator Mobile App Functional Specification

Ronan Green

C00270395

Contents

| | |
|--|---|
| Section 1: Project Overview | 3 |
| 1.1 Introduction | 3 |
| 1.2 Project Description..... | 3 |
| 1.3 Scope | 3 |
| 1.4 Audience | 4 |
| 1.5 Problem Statement | 4 |
| 1.6 Value Proposition | 4 |
| 1.7 Business Model Canvas | 5 |
| Section 2: Functional Specification..... | 5 |
| 2.1 Core..... | 6 |
| 2.1.1 Ingredients Identification | 6 |
| 2.1.2 Recipe Recommendations | 6 |
| 2.1.3 Shopping List Creation | 6 |
| 2.1.4 Barcode Scanning | 7 |
| 2.1.5 Healthy Alternatives. | 7 |
| 2.1.6 User Profile | 7 |
| 2.2 Non-Core | 7 |

| | |
|---|----|
| 2.2.1 Fingerprint Authentication..... | 7 |
| 2.2.2 Recipe Rating..... | 7 |
| 2.2.3 Dark Mode..... | 7 |
| 2.2.4 Recipe Scaling | 7 |
| 2.2.5 Nutritional Value Display | 8 |
| Section 3: Non-Functional Requirements..... | 8 |
| 3.1 Usability | 8 |
| 3.2 Reliability | 8 |
| 3.3 Performance | 8 |
| 3.4..... | 9 |
| Section 5: System Overview | 10 |
| 5.1 Context Diagram..... | 10 |
| 5.2 Constraints..... | 10 |
| 5.3 Assumptions | 10 |
| Section 6: Use Cases..... | 11 |
| 6.1 Diagram | 11 |
| 6.2 Brief Use Cases..... | 11 |
| 6.3 Detailed Use Cases | 14 |
| Section 8: Appendices..... | 17 |

Section 1: Project Overview

1.1 Introduction

The following is a Functional Specification for my fourth-year project, EasyChef. This functional specification will be explained through FURPS, covering all functional and nonfunctional requirements, Use cases, system architecture and the constraints for the Application.

1.2 Project Description

EasyChef is an android application designed to improve user's lives when it comes to ingredient management and cooking. The EasyChef is a new and improved way to manage your recipes and shopping lists while not having to waste precious time inserting ingredients or scoring for recipes. With the power of AI the EasyChef will help users improve their eating habits, ingredients management, and overall convenience when cooking.

The core of the app is built around image recognition, barcode scanning and manual input from the user to find the most accurate and tasty recipes that fit the users tastes no matter how unique. Not only will it provide recipes it will offer a personal experience for each user making their recipe list their own and keeping track of loved items in the shopping list for the user to buy. From home chefs to new cooks this app will help track the macro nutrients of your ingredients offering healthier options while sacrificing little taste.

The AI model is trained off various datasets(Food 101, ImageNet) allowing for accurate ingredient identification as well as the user's preferences to allow good recommendations for recipes and healthy alternatives.

1.3 Scope

The EasyChef will allow users to take pictures of various ingredients and create a recipe using AI to recognise the ingredients and machine learning to produce a recipe best suited to the user's needs.

It will also have various other features such as creating shopping list and providing healthier alternatives.

The app will have barcode scanning also implemented to scan ingredients if the camera cannot pick the item.

As camera quality varies a manual input will also be included as an option if the camera is struggling to pick items/barcodes up or if the user would rather input the data manually.

1.4 Audience

There are many applications available nowadays that help with recipe creation and nutritional help. However, people also want to be able to do mundane tasks quicker and quicker. With the ever-evolving AI I plan to make an efficient replacement to the out-of-date Recipes apps, and implement a quick and accurate photo related app.

This will be targeted towards home cook chefs, people wanting specific recipes for diets and a way to manage all your food related needs for mother who want to know what they need without thinking.

