Recipe Book

---

https://github.com/SewonKim0/OOP-Final-Project

Group#18
Elizabeth Lu
Sewon Kim
Abigail Zhou

12/11/2024

Table of Work

(Please write x in the boxes to mention what each student achieved in this project)

	Elizabeth Lu	Sewon Kim	Abigail Zhou
Project Description	X	X	X
Uses Cases Diagram(s)	X		
Sequence Diagrams	X	X	X
Class diagram(s)	X	X	X
Implementation	X	X	X
Conclusion		X	

Table of Contents

System Analysis – Project Description

General Description, Goals and Benefits

The Recipe Book is an application designed to help users manage and store their recipes with functionalities such as adding, browsing, viewing, editing, and deleting recipes. Each recipe contains fields for name, cook time, rating, ingredients, preparation steps, notes, and the last updated date. After creation, any preexisting recipes can be modified using the edit button, and users can change the recipe's rating using the upvote and downvote buttons. All changes made can be saved using the save button. The application allows users to search for recipes by name or by ingredients and also sort the table based on column of choice. The goal of this application is to provide a platform for people looking to store their recipes in an intuitive and easy-to-use application. The application allows for ease of access via search and filtering functions, making it easy to both store and retrieve recipes. The tool benefits individuals who enjoy cooking and need a platform to manage and store many recipes.

System inputs and outputs

Inputs:

- Program reads recipe book data from "backend/recipe_book.json"
- User interactions via Java Swing GUI:
    - keyboard input (typing in name, cook time, ingredients, steps, notes, and rating via text fields)
    - mouse input (clicking buttons, selecting recipe)

Outputs:

- A table, with each row representing a recipe
- Program writes recipe book data to "recipe_book.json"

Special requirements

Setup and Steps To Run Project:

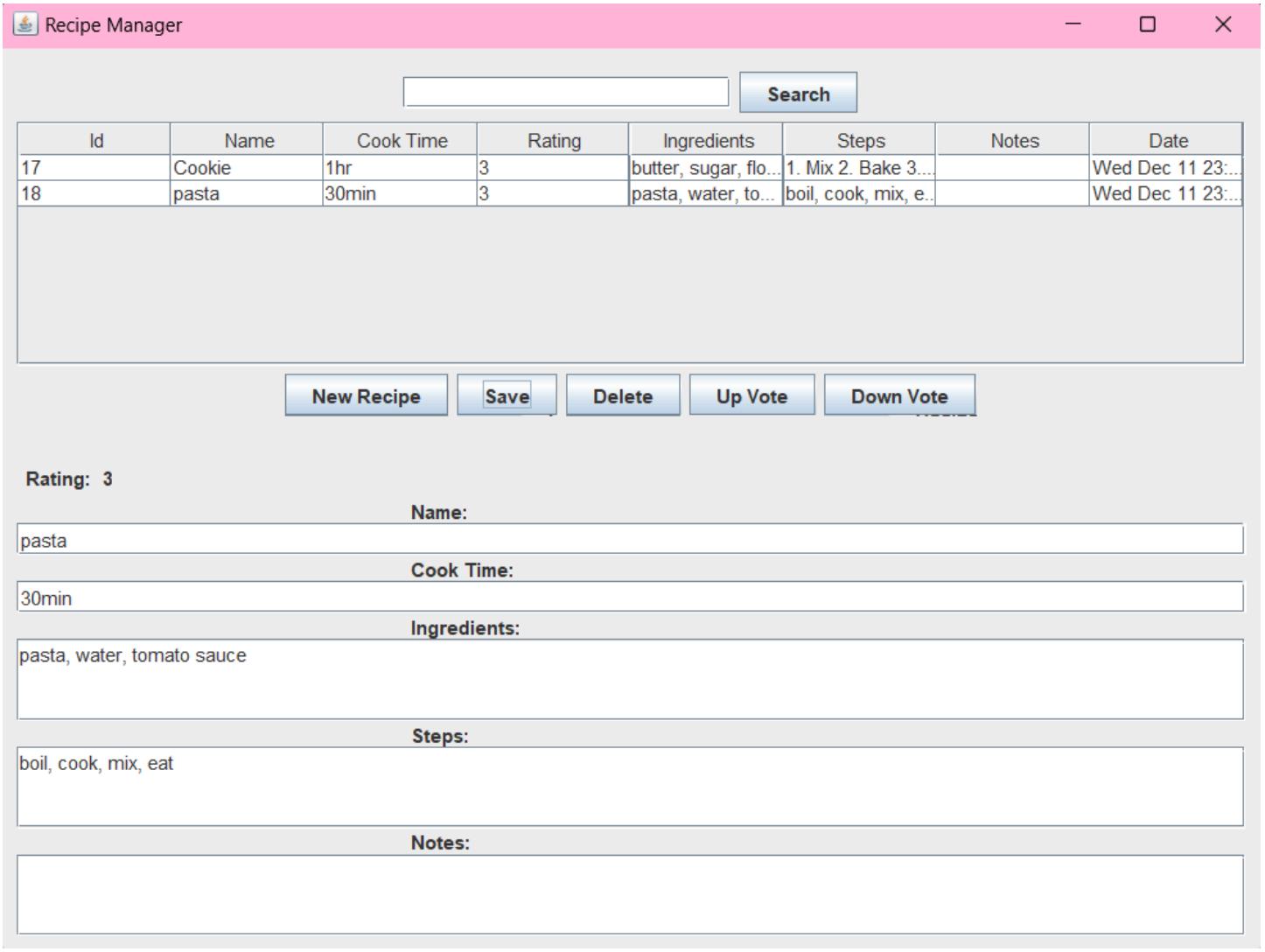
This project will use Google's Gson library in order to serialize and deserialize java objects into JSON format. Therefore, when running the project, it is necessary to include this dependency (included as a .jar in the repository) as part of the classpath.

