Project Planning and Methodology

Problem Statement:

Figuring out what to cook for every single meal is always a hassle. Even if you already know a bunch of recipes. You can compile your recipes in bookmark tabs or sticky notes in cookbooks, