Software Requirements and Design Document

For

Group <8>

Version 1.3

Authors:
Alex J
John F
Justin N
Mason J

1. Overview (5 points)

As per the last RD our current system runs using HTML5 on the front end for user interaction and bootstrap was implemented to style the website as well. On the back end flask was used to store and pull data .There is one overall database with several tables, one which keeps track of the recipes, another to store ingredients associated with each recipeID, the third table stores each step related to a RecipeID. There is a table which also stores ingredients and the measurements for that recipe. The system is simplistic in nature and should be easy for anyone to use once they just pick it up. There are 4 main functions to the app.py, Add a recipe, search for a recipe, add ingredients and display all ingredients.

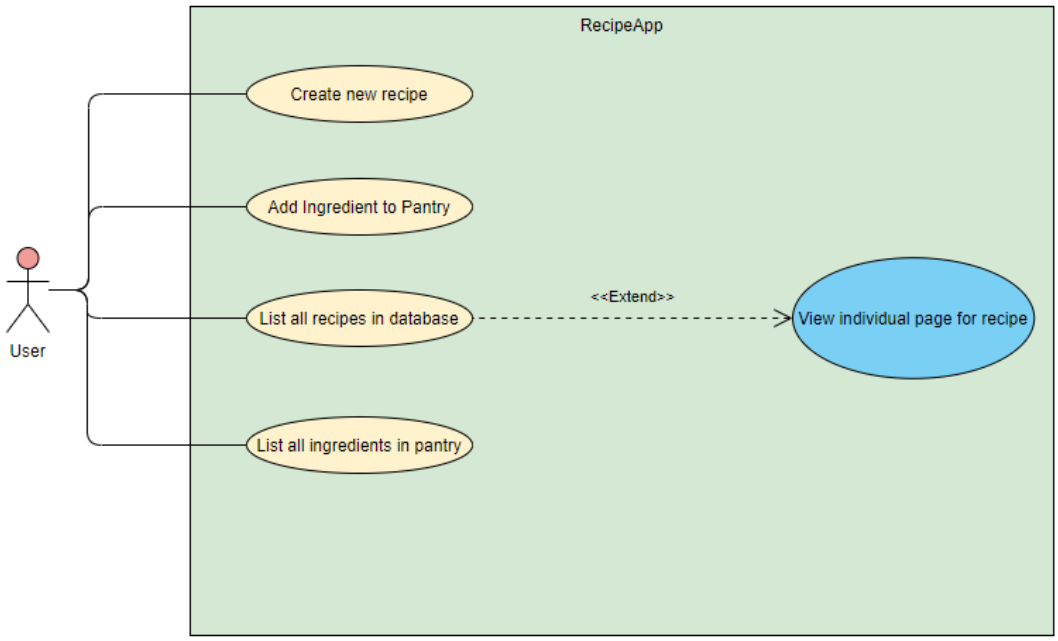
2. Functional Requirements (10 points)

1. System should allow user to submit a recipe to the database (High)
2. System should allow user to search for recipes by name (high)
3. System should allow user to list the entire recipe database (High)
4. System should allow user to submit ingredients to the pantry database (High)
5. System Should allow user to view all ingredients that have been entered into the pantry table (High)

3. Non-functional Requirements (10 points)

The application touches basis with everything it needs to do, but is not large enough where efficiency is of utmost importance. No new functions are needed outside of filling the database with recipes and making sure that we can now account for improper input from users.

4. Use Case Diagram (10 points)



Create new recipe: The user can create recipes from scratch by inputting data for the recipe name, the text for the steps of the recipe and the list of ingredients used in the recipe. Individual