In order to run the java project from the command line, run the following procedure.

- Navigate to the project root directory in terminal
- Run "javac -cp libs/gson-2.11.0.jar -sourcepath src -d out src/frontend/Frontend.java" in order to compile the java code and output it to a separate /out directory.
- Run "java -cp "out;libs/gson-2.11.0.jar" frontend.Frontend" in order to run the outputted compilation.
- Afterwards, a GUI window for the application will show up.

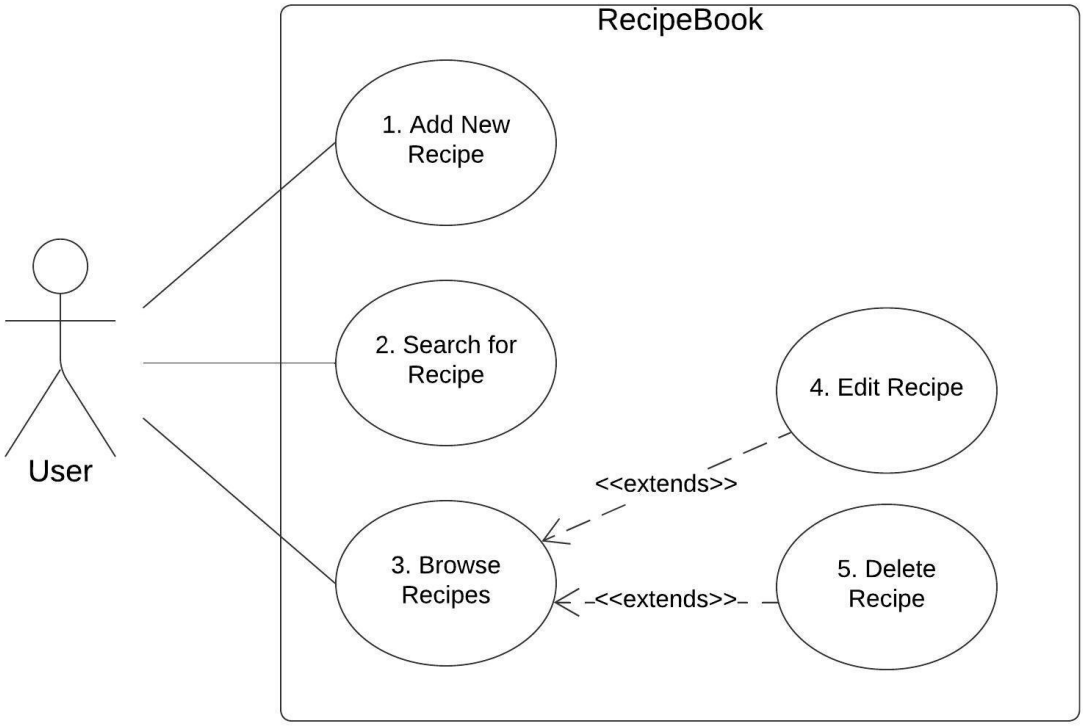
System Analysis – Project Description

GUI



System Analysis – Project Description

Use Case Diagram



Use Case Descriptions

1. Add a New Recipe	
Overview	The user clicks on the “New Recipe” button to open an empty Recipe Form. The user must fill out the required fields “Name”, “Ingredients”, “Steps” before the user is allowed to save. Once the user is done filling out the form and the required fields are not empty, the user clicks the “Save” button to successfully add the new recipe.
Related Use Cases	
Actors	User

2. Search for Recipe	
Overview	The user types in a query into the Search Bar at the top of the page and clicks on the “Search” button. The list of recipes with related names or ingredients is displayed in the recipe list table.
Related Use Cases	
Actors	User

3. Browse Recipes	
Overview	User scrolls through the list of all saved recipes (displayed alphabetically), including their name, date, rating, and total cook time as individual columns. The user is able to sort the recipes based on a specific column by clicking on the column header.
Related Use Cases	4. Edit Recipe, 5. Delete Recipe
Actors	User