1.5 Problem Statement

Now more than ever, people are finding themselves with less time, making it increasingly difficult to plan healthy meals and shopping lists. Despite these time constraints, studies show an increase in home cooking. For example, from 2003 to 2016, home cooking in the U.S. increased by 4% among educated women and 14% among men[1]. Similarly, a study conducted in France revealed that during lockdown, home cooking increased by 42% among both men and women, with 60% of French people continuing to cook more frequently post-lockdown[2].

However, the obvious alternative to home cooking, takeaways, remains prevalent, particularly among higher-income households[3]. Studies from New Zealand show that takeaways tend to be unhealthy, with higher energy content and poor nutritional quality, often appealing to price-sensitive customers seeking convenience[4].

In Ireland, home cooking remains a significant part of cultural heritage, with 75% of parents learning their cooking skills from their mother[5]. This tradition brings a homely connection to food preparation, allowing families to control what they eat while exploring different food cultures[6].

1.6 Value Proposition

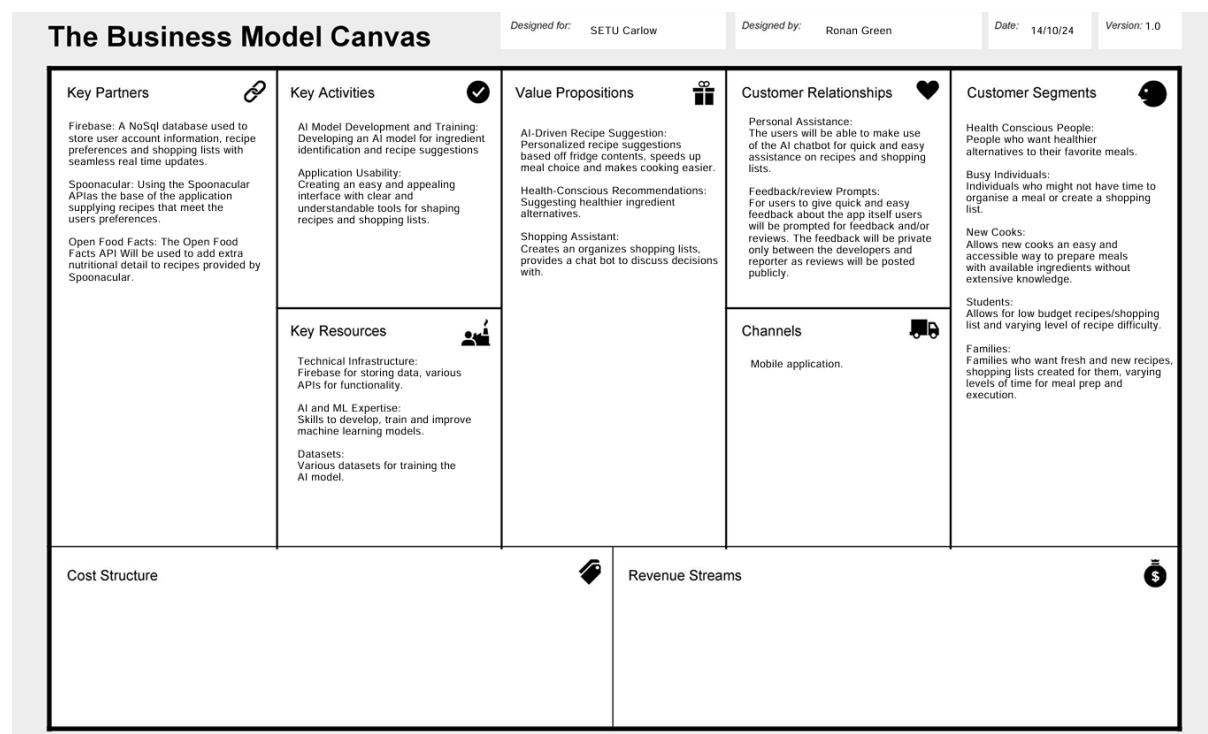
The EasyChef application will provide a streamlined and simple way to easily plan and create meals using AI-powered recipe generation and ingredient management. Aimed at busy individuals, people who want to plan healthy meals or students on a tight budget, EasyChef will provide a stress free experience making all the steps of

planning become as simple as taking a picture and accepting a recipe. EasyChef features will allow for:

- Time Saving: Automatically generate shopping lists and recipes based on the ingredients you already have
- Personalised recommendations: Receive personalised recipe recommendations tailored to your dietary preferences and nutritional needs.
- Health-conscious choices: Get suggestions for healthier ingredient swaps to ensure your meals are nutritious without compromising on taste.