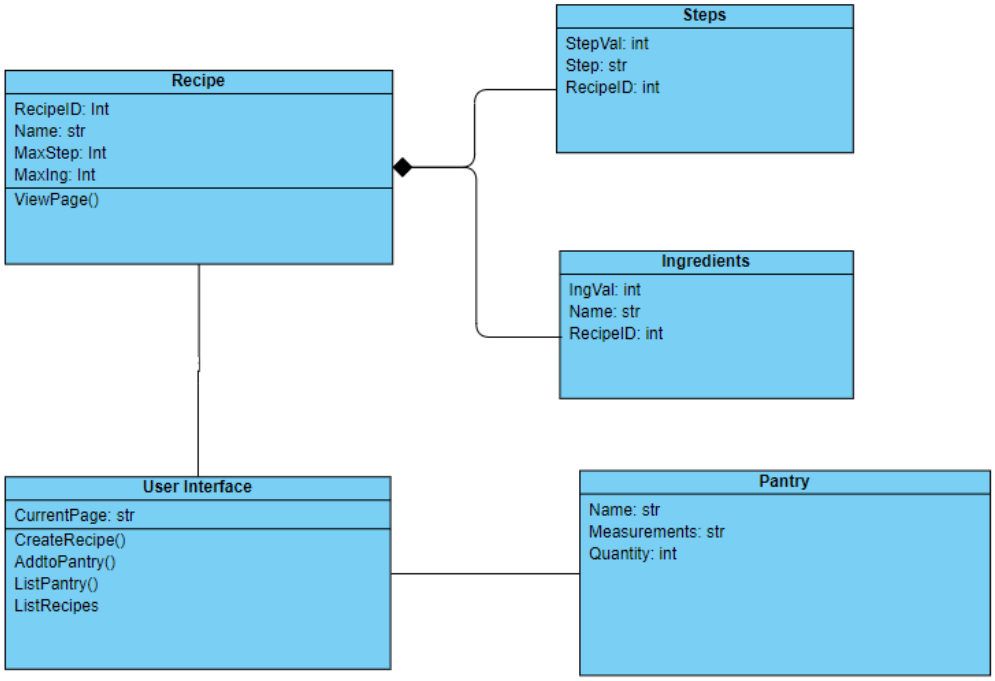
steps and ingredients are divided by ‘\n’ characters in their respective text boxes while the system tracks the RecipeID across the three tables being used for this use case.

Add ingredients to Pantry: The user has full freedom in adding ingredients to their pantry, tracking the ingredient name, the quantity of said ingredient and the metric by which the quantity is being measured.
List all ingredients in pantry: User is brought to a page that lists all existing ingredients contained in the pantry as well their current quantity.

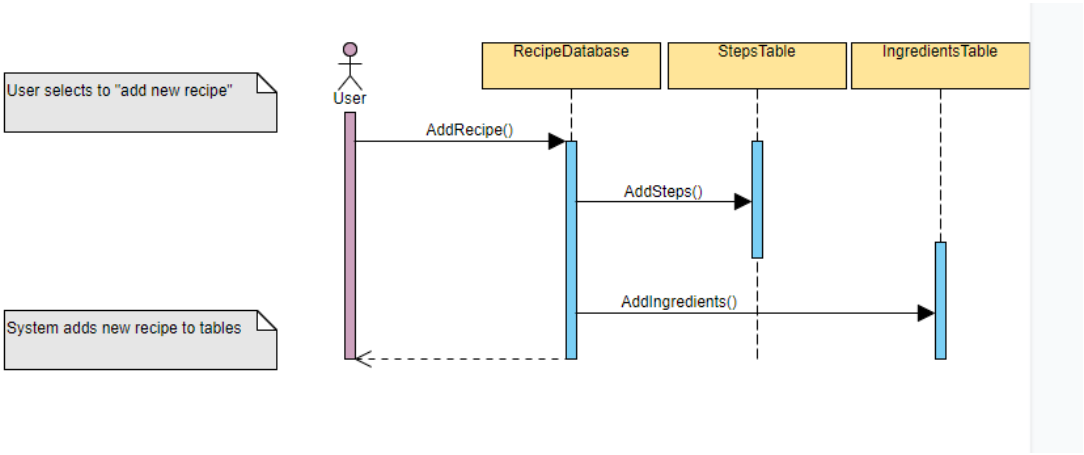
List all recipes in database: User is brought to a page to list all existing recipe names, and will have the option to click said names to extend into the use case for viewing an individual recipe page.

The individual recipe page will list the recipe name as well as an organized view of the steps and ingredients of the recipe given for the use case.