4. Edit Recipe

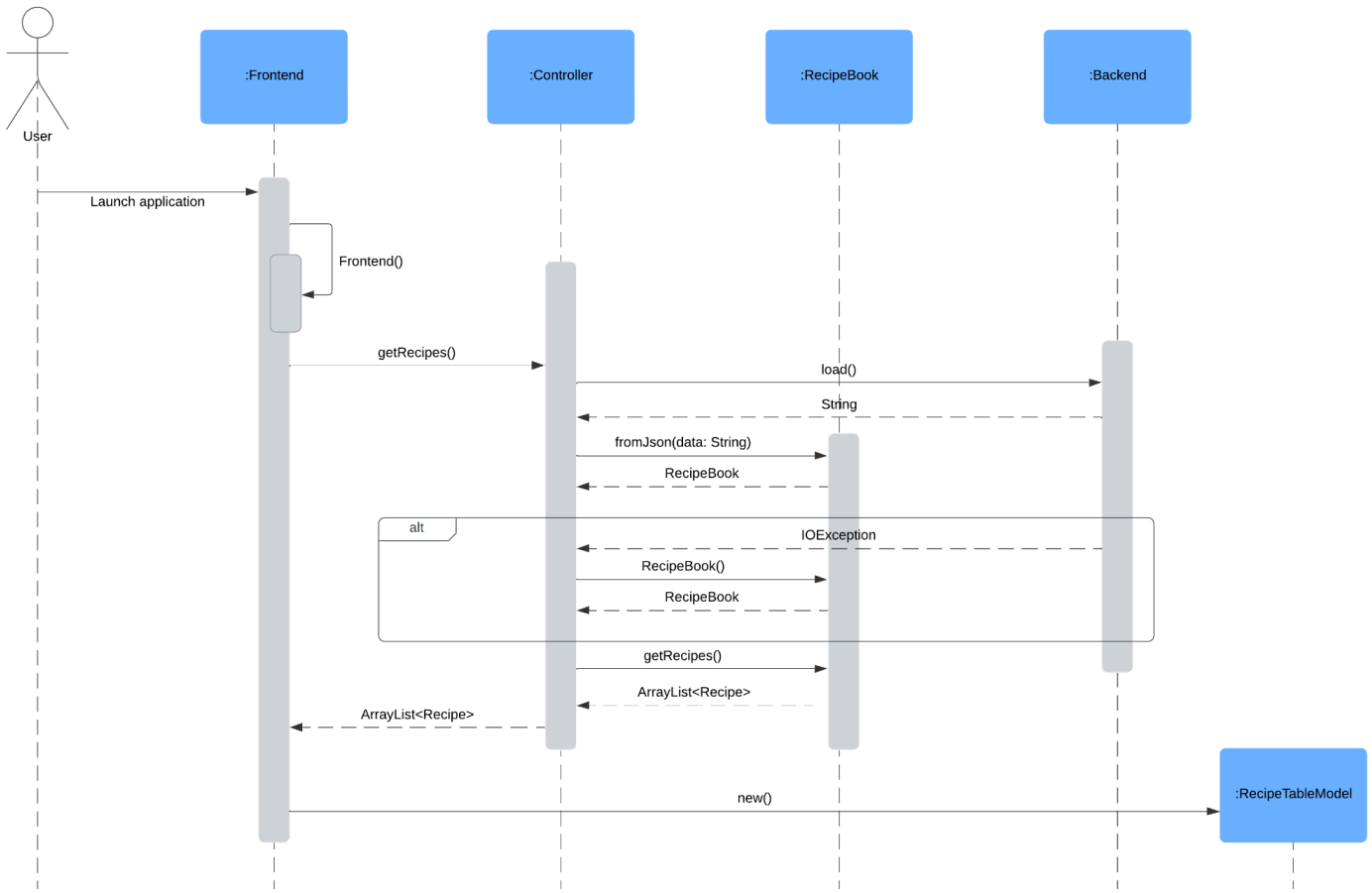
Overview	User selects an existing recipe from the Recipe List, which will display the recipe contents in the Recipe Form. The user can edit any field of the recipe. The required fields cannot be modified to be empty and in the case that they are, the save will fail. User clicks the “Save” button when finished editing and the changes will be saved. Upon successful save, a modal will pop up confirming the successful save.
Related Use Cases	3. Browse Recipes
Actors	User

5. Delete Recipe	
Overview	User selects an existing recipe from the Recipe List, which will display the recipe contents in the Recipe Form. The user clicks the “Delete” button to delete the current recipe. Upon successful deletion, a modal will pop up confirming the successful deletion.
Related Use Cases	3. Browse Recipes
Actors	User

