Software Requirements Specification

for

Brew Day!

Version 4.0 approved

Prepared by Zhenghao Wu, Anqin Zha, Renjie Deng and Ruichao Zhong

Dijkstra

Tuesday, March 26, 2019

Table of Contents

Revision History

Name	Date	Reason For Changes	Version
Zhenghao Wu, Anqin Zha, Renjie Deng and Ruichao Zhong	2019-03-04	The first version of the SRS	1.0
Zhenghao Wu, Anqin Zha, Renjie Deng and Ruichao Zhong	2019-03-12	Add two feature details and corresponding user interface design Change reference to APA format Add a design and implementation constrain Update the user case diagram	2.0
Zhenghao Wu, Anqin Zha, Renjie Deng and Ruichao Zhong	2019-03-19	Correct feature details and UI design for the first two feature Add 3 new feature details and corresponding UI design	3.0
Zhenghao Wu, Anqin Zha, Renjie Deng and Ruichao Zhong	2019-03-26	Correct feature details and UI design for the Maintain Equipment Information feature Add Class diagram for the system Add a sequence diagram for “add” use case in maintaining ingredients.	4.0

1. Introduction

1.1 Purpose

This document is for product named ‘‘Brew *Day!” which* developed by group *Dijkstra*. The document will be covered Introduction of the document, overall description, main features, external requirements and other non-functional requirements of product named *Brew Day!*

This document is written to help developers to develop the software conveniently, and for client to know what major features this product will provide.

This document is for the whole system of “*Brew Day!”* And no document for the partial function is provided.

1.2 Document Conventions

The title will use Times New Roman as font, for the first level title, the font size is 18, for the second level title, the font size is 14.

The main part of this document will use the font “Arial” and the font size is 11. Additionally, this document will use **bold text** to emphasize the important words. and *italic text* to indicate product names and company names in the main part.

For the acronyms in this document, please refer to Appendix A in part 7.

1.3 Intended Audience and Reading Suggestions

This Document is written for clients, developers and tester.

- For clients, the description of this product in part 2 and main features of this product in part 3 are recommended.
- For developers, the whole document is recommended to read.
- For tester, the content in part 2 and the nonfunctional requirements in part 5 are recommended.

1.4 Project Scope

This product is made for home brewer to brew beers. With this product, home brewer will have the ability to record and modify the information of ingredients relative to brewing beer to create a better brewing experience.

1.5 References

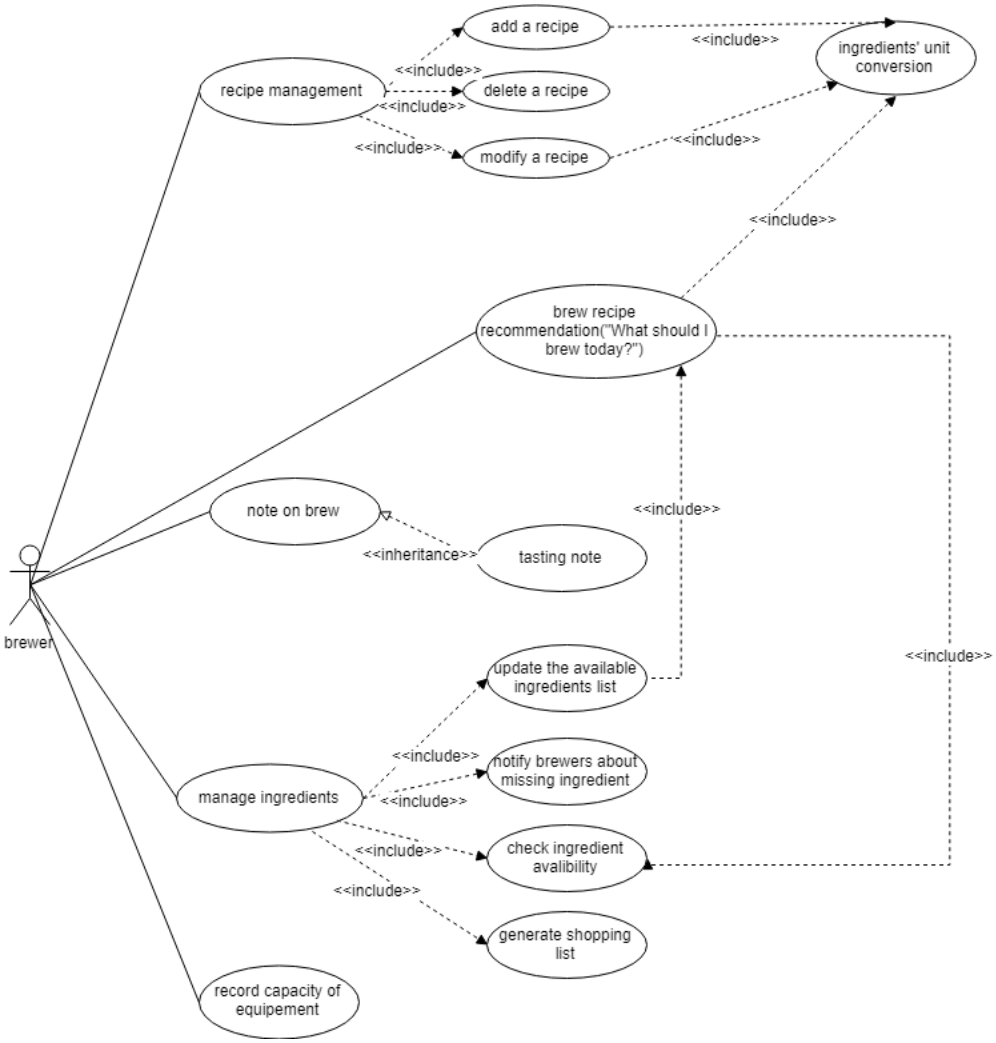
- Brew Target: Main brewtarget source code repository. (2019). GitHub. Retrieved 11 March, 2019, from https://github.com/Brewtarget/brewtarget
- Allaboutbeercom. (2019). All About Beer. Retrieved 11 March, 2019, from http://allaboutbeer.com/learn/glossary/

2. Overall Description

2.1 Product Perspective

This is a new independent system and there is no context in this system.

2.2 Product Features



Scenario 1: Add recipe

- Brewer enter the recipe management and click add recipe
- The system displays a recipe input form
- The Brewer input the recipe information
- The system converts the ingredient unit
- The system save the recipe to database

2.3 User Classes and Characteristics

The only user class is “all-grain” brewers for their home brewing on a small scale (the "extract" brews are not supported). Majority of them are expert in brewing. Additionally, they have some basic knowledge about computer. For example, they know how to use an application in Windows desktop.

2.4 Operating Environment

The application is desktop-based. In addition, the application should be able to run on Microsoft Windows.

2.5 Design and Implementation Constraints

Language requirements: Software uses English as the language of the user interface

2.6 User Documentation

An independent user manual will be provided, which will be written in English.

2.7 Assumptions and Dependencies

The user should have a PC running Windows. And user have a Java Runtime Environment 11 to run the program.

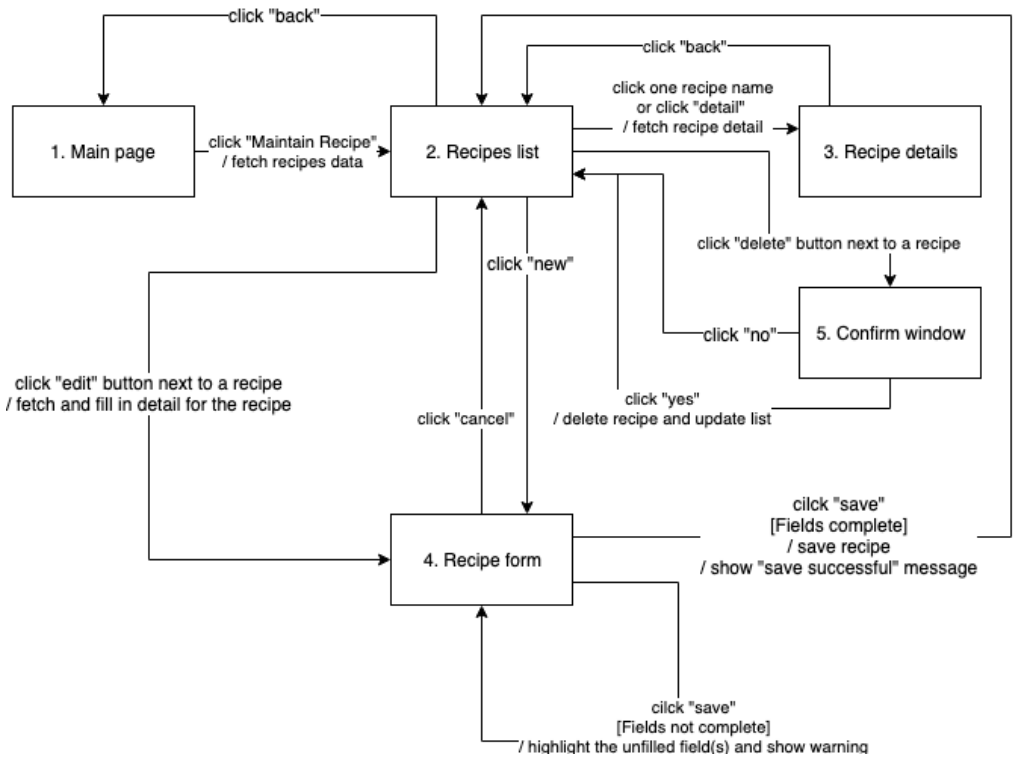
3. System Features

3.1 Maintain recipes

3.1.1 Description and Priority

Users can use this feature to view their recipes and manage their own recipes by doing addition, deletion and modification operations to the recipes. This is a high priority feature in the system.