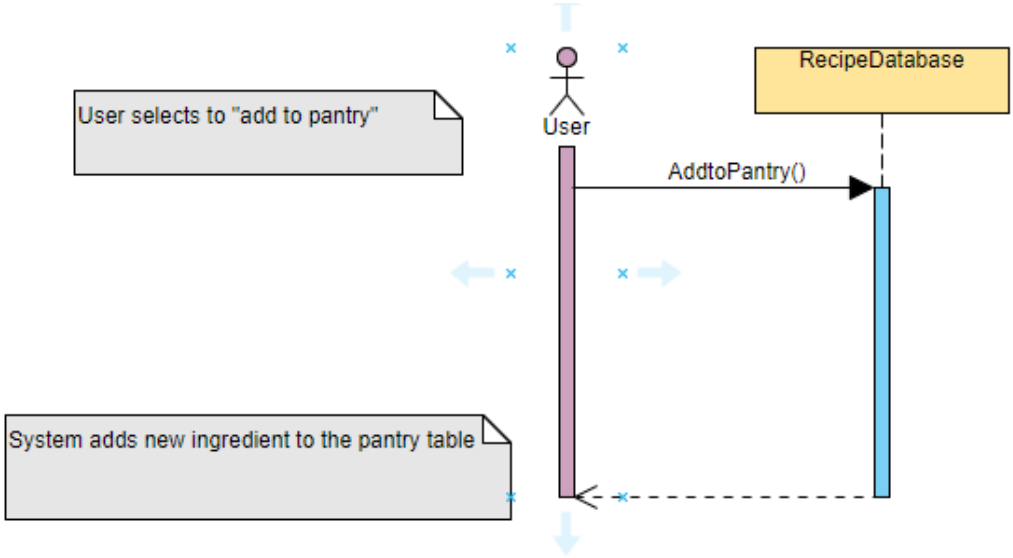
5. Class Diagram and/or Sequence Diagrams (15 points)



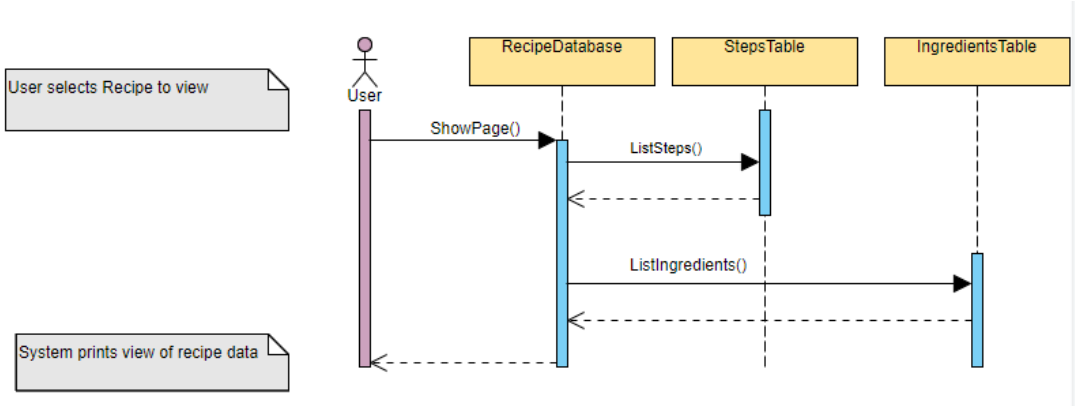
Create a Recipe:



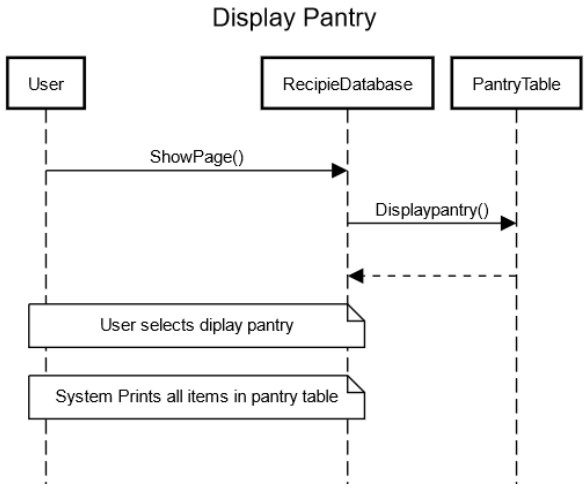
Add to Pantry:



View Recipe Page:



View Pantry:



6. Operating Environment (5 points)

The app.py uses html5 on the front end for the website and bootstrap for the detailing. Bootstrap was used for its prebuilt libraries and ease of use. Flask ver 2.0.1 is used on the back end to talk to between the databases used to store information.

7. Assumptions and Dependencies (5 points)

An external dependency the project relies upon is the use of the online bootstrap library. Most css related attributes are prebuilt and pulled from an online source. So if that online repository were to go down then so would the website's css styling. Another factor to consider is if users input their instructions in the correct manner, if not this could lead to user input being nonsensical because there is no way to verify what they put in is correct. So user error is a big concern in this phase. An Assumed factor taken into consideration is when the user inputs their own recipes there is always the possibility that the user will try to break the system by inputting wrong data. So in order to handle this, fields that require measurements will be restricted to numbers. Users will be restricted to a scroll bar of ingredients to choose from in order to prevent users inputting false input. In regards to instructions, the user will have to input these themselves due to the fact that recipe instructions cannot be so easily limited as ingredients.