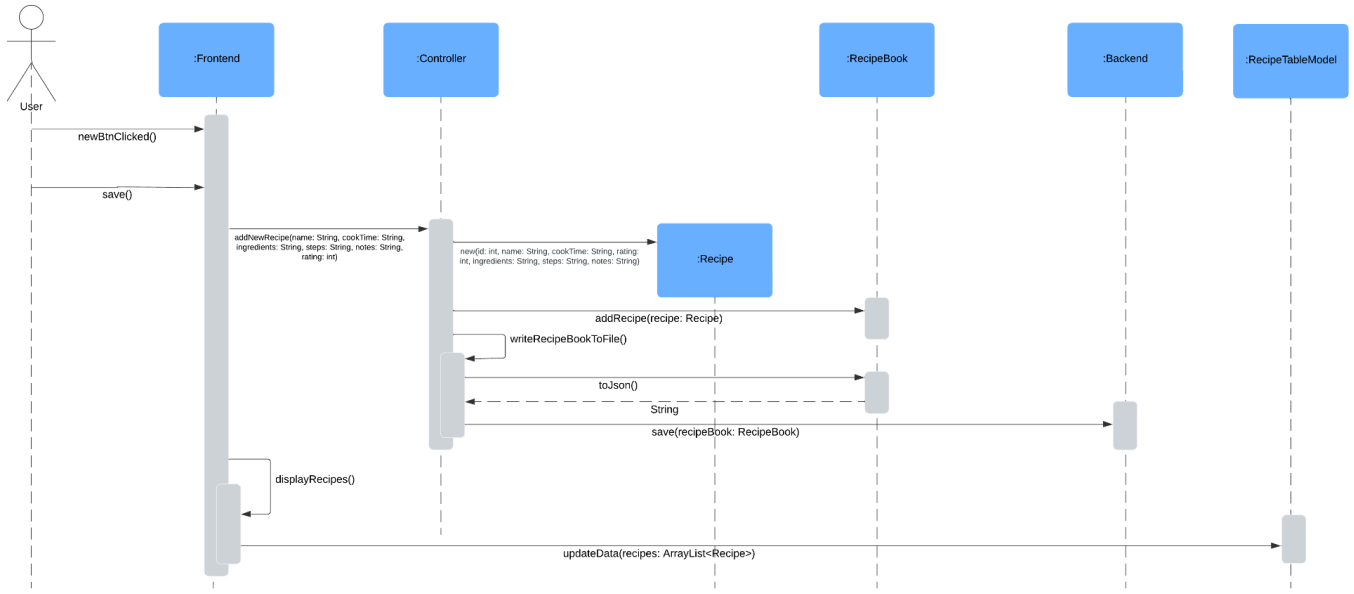
System Design

Sequence Diagrams

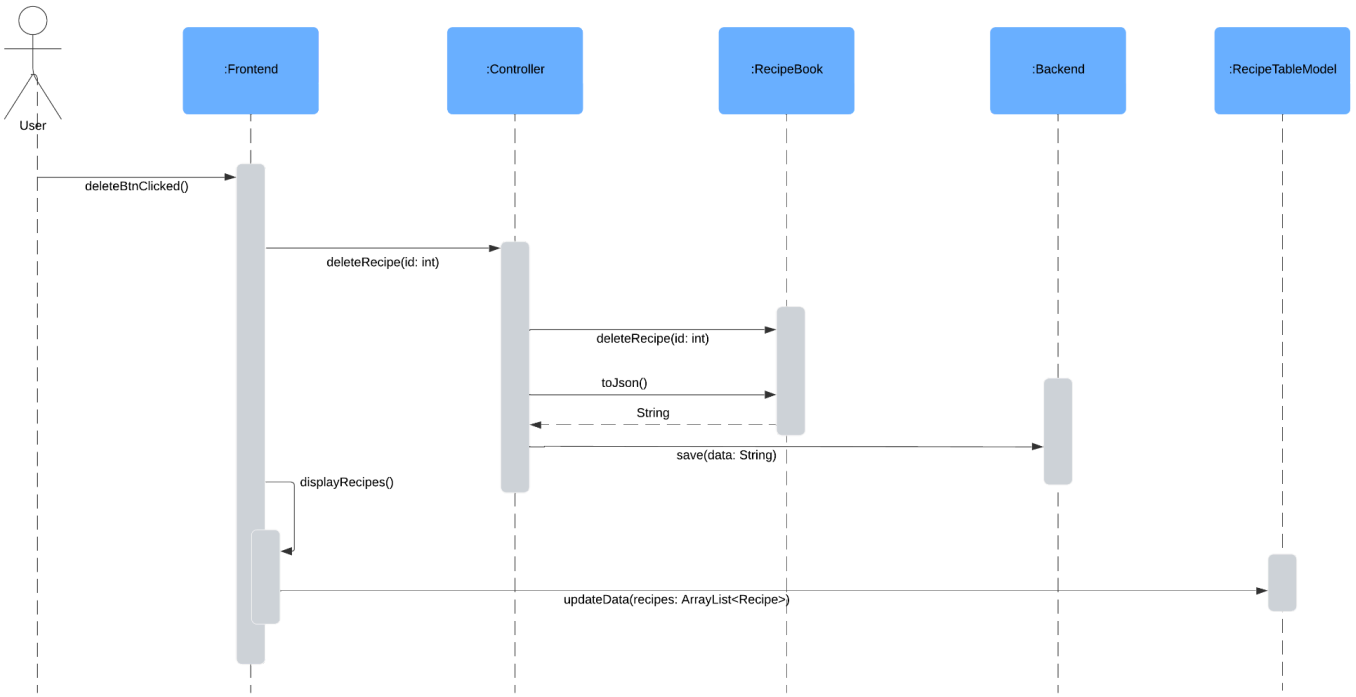
(Launch) Sequence Diagram



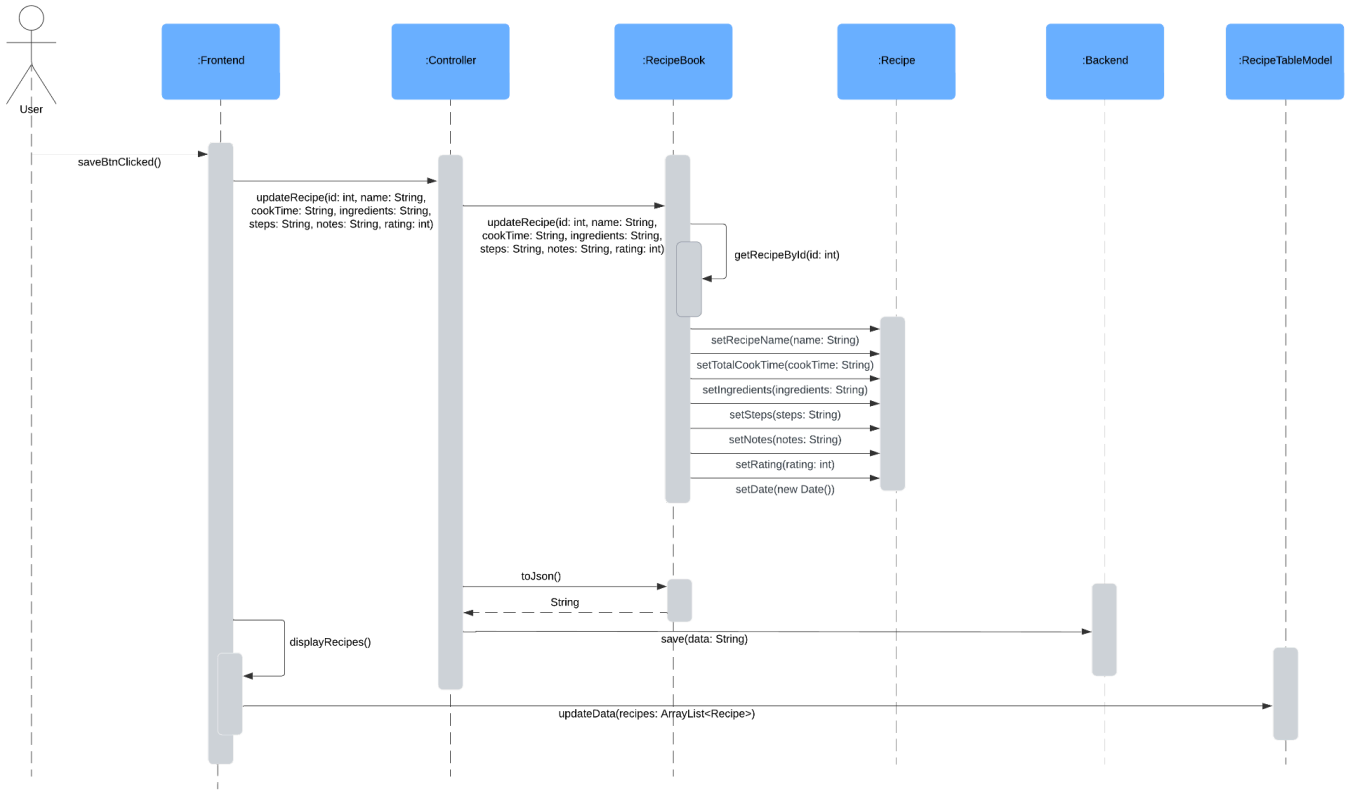
(Add New Recipe) Sequence Diagram



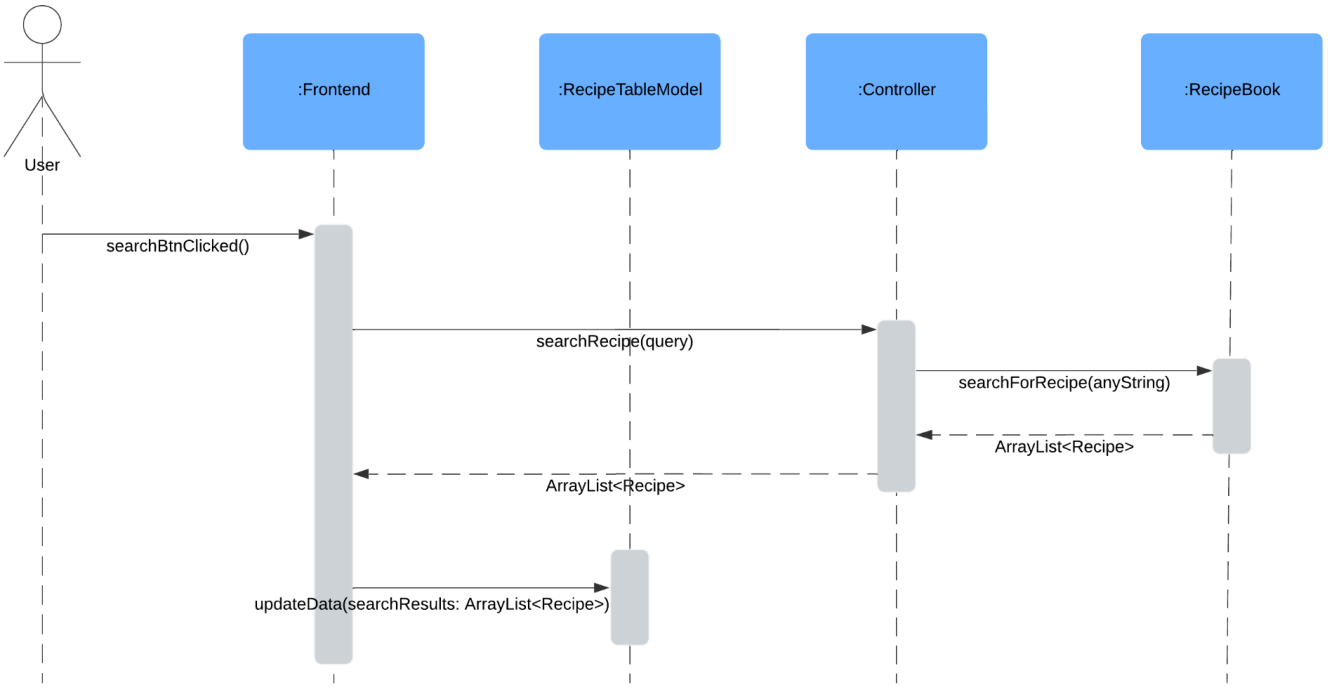
(Delete Recipe) Sequence Diagram



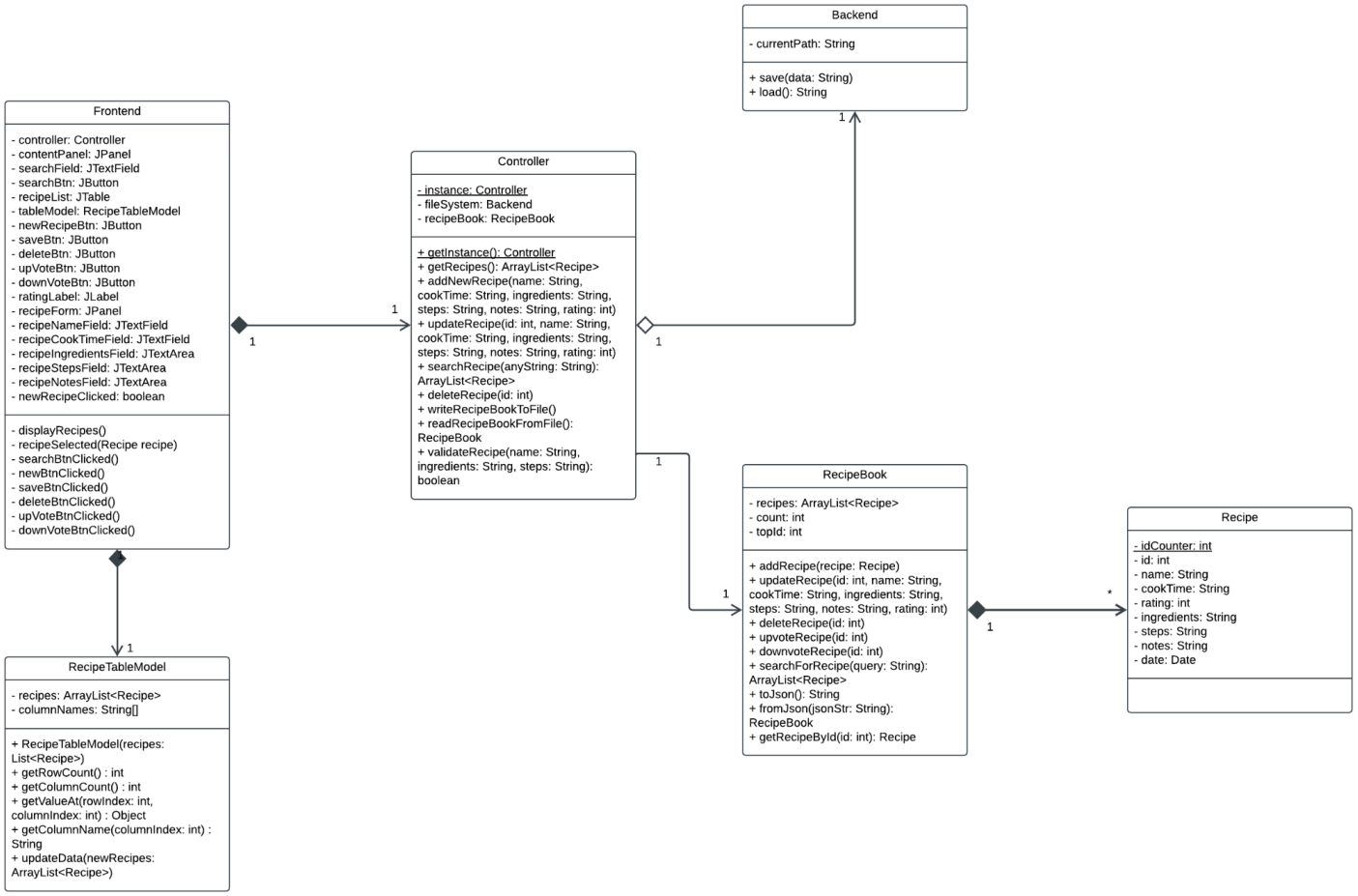
(Edit Recipe) Sequence Diagram



(Search for Recipe) Sequence Diagram



Class diagram



Conclusion

This concludes our reflection of our java project. We built an object oriented application to facilitate the convenience of storing and retrieving detailed information on a list of recipes. Through our intuitive user interface and controls, we hope that users will be able to easily navigate through and utilize the functionalities offered by our application.

By following the principles and design patterns of object oriented programming, such as the Singleton design pattern, we were able to build a highly readable and maintainable codebase that not only utilizes memory efficiently but also provides potential for future developers to add on to our work. We have enjoyed designing and developing for this application and hope to make use of it in our daily lives.