Planning meals can be hard, especially with limited ingredients, or mundane meals. EasyChef will help keep your meals fresh, save you time, and ensure you use your ingredients to their full potential. Not only will it make your cooking life easier but will also allow you to easily shop for the ingredients again with shopping list generated to your preferences and needs. All this will be done with the EasyChef trained AI-model allowed for the ultimate recipe generating experience.

1.7 Business Model Canvas



Section 2: Functional Specification

This section covers the core and non-core functionality of the Application. Each function is a core function of the application allowing the app to function as expected. Each feature is designed to enhance the user's experience and ability to control their flavours.

2.1 Core

2.1.1 Ingredients Identification

- One of the main features of this application is identifying ingredients that have been taken in a picture. Using the camera option in the application allows users to display an array of ingredients and capture them for the system to identify.
- This can be done over a multitude of pictures if all the ingredients don't fit in the first image, or if additional ingredients need to be added via barcode.
- The system will use the trained AI Model to identify each ingredient and then display a recipe suitable for the user to make with those ingredients. The ingredients are checked against Open Food Facts API to get a detailed breakdown of the ingredient.
- If the camera can't pick up the item or a barcode can't be scanned users can input the ingredients manually.
- The items identified are displayed to the user and the user can accept or reject the proposed ingredients.

2.1.2 Recipe Recommendations

- This is another core feature of the application and can be used by the user in many ways, these include taking a picture of ingredients, scanning the barcode of ingredients, manual input or simply just asking for a recipe.
- The system uses the show/scanned/inputted/favourited ingredients along with ingredients that have been saved as favourites by the user or system to recommend the ideal recipe for the user.
- Users are able to adjust the recommended recipes by selecting whether they would like certain ingredients included/excluded to if they want a specific meal plan e.g. gluten free, high protein, vegetarian.

2.1.3 Shopping List Creation

- This allows the user to add ingredients directly to a shopping list from their favourite recipes or commonly bought items.
- The shopping list will have recommendations based on what the AI model has learned about the user's taste. The user can view, edit and delete items from this shopping list.

- Items can be added via picture. as well as barcode if the user does not want to do it manually.

2.1.4 Barcode Scanning

- The barcode scanner allows users to scan barcodes of items to add them to a recipe or shopping list that cannot be recognised in a picture.
- The barcode is checked against Open Food Facts database allowing users to receive a detailed breakdown of the ingredient.
- If the item is not found the user can manually input the item details, which will be added to the Open Food Facts API.

2.1.5 Healthy Alternatives.

- This functionality is suggested in both in recipes and shopping lists. The AI model will reference liked ingredients of the user and find possible healthier solutions (e.g. swapping cream for a lower calorie substitute).

2.1.6 User Profile

- Users will be able to personalise the recommendations and recipes with their dietary preferences, favourite recipes, shopping list items along with personalisation of username and interface (e.g. light and dark)

2.2 Non-Core

2.2.1 Fingerprint Authentication

- This functionality will allow users to login to their accounts using fingerprint authentication. This feature will add extra security and convenience to users allowing them to access the app quicker.

2.2.2 Recipe Rating

- This allows users to rate the recipes they have tried, enabling the AI model to see which recipes are low, medium, or high on the users list allowing it to make more accurate decisions when recommending recipes.

2.2.3 Dark Mode

- The dark mode feature is prevalent and popular in many apps nowadays, and this will be no different. This will allow users to change the interface from light to dark or vice versa.

2.2.4 Recipe Scaling

- This functionality allows users to scale their recipes to account for cooking for more/less people. The measurements of the ingredients get larger or smaller depending on the number the person wishes to cook for.