3.1.2 Stimulus/Response Sequences



3.1.3 Functional Requirements

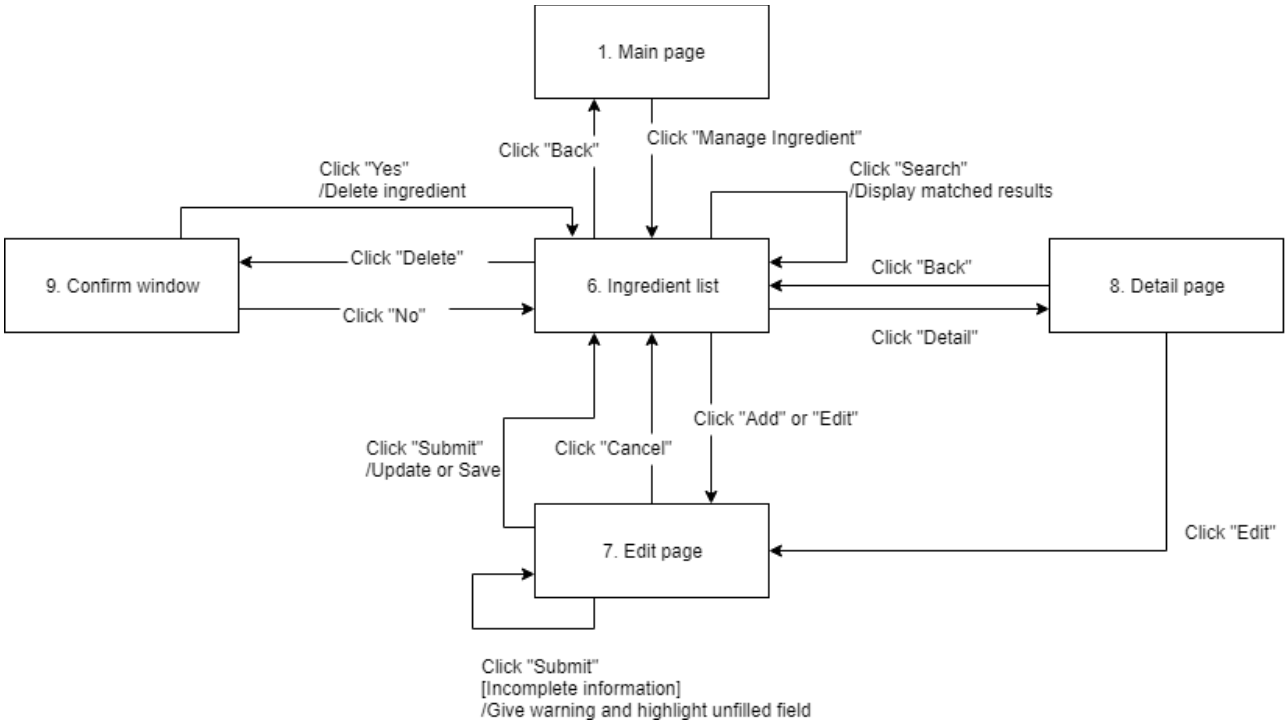
REQ-1: When the user creates and modifies the recipe, the system displays a recipe form for filling out. When the user submits the form, the system checks that all required fields of the form have been filled in and the format is correct. If the request is not met, the system rejects the submission of the form, highlights the fields that not met the requirement in the form, and displays a warning message.

3.2 Maintain ingredients

3.2.1 Description and Priority

The feature is to let users manage amount of ingredients in their house. If they have bought a new ingredient, they can use “add” to add it to system. If the ingredient is existing and its amount has changed, we can use edit page to change its amount.

3.2.2 Stimulus/Response Sequences



3.2.3 Functional Requirements

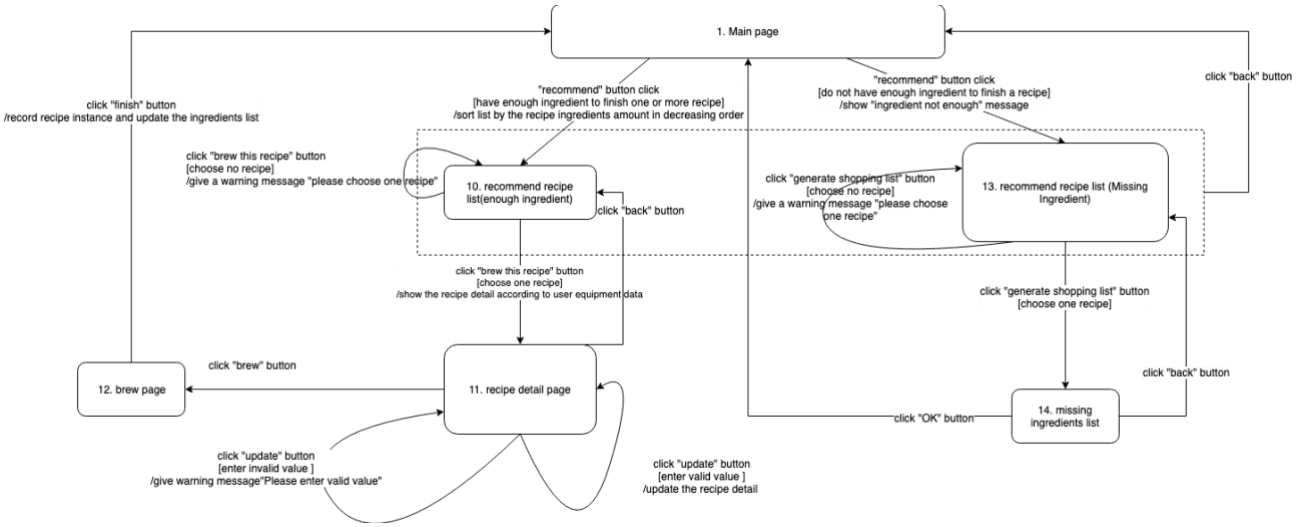
REQ-2: If any of user's input is empty, the system will ask user to complete the form.
REQ-3: If any of user's input is not match its data type, the system will ask user to fix.

3.3 Recommend a recipe

3.3.1 Description and Priority

This feature will give a recommend recipes list to user in order to help user brew with the available ingredients, maximizing the use of the ingredients, and the batch size. If the ingredients are not enough for any recipe, user also can select one of them to print out the missing ingredient list for them. This feature is in high priority.

3.3.2 Stimulus/Response Sequences



3.3.3 Functional Requirements

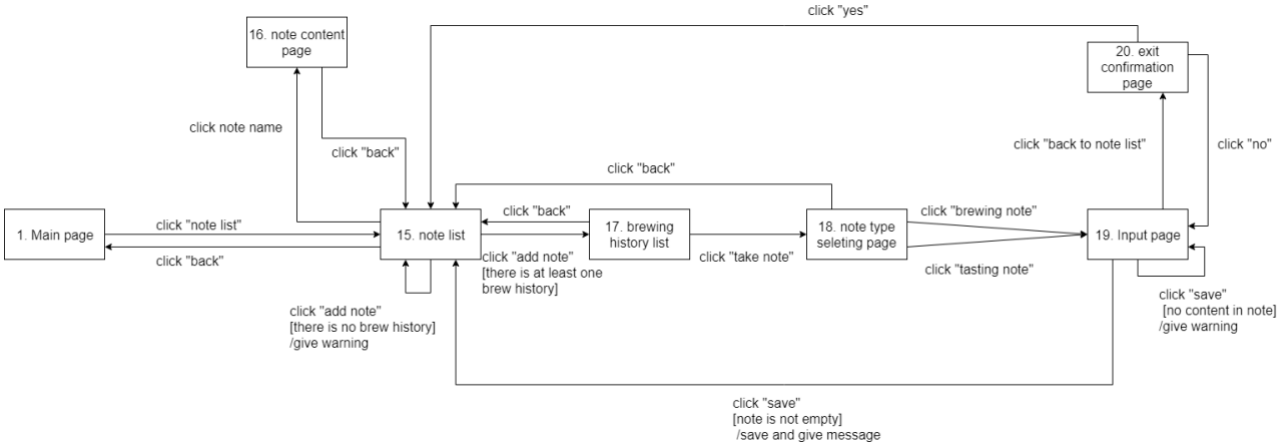
REQ-4: The listing order in the recommend recipe list should be in the number of ingredients used. The more ingredients the recipe used, the higher the recipe will place in the list.
REQ-5: A valid value in recipe detail page should greater than 0 and less or equal to equipment volume.

3.4 Write note

3.4.1 Description and Priority

This feature is for user to take notes on each brewing, which including brewing note and tasting note. This feature is median priority.

3.4.2 Stimulus/Response Sequences



3.4.3 Functional Requirements

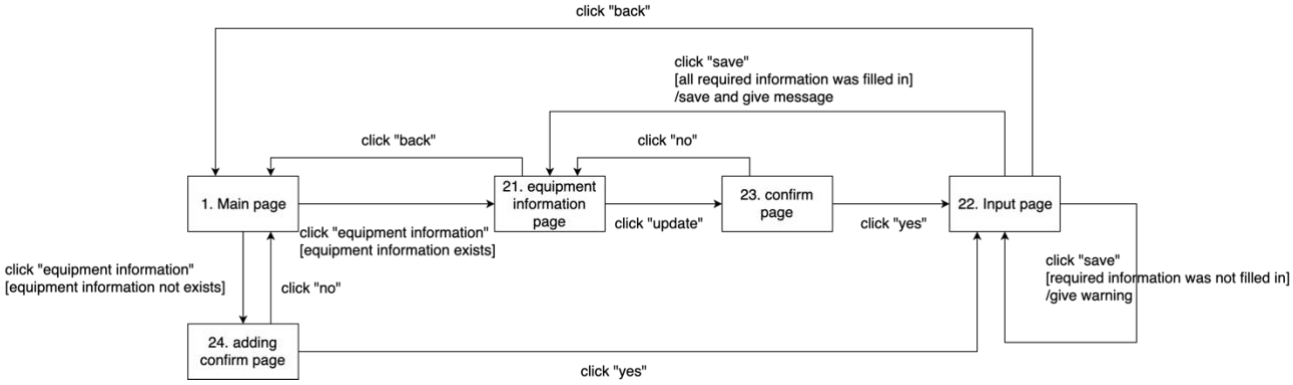
REQ-6: The note can only be saved after users input at least one character. If user have no input, the system will give a warning and return to the input state.
REQ-7: User may press “back to main page” by mistake while her/his inputting was not finished. A confirming window will jump out to verify whether users really want to exit the inputting page.
REQ-8: The note should not be too long to affect readability of that note. The system will forbit users to input more character when users’ input exceeds 100000 characters.

3.5 Maintain Equipment Information

3.5.1 Description and Priority

This feature is for user to maintain equipment information, which is the high priority feature.

3.5.2 Stimulus/Response Sequences

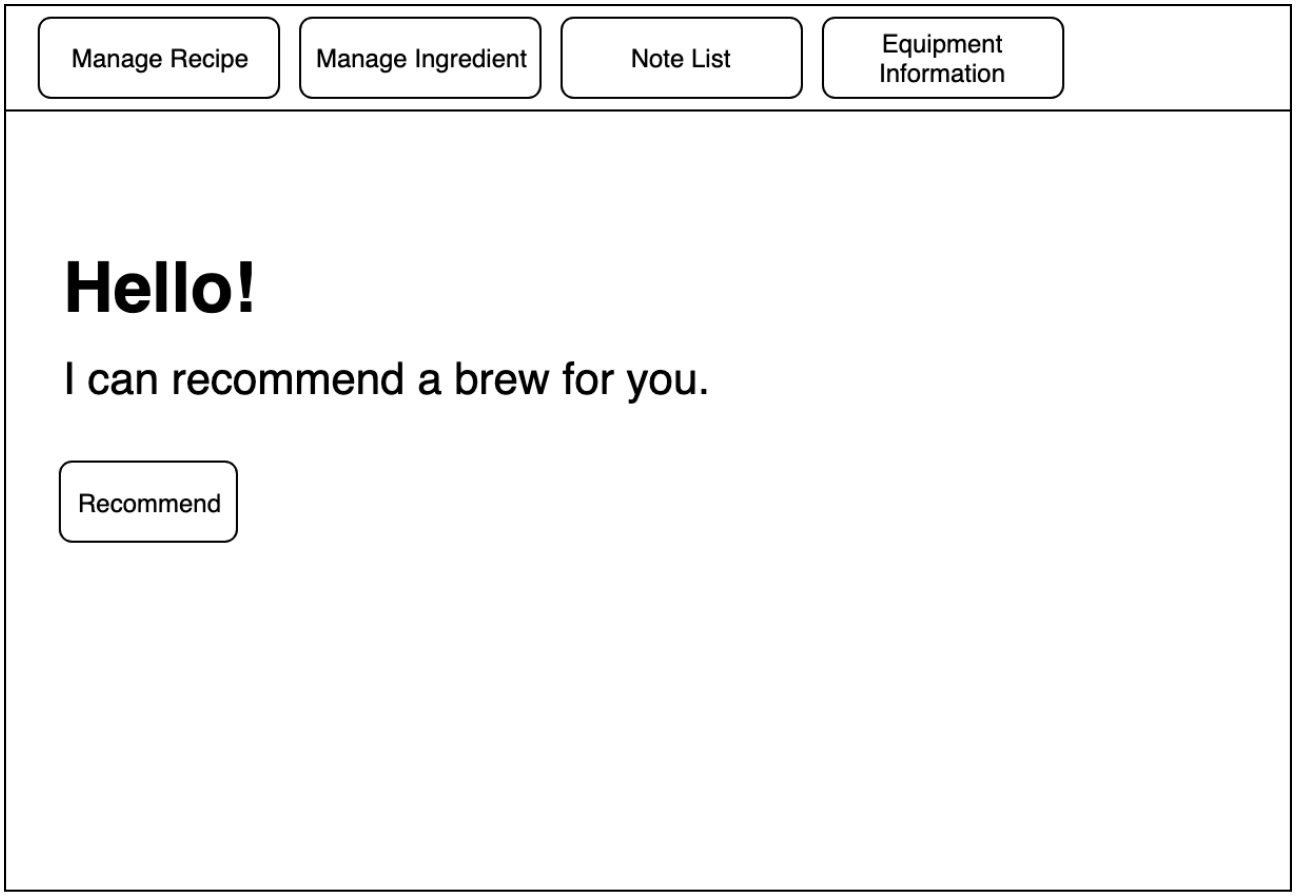


3.5.3 Functional Requirements

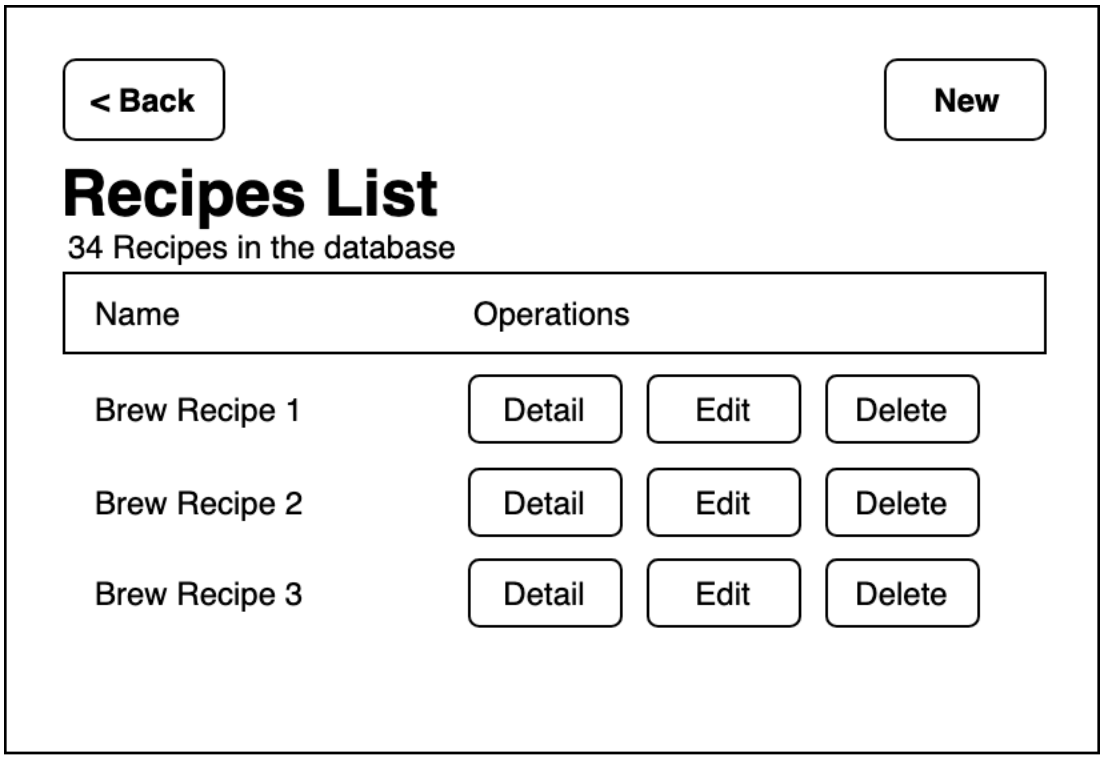
REQ-9: The note can only be saved after users input all required information, if user have not input all required information, the system will give a warning and back to the input state.
REQ-10: User may press “back to main page” by mistake while her/his inputting was not finished. A confirming window will jump out to verify whether users really want to exit the inputting page.
REQ-11: Some equipment information has specific information to input, for example, user can only input number for volume of an equipment, if the input contains words or symbols, the system should pop out a warning and let user to input again.

4. External Interface Requirements

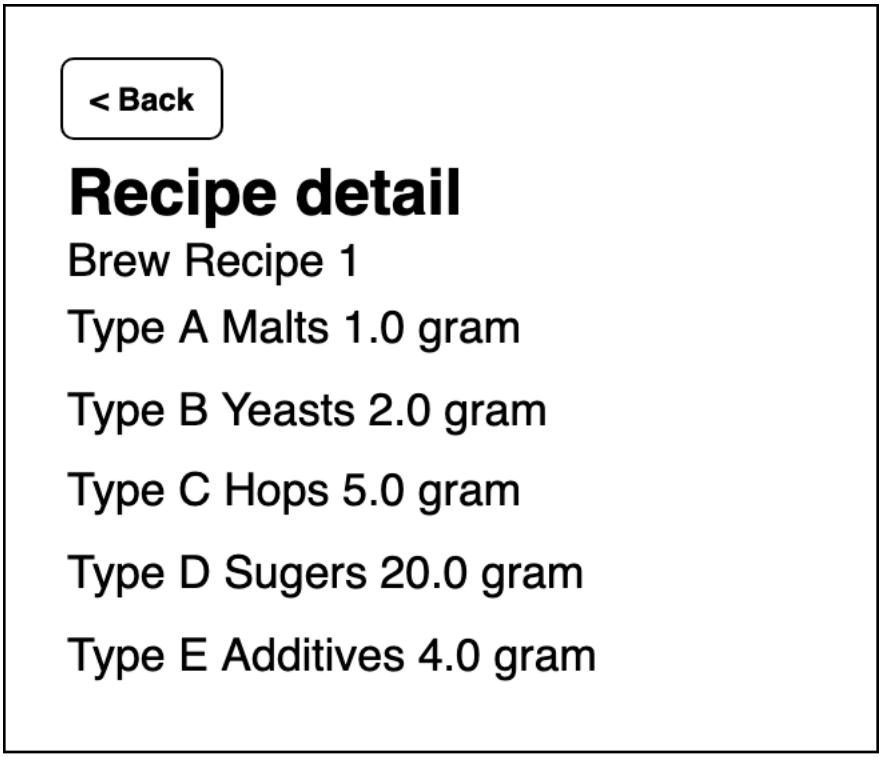
4.1 User Interfaces



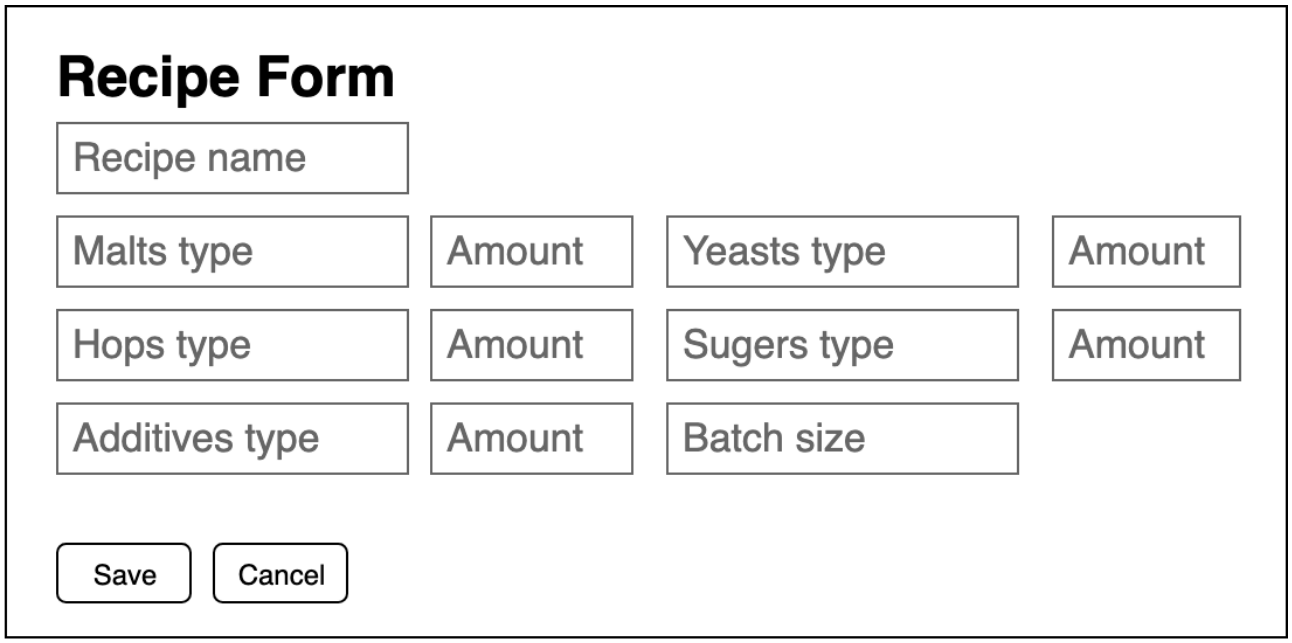
1. Main page



2. Recipes list

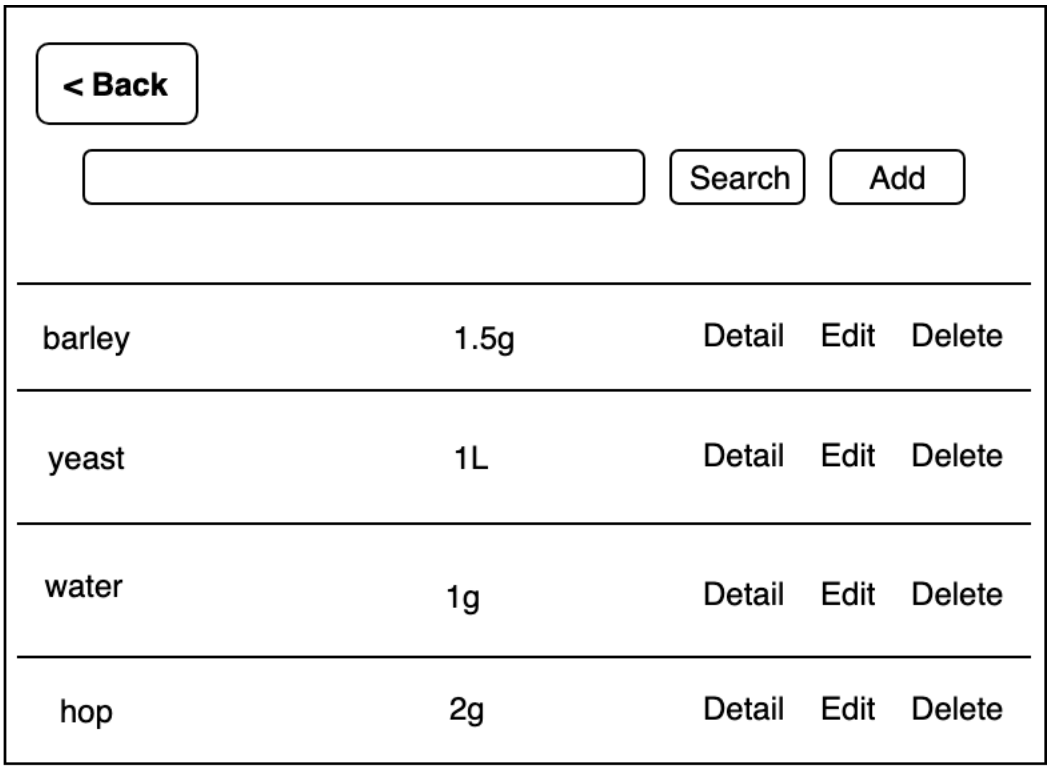


3. Recipe details

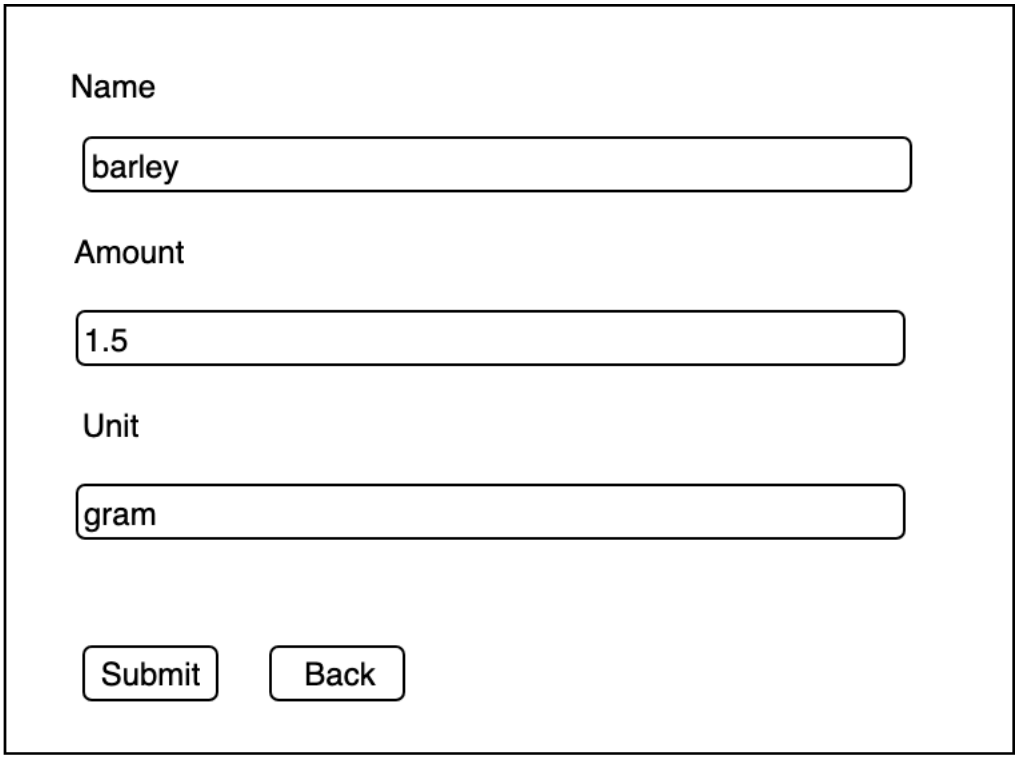


4. Recipe form

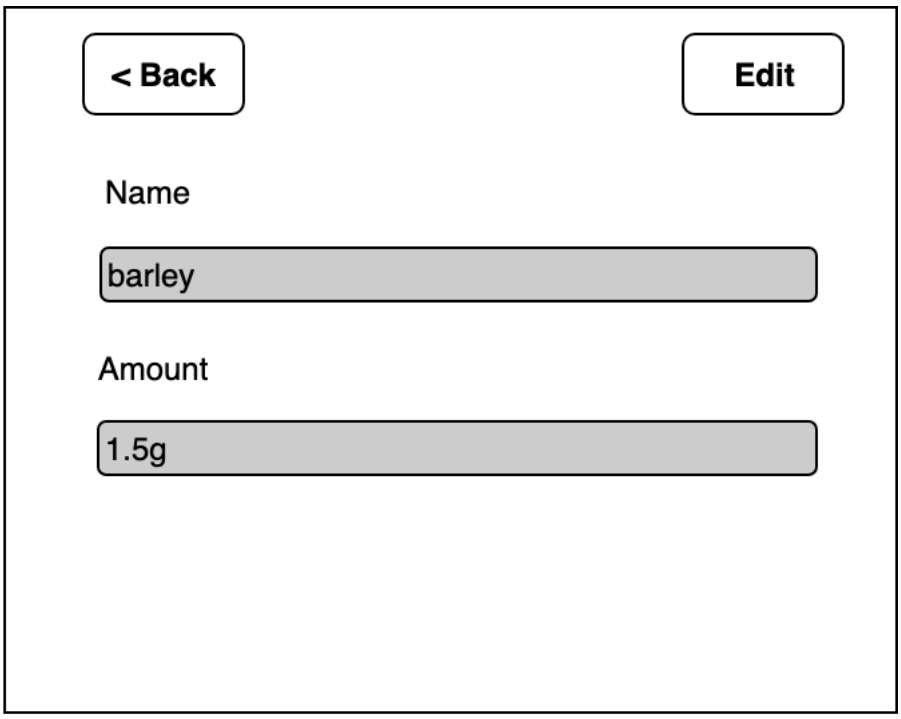
5. Confirm window



6. Ingredient list

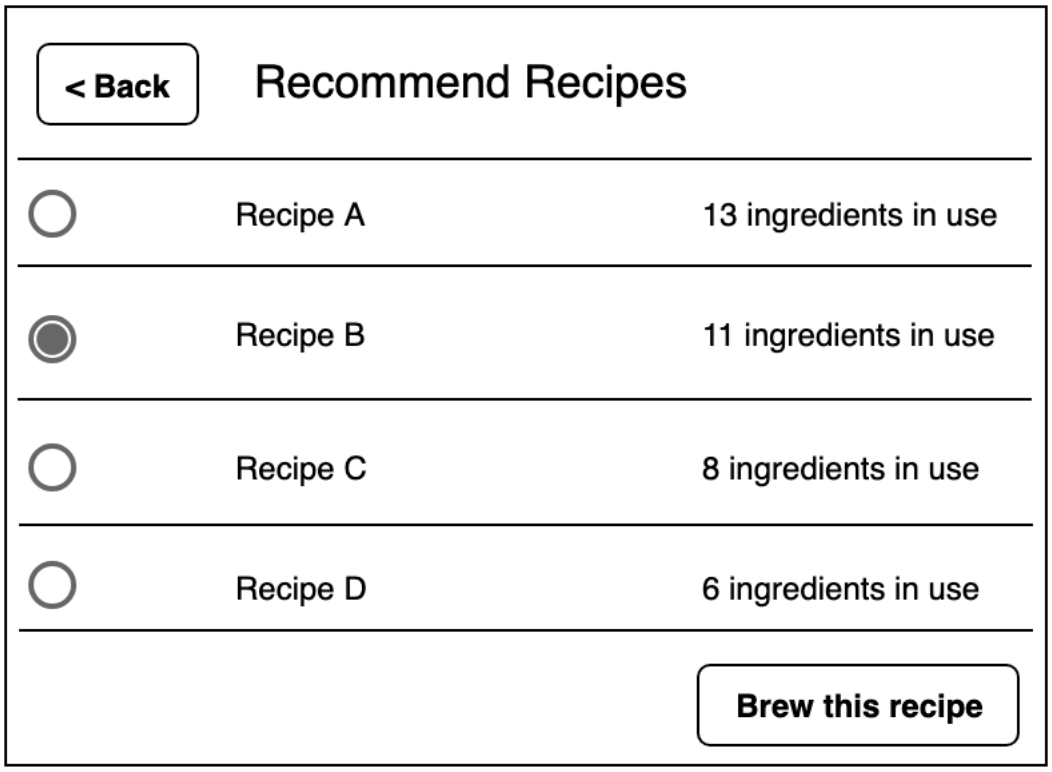


7. Edit page

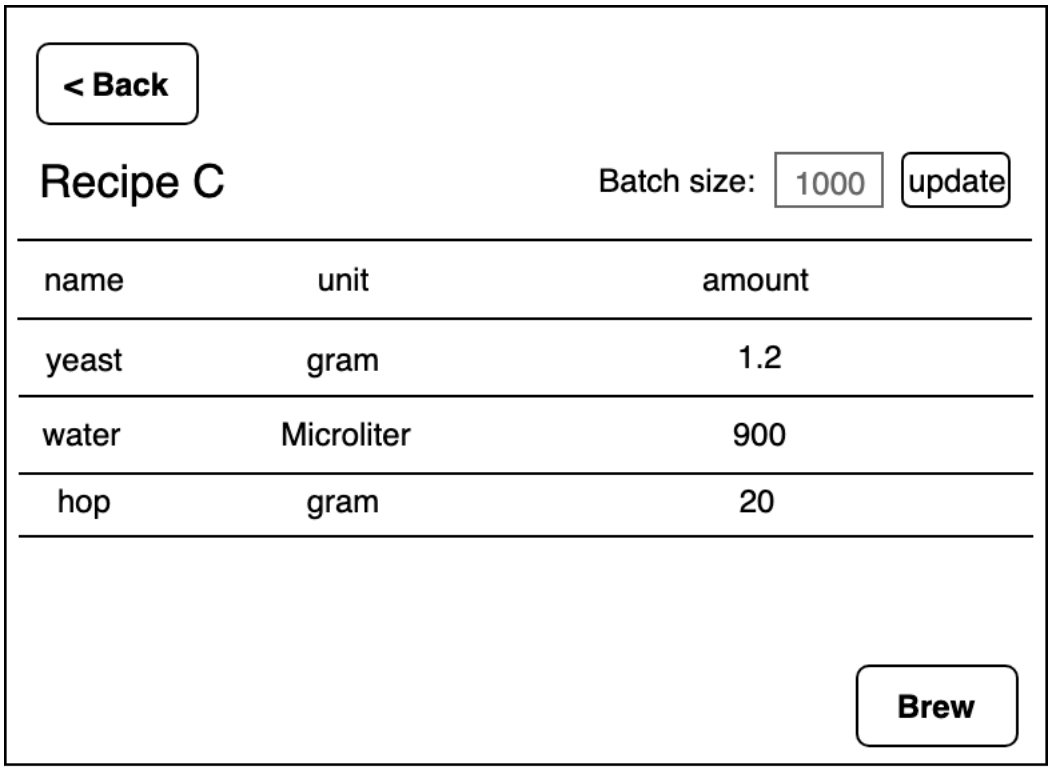


8. Detail page

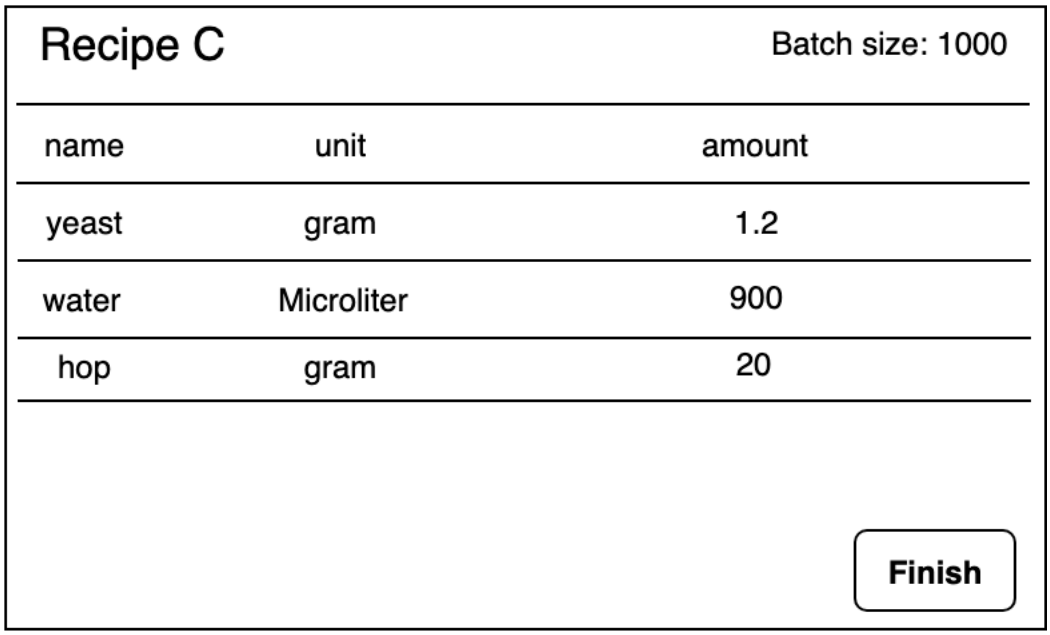
9. Confirm window



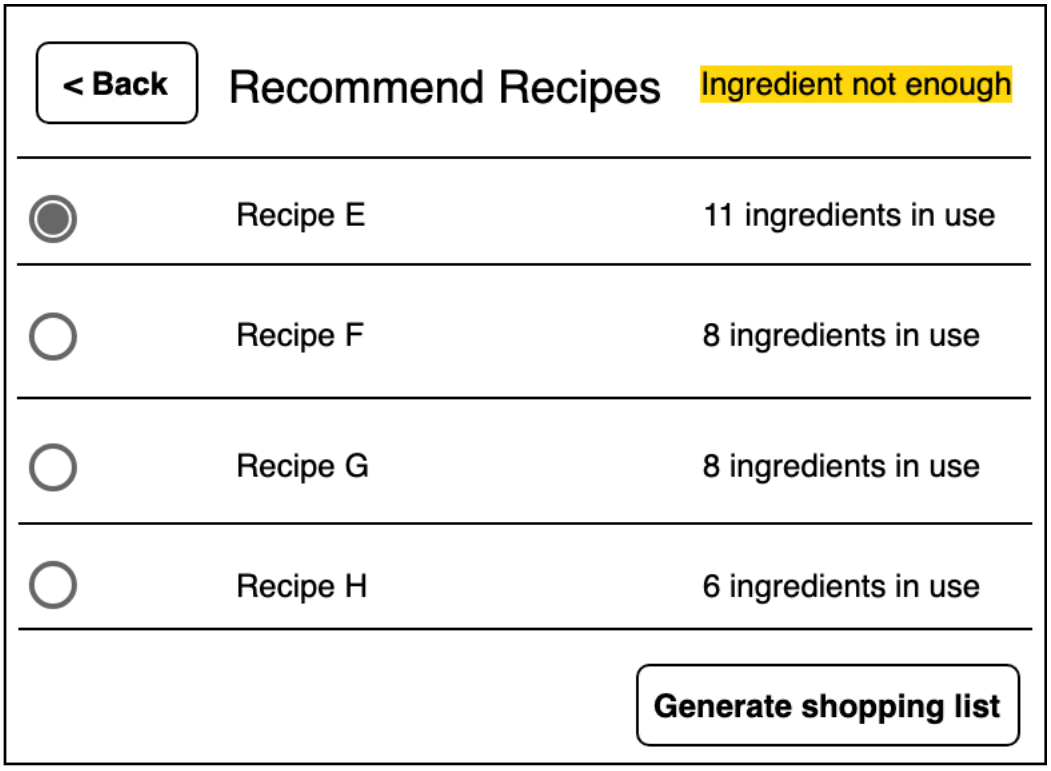
10. Recommend Ingredient list (enough ingredient)



11. Recipe detail page



12. Brew page



13. Recommend Ingredient list (Missing ingredient)

< Back

Missing Ingredients list for Recipe E

name	unit	amount
yeast	gram	3.2
water	Microliter	850
hop	gram	23

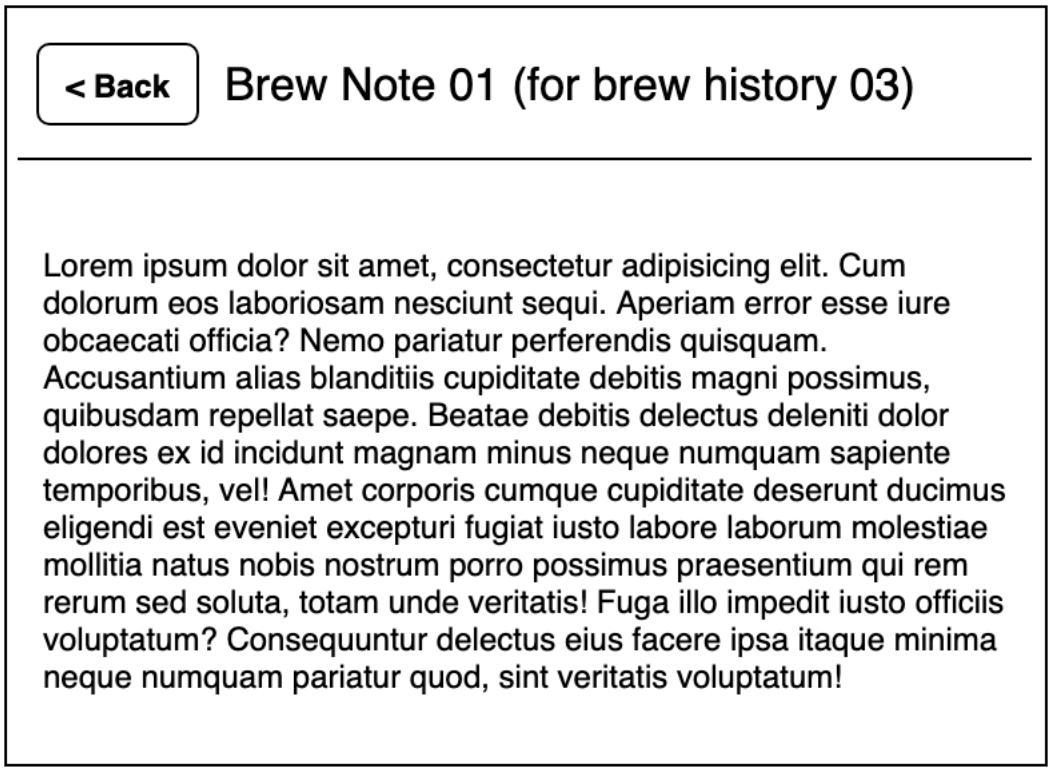
OK

14. Missing ingredients list

< Back Note list Add note

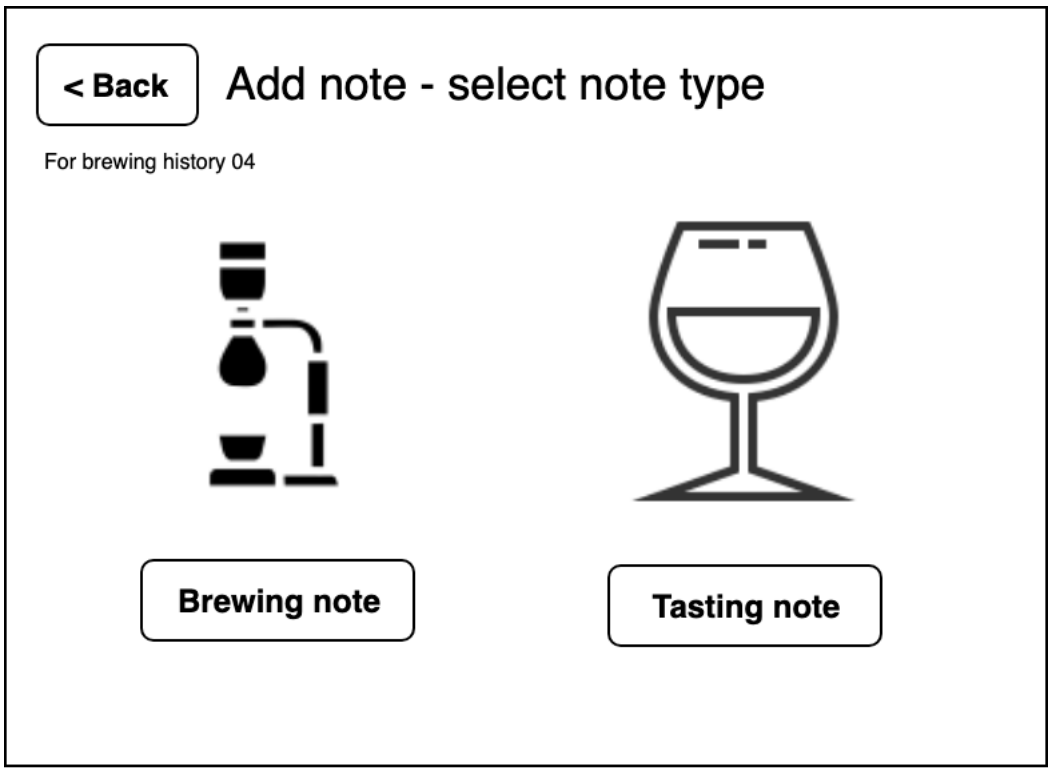
Brew Note 01	for brew history 03
Brew Note 02	for brew history 04
Taste Note 03	for brew history 03
Brew Note 04	for brew history 05
Brew Note 05	for brew history 05
Brew Note 06	for brew history 06

15. Note list

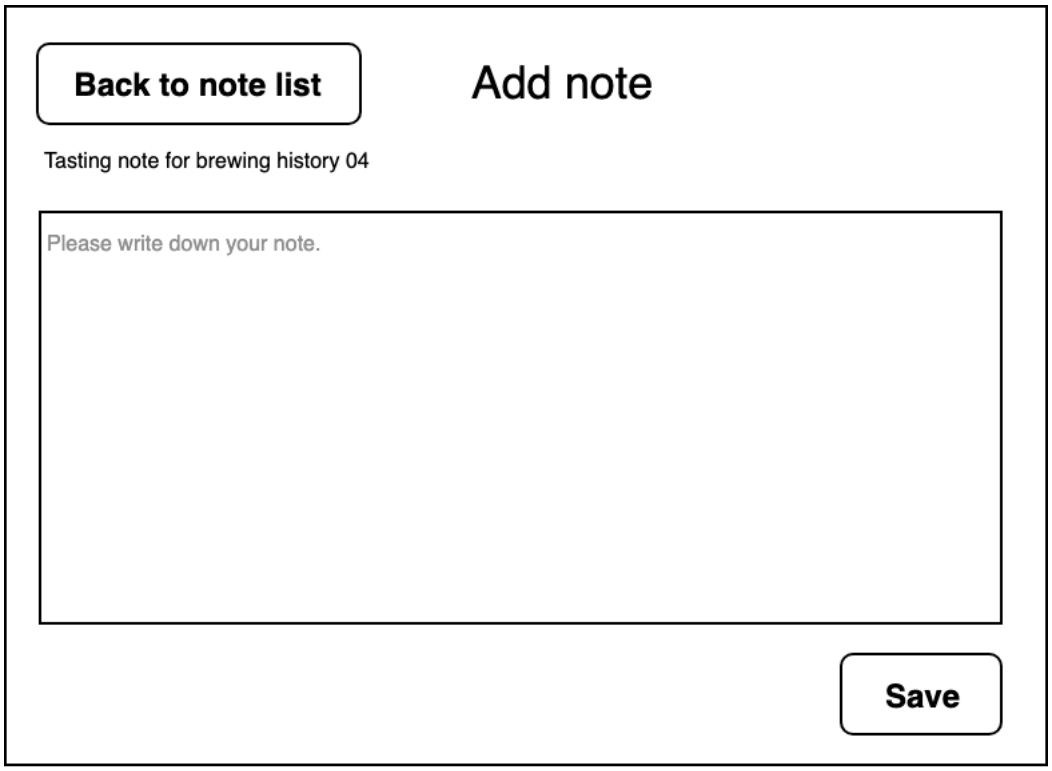


16. Note content page

17. Brewing history list

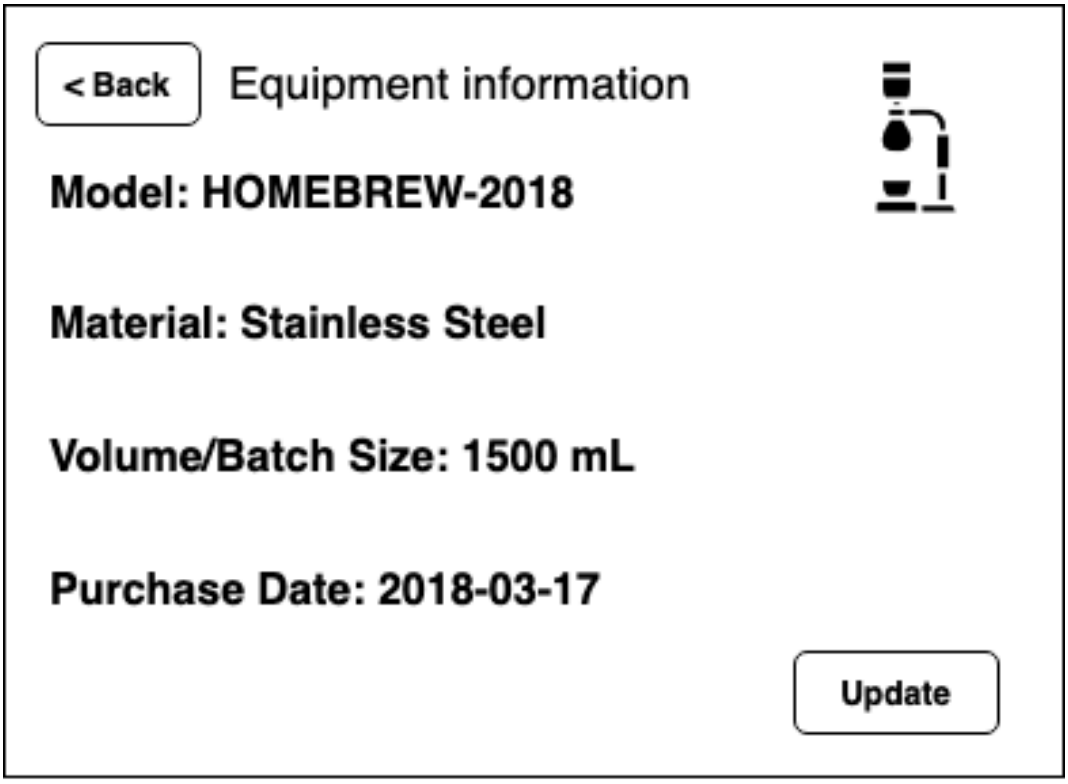


18. Note type selecting page

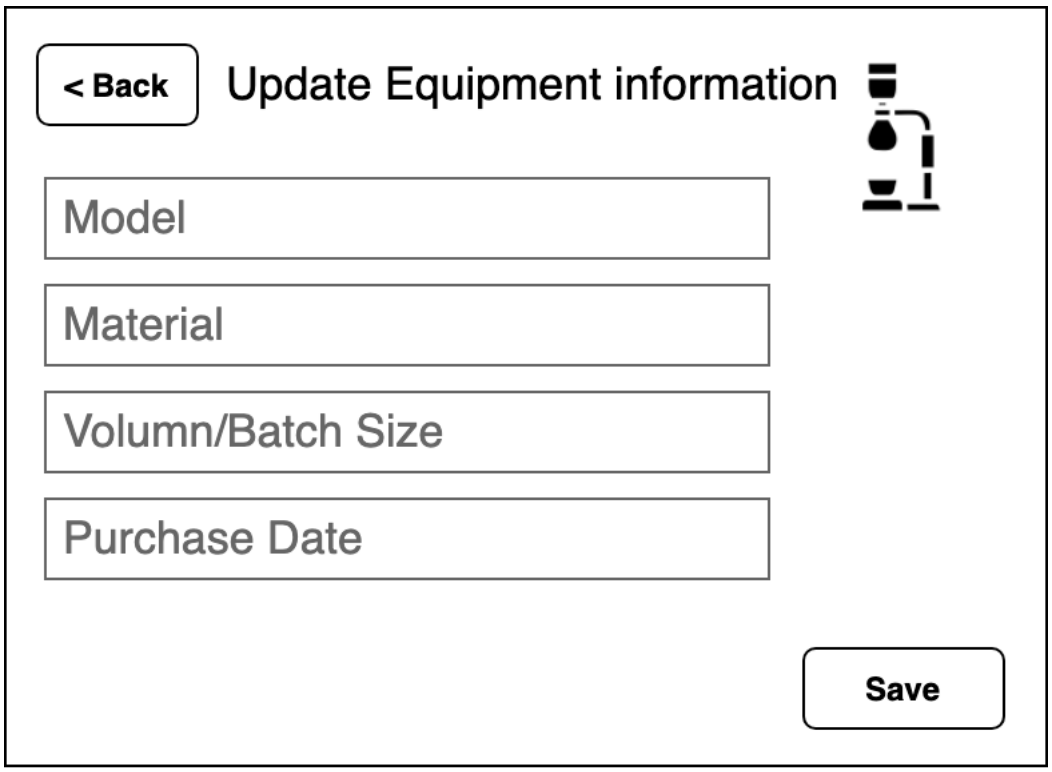


19. Input page

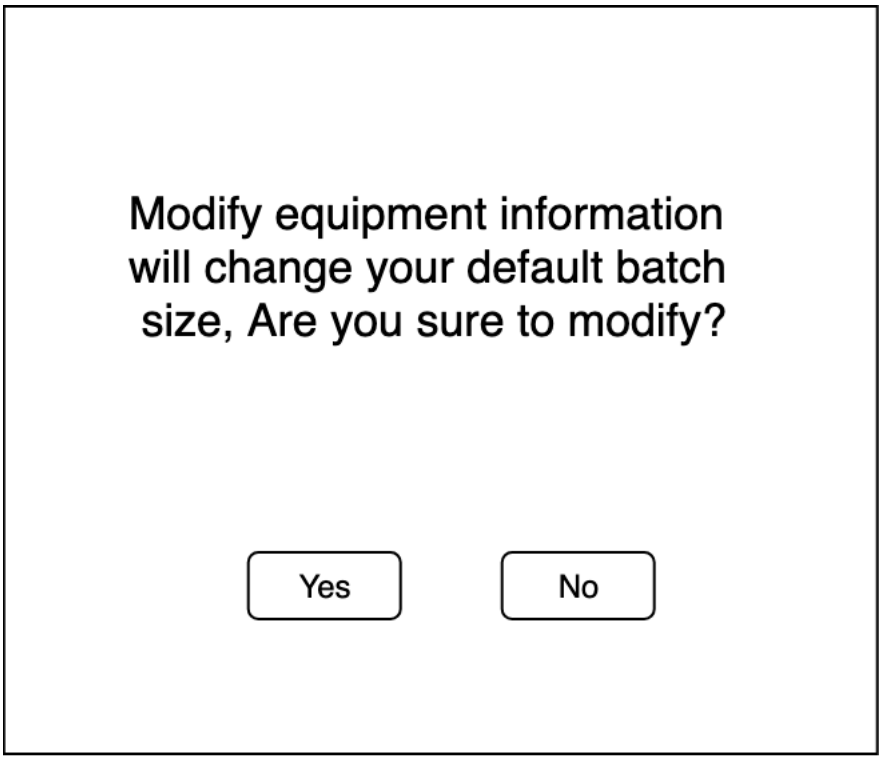
20. Exit confirmation page



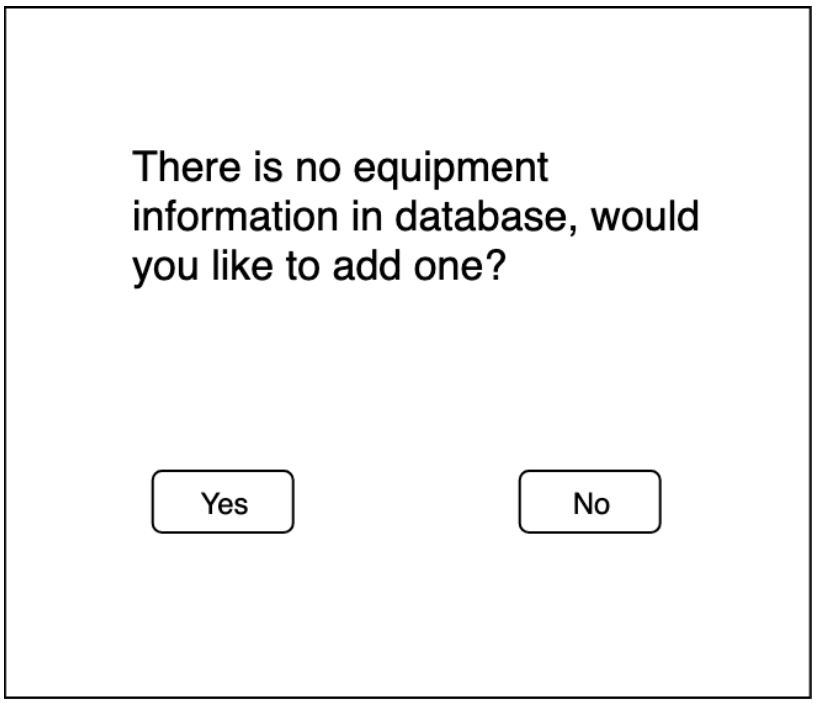
21. Equipment information page



22. Input page



23. Confirm page



24. adding confirm page

4.2 Hardware Interfaces

The software does not have the function to control the hardware.

4.3 Software Interfaces

The software queries the database to display the information displayed on the software user interface, and the user can operate the software user interface to perform database addition, query, deletion and modification.

4.4 Communications Interfaces

This software is an offline software which means communication function will not be provided.

5. Other Nonfunctional Requirements

5.1 Performance Requirements

The response of search operation should be less than one second, other operations should be less than two seconds.

5.2 Safety Requirements

The system will not check whether the user's recipe is safety or not. Users should be responsible for their own recipe.

5.3 Security Requirements

Password protection is not provided in the software.

5.4 Software Quality Attributes

Maintainability: Detailed documents would be written to ensure the software could be easily maintained in the future. Each method has comments to help developer to understand.

Robustness: The software has a Mean time to failures (MTTF) for 2000 hours per failure.

Usability: The software will display all the functions directly on the window and would be easy for users to learn to use. Each function should be able to access within 10 clicks.

Reusability: With modular design, some functions of the system could be reuse on other software.

6. Other Requirements

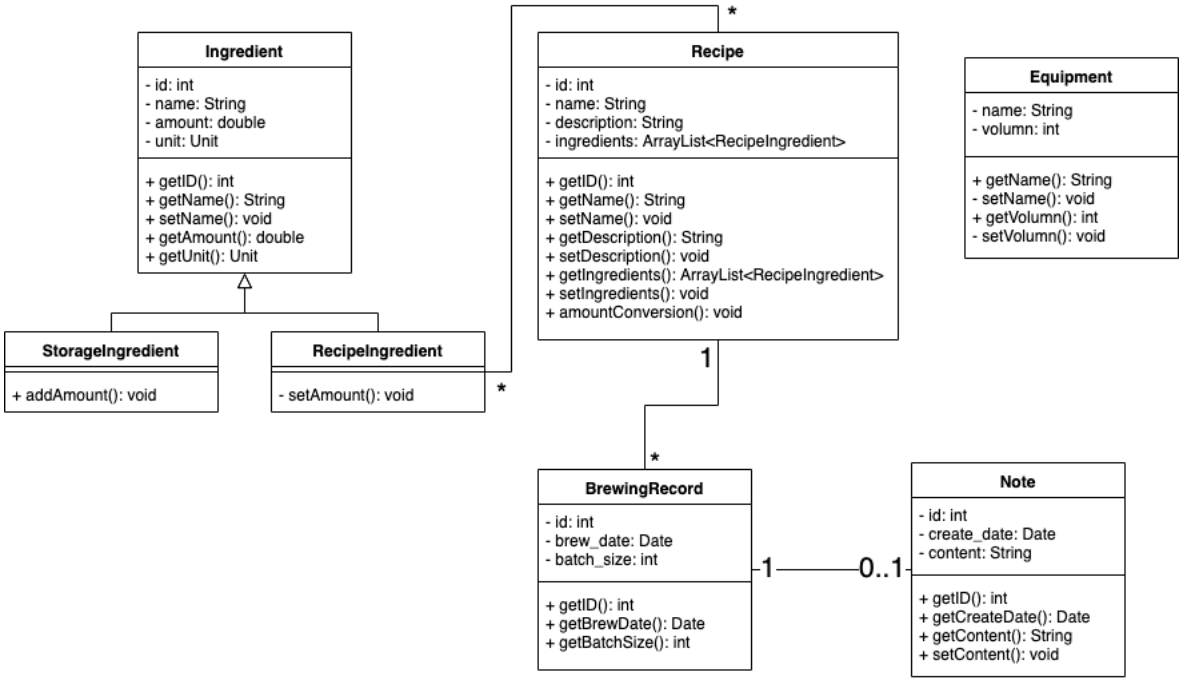
TBD

Appendix A: Glossary

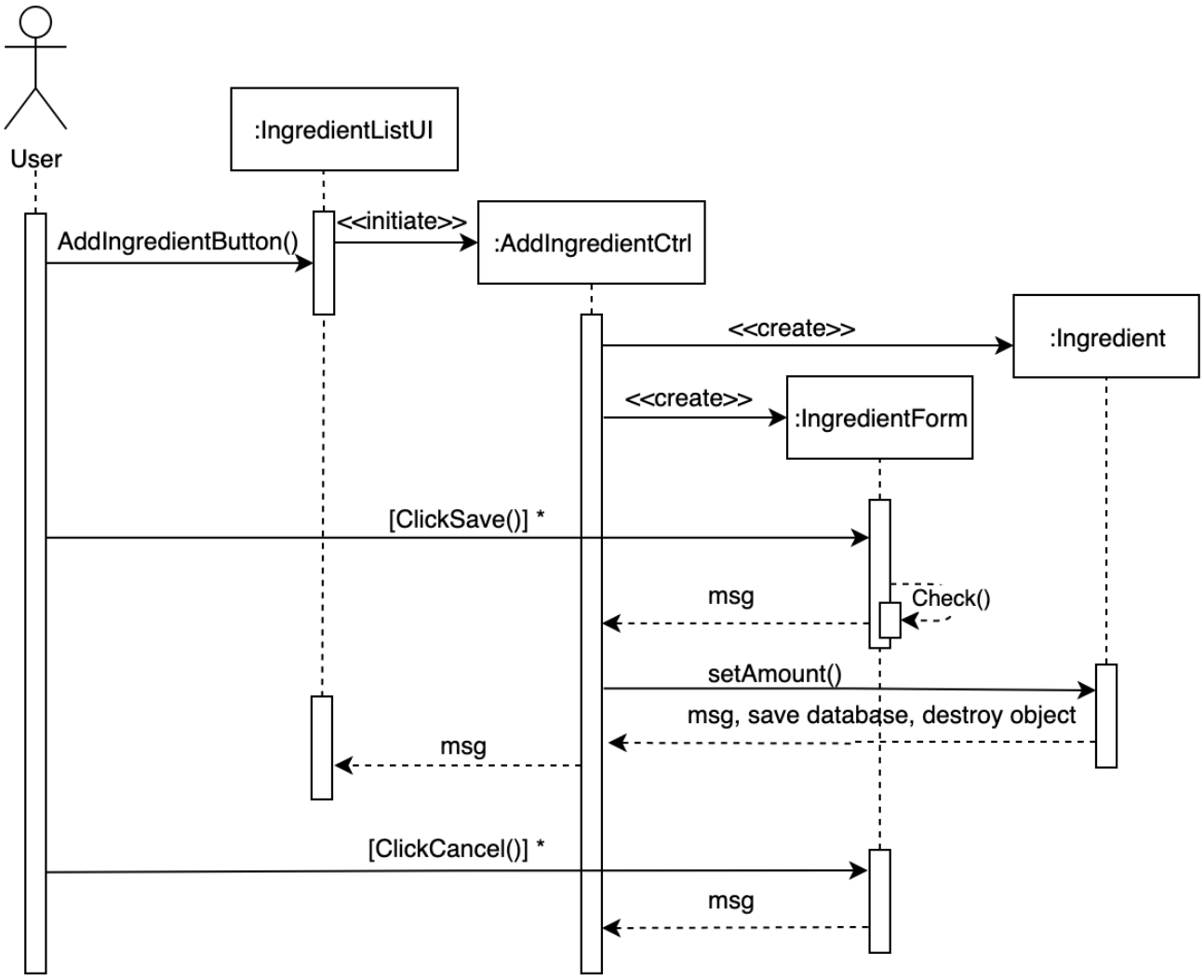
- N/A: Not Applicable
- TBD: To be determined

Appendix B: Analysis Models

Class Diagram



Sequence Diagram: “Add” for use case in maintaining ingredients.



Appendix C: Issues List

TBD