2.2.5 Nutritional Value Display

- This function will display detailed information about ingredients that are part of recipes and shopping lists. It will give a break down of the calories, nutrients, carbs, proteins, fats allowing the user to see their intake for their ingredients.

Section 3: Non-Functional Requirements

3.1 Usability

A large part of EasyChef is making peoples cooking lives easier and to do this it is paramount the app is designed with a minimalistic UI/UX design allowing users ease of use while using the application. This will be done by only having necessary icons displayed, formatting and sectioning everything in a cohesive and satisfying manner. This approach will allow users to quickly access the app's core functions, such as ingredient identification, recipe generation, and shopping list creation. The app will first open with a step by step on how to access each core feature to allow for quick access and adaptability from users to the applications interface. EasyChef will make meal planning as efficient as possible, allowing users to save time and enjoy a stress-free cooking experience.

3.2 Reliability

A fundamental part of EasyChef's success hinges on the reliability as users will depend on the app for daily meal planning, ingredient management and recipe generation. To ensure this the app will be developed with stable, accurate and reliable features such as image and barcode recognition having minimal errors allowing users to efficiently plan their meals. Using an offline supported NoSql database will ensure that users have secure access to their recipes and shopping lists at all times, without needing to repeat tedious tasks. This will help maintain a seamless and dependable experience, even when network connectivity is limited.

3.3 Performance

To keep up the theme of allowing users quick and easy access to new recipes ingredients must be identified and recipes must be retrieved in a timely manner which studies seems to be about 2 seconds of wait time.[7] Allowing for the AI Model to identify ingredients the time EasyChef aims for is 5-7 seconds for the total time of identify and retrieving a recipe excluding the time for user interaction. This ensures that users can efficiently get their recipes without experiencing unnecessary delays, keeping the app responsive and aligned with user expectations for speed.

3.4 Supportability

Supportability makes EasyChef maintainable, extendable, and debuggable throughout its lifetime. A well-structured codebase, proper detailed documentation of the code, and tools for monitoring app performance will be employed to achieve this.

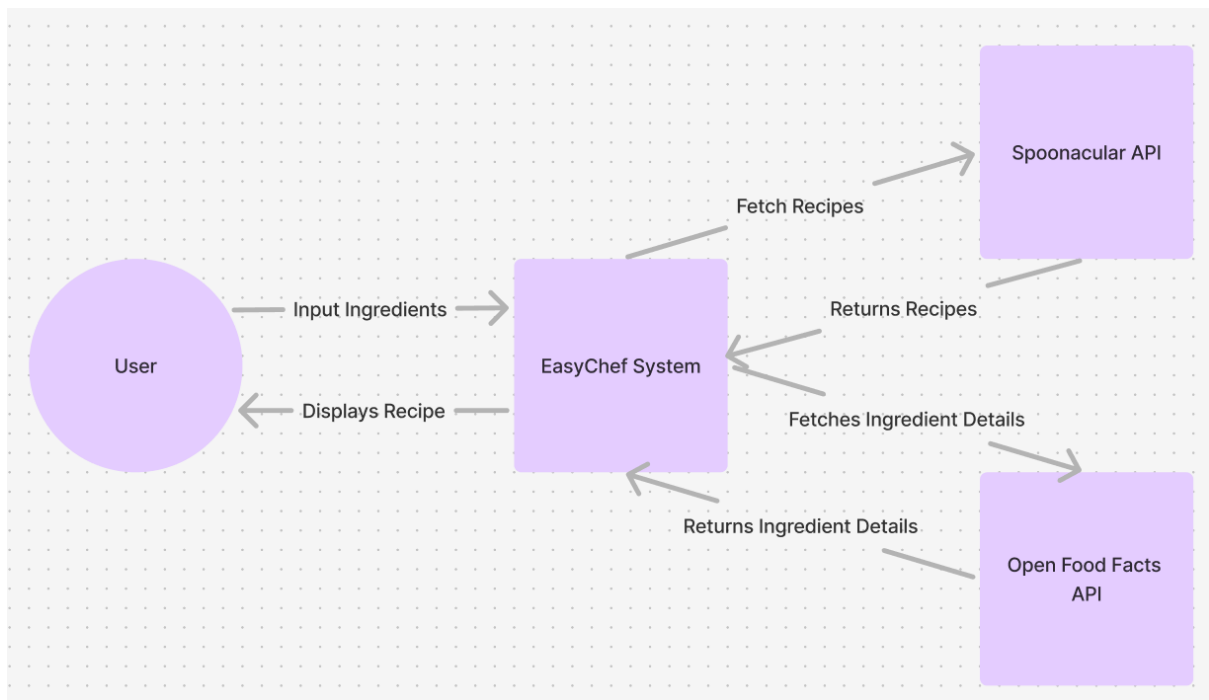
All these features will ease the burden on developers for future enhancements and fixes to the app. Furthermore, EasyChef will implement an efficient responsive customer support system to help maintain the performance of the app by improving seamlessness in the user's experience.

3.5 Security

At EasyChef, we take security and protection of user data very seriously, following standards set by the General Data Protection Regulation (GDPR). We make sure that any personal information processed by the application is done in a manner that is lawful, fair, and transparent. Of particular importance is the fact that EasyChef does not store images taken through the app. Images are only used for the purpose of identifying ingredients, and they are entirely and irreversibly deleted immediately after being processed. This ensures that no data of the image remains behind; thus, this makes the access or misuse of any sort less probable. With these practices in place, user privacy is hence guaranteed as part of the core values of the app.

Section 5: System Overview

5.1 Context Diagram



5.2 Constraints

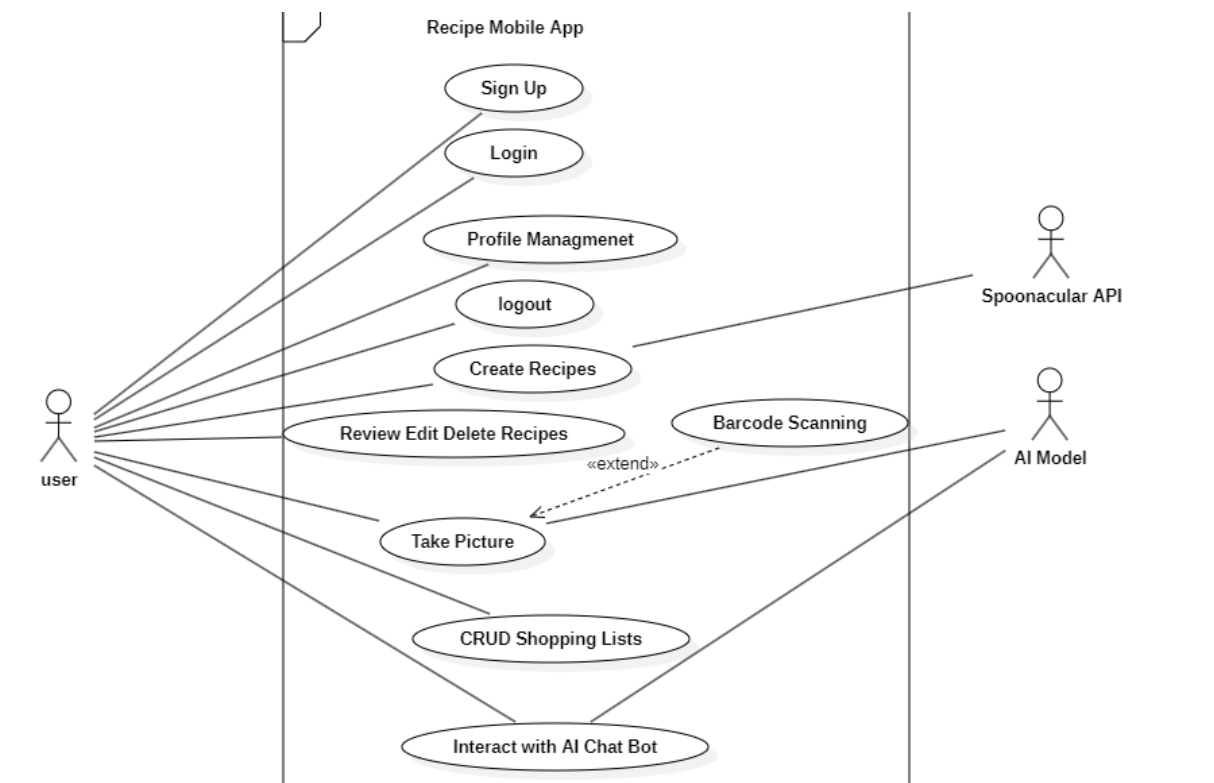
- Varying camera quality will lead some phones not usable for the app as their camera quality may be too low.

5.3 Assumptions

- Android device has android 5.0 or higher installed.

Section 6: Use Cases

6.1 Diagram



6.2 Brief Use Cases

| | |
|---------------|--|
| Use Case Name | Create Recipe |
| Unique ID | WW001 |
| Actors | Users, Database, Spoonacular, AI Model |
| Description | This use case is initiated when a user selects create recipe. This feature will create a new recipe using the AI Model and the Spoonacular API. It will use like items from the user's liked foods and will prompt the user for recipe details. With the information it will display a recipe the user can save. |

| | |
|---------------|--|
| Use Case Name | Take Picture of Ingredient |
| Unique ID | WW002 |
| Actors | User, AI Model, Spoonacular API |
| Description | The user opens the camera in the app and take a picture that include all |

| | |
|--|---|
| | relevant ingredients they wish to be a part of the recipe. The AI Model will identify the items and search the Spoonacular API for a relevant recipe. The user is then shown a recipe and they can decide whether they like it or not. If the user likes the recipe it will be added to the database. |
|--|---|

| | |
|---------------|--|
| Use Case Name | Chat with AI chat bot |
| Unique ID | WW003 |
| Actors | User, AI Model |
| Description | Using the OpenAI API as the base, this chatbot allows users to interact and talk about their recipes and shopping lists. Using the information from the AI Model (which contains like/dislikes) the chatbot will be able to help the user achieve the ideal recipe and shopping lists. |

| | |
|---------------|---|
| Use Case Name | View Recipe |
| Unique ID | WW004 |
| Actors | User, Database |
| Description | The view recipe use case will be triggered when a user wants to view recipes saved associated with their account. The system will retrieve what recipes the user had favourited and from there the user can choose which one they want to view. |

| | |
|---------------|--|
| Use Case Name | Edit Recipe |
| Unique ID | WW005 |
| Actors | User, Database, AI Model |
| Description | The edit recipe use case will be triggered when a user wants to edit recipes saved associated with their account. The system will retrieve what recipes the user had favourited and from there the user can choose which one they want to edit. The user will be able to edit the name or select an item in the recipe they want to change manually or get the AI model to recommend an alternative. |

| | |
|---------------|---------------|
| Use Case Name | Delete Recipe |
|---------------|---------------|

| | |
|-------------|--|
| Unique ID | WW006 |
| Actors | User, Database |
| Description | The delete recipe use case will be triggered when a user wants to delete recipes saved associated with their account. The system will retrieve what recipes the user had favourited and from there the user can chose which one they want to delete. |

| | |
|---------------|--|
| Use Case Name | View Shopping List |
| Unique ID | WW007 |
| Actors | User, Database |
| Description | The view shopping list use case will be triggered when a user wants to view their shopping list saved associated with their account. The system will retrieve the shopping list and from there the user can view the list. |

| | |
|---------------|---|
| Use Case Name | Edit Shopping List |
| Unique ID | WW008 |
| Actors | User, Database |
| Description | The edit shopping list use case will be triggered when a user wants to edit their shopping list saved associated with their account. The system will retrieve the shopping list and from there the user can edit the list. The user will be able to edit the name or select an item in the shopping list they want to change manually or get the AI model to recommend and alternative. |

| | |
|---------------|---|
| Use Case Name | Delete Shopping List |
| Unique ID | WW009 |
| Actors | User, Database |
| Description | The delete shopping list use case will be triggered when a user wants to delete shopping list saved associated with their account. The system will retrieve the shopping list and from there the user can delete the shopping list. . |

| | |
|---------------|----------|
| Use Case Name | Register |
| Unique ID | WW010 |

| | |
|-------------|--|
| Actors | User, Database |
| Description | This use case is a onetime use case for users and is started when they first want to register an account. The users enter the relevant information, name, email, password. Following this, the system will validate the information and create the user account. |

| | |
|---------------|---|
| Use Case Name | Login |
| Unique ID | WW011 |
| Actors | User, Database |
| Description | Initiating then a registered user attempts to access the application using their account. They will be prompted to enter username/email and password, and the system will validate the user's data before allowing them access to site. |

| | |
|---------------|---|
| Use Case Name | Logout |
| Unique ID | WW012 |
| Actors | Users |
| Description | This is initiated when the user decide they want their account to be logged out of on the device. This will stop the device being logged into that account and the app will display the register/login screen even if the app is closed and open again. |

| | |
|---------------|--|
| Use Case Name | Manage Profile |
| Unique ID | WW014 |
| Actors | User, Database |
| Description | This use case is triggered when the user selects profile management. This will allow the user to edit various qualities about their profile e.g. edit username, edit password, edit recipe preferences, dark/light mode. |

6.3 Detailed Use Cases

| | |
|---------------|--|
| Use Case Name | Create Recipe |
| Unique ID | WW015 |
| Actors | Users, Database, Spoonacular, AI Model |

| | |
|-------------------|--|
| Description | This use case is initiated when a user selects create recipe. This feature will create a new recipe using the AI Model and the Spoonacular API. It will use like items from the user's liked foods and will prompt the user for recipe details. With the information it will display a recipe the user can save. |
| Preconditions | User is registered and logged in |
| Trigger | The user selects create recipe. |
| Main Path | <ol style="list-style-type: none"> 1. The user prompts the system to create the recipe. 2. The recipe uses liked foods and prompts the user for extra information to create the recipe. 3. Once the user has input the data it will use the AI Model and Spoonacular API to display a recipe for the user. 4. The user accepts the recipe, and it gets added to the database for the user to view. |
| Post Conditions | The user has an added recipe in the database |
| Alternative Flows | 4a: <ol style="list-style-type: none"> 4. The user rejects the recipe. 5. The system gets a new recipe and prompts it to the user. |

| | |
|---------------|--|
| Use Case Name | Take Picture of Ingredient |
| Unique ID | WW016 |
| Actors | User, AI Model, Spoonacular API |
| Description | The user opens the camera in the app and take a picture that include all relevant ingredients they wish to be a part of the recipe. The AI Model will identify the items and search the Spoonacular API for a relevant recipe. The user is then shown a recipe and they can decide whether they like it or not. If the user likes the recipe it will be added to the database. |
| Preconditions | <ul style="list-style-type: none"> - The user is registered and logged in. - The user has given the app permission to access the camera. |

| | |
|-------------------|--|
| Trigger | The user opens the camera in the app. |
| Main Path | <ol style="list-style-type: none"> 1. User opens the camera 2. The user takes a picture of the ingredients they wish to have in the recipe. 3. The AI Model then recognises the ingredients identifying each one and then making a list to send to the Spoonacular API. 4. The user will get prompted to take another picture to add more ingredients 5. The Spoonacular API then checks for a matching recipe and then it get displays to the user 6. The user then accepts the recipe and the recipe gets added to the database for the user to view. |
| Post Conditions | New recipe is added to the database |
| Alternative Flows | <p>2a:</p> <ol style="list-style-type: none"> 2. The user takes a picture of the barcode. 3. The item then gets checked against Open Food Facts API. 4. The item is temporarily stored, and the user is prompted to scan again <p>3a:</p> <ol style="list-style-type: none"> 3. The system cannot pick up any items. This can be due to poor picture or unclear items. 4. The system prompts the user to take another picture. <p>6a:</p> <ol style="list-style-type: none"> 6. The user rejects the proposed recipe, and the system then finds a new one. |

| | |
|---------------|---|
| Use Case Name | Interact with AI Chat Bot |
| Unique ID | WW017 |
| Actors | User, AI Model |
| Description | Using the OpenAI API as the base, this chatbot allows users to interact and talk about their recipes and shopping lists. Using the information from the AI Model (which contains like/dislikes) the chatbot |

| | |
|-------------------|---|
| | will be able the help the user achieves the ideal recipe and shopping lists. |
| Preconditions | The user must be registered and logged into the system. |
| Trigger | The user selects the AI chatbot |
| Main Path | <ol style="list-style-type: none"> 1. The user is logged in an has selected the chatbot. 2. The user prompts the chatbot with a question("I want more protein in my diet") 3. Using the information provided by the system, such as user recipes, shopping list, preferences the chatbot thinks of a response 4. The chatbot will respond prompting the user to reply again. ("A good solution would be to replace recipe X's chicken thighs with chicken breast, would you like me to make this change?") 5. The user accepts this change the recipe will be updated accordingly. ("Yes") |
| Post Conditions | The recipe has been updated in the database |
| Alternative Flows | 5a. User Rejects Recommendation: <ol style="list-style-type: none"> 1. The user rejects the chatbot suggestion and no changes are made. |

Section 8: Appendices

[1]

Taillie, L.S. (2018). Who's cooking? Trends in US home food preparation by gender, education, and race/ethnicity from 2003 to 2016. *Nutrition Journal*, 17(1).
doi:<https://doi.org/10.1186/s12937-018-0347-9>.

<https://link.springer.com/article/10.1186/s12937-018-0347-9>

[2]

Sarda, B., Delamaire, C., Serry, A.-J. and Ducrot, P. (2022). Changes in home cooking and culinary practices among the French population during the COVID-19 lockdown. *Appetite*, 168, p.105743. doi:<https://doi.org/10.1016/j.appet.2021.105743>.

<https://www.sciencedirect.com/science/article/pii/S0195666321006504>

[3]

Turrell, G. and Giskes, K. (2008). Socioeconomic disadvantage and the purchase of takeaway food: A multilevel analysis. *Appetite*, 51(1), pp.69–81. doi:<https://doi.org/10.1016/j.appet.2007.12.002>.

<https://www.sciencedirect.com/science/article/abs/pii/S0195666307004308>

[4]

Mahawar, N., Jia, S.S., Korai, A., Wang, C., Allman-Farinelli, M., Chan, V., Raeside, R., Phongsavan, P., Redfern, J., Gibson, A.A., Partridge, S.R. and Roy, R. (2022). Unhealthy Food at Your Fingertips: Cross-Sectional Analysis of the Nutritional Quality of Restaurants and Takeaway Outlets on an Online Food Delivery Platform in New Zealand. *Nutrients*, 14(21), p.4567. <https://www.mdpi.com/2072-6643/14/21/4567>

<https://www.mdpi.com/2072-6643/14/21/4567>

[5]

McCloat, A., Mooney, E., & Hollywood, L. E. (2017). Have Irish parents put cooking on the back burner? An Island of Ireland study of the food skills, cooking confidence and practices of parents. *British Food Journal*, 119(5), 992-1002.

<https://doi.org/10.1108/BFJ-09-2016-0440>

<https://www.emerald.com/insight/content/doi/10.1108/BFJ-09-2016-0440/full/html>

[6]

Simmons, D., & Chapman, G. E. (2012). The significance of home cooking within families. *British Food Journal*, 114*(8), 1184-1195.

<https://doi.org/10.1108/00070701211252110>

<https://www.emerald.com/insight/content/doi/10.1108/00070701211252110/full/html>

[7]

Nah, F.F.-H. (2004). A study on tolerable waiting time: how long are Web users willing to wait? *Behaviour & Information Technology*, 23(3), pp.153–163.

doi:<https://doi.org/10.1080/01449290410001669914>.

<https://www.tandfonline.com/doi/abs/10.1080/01449290410001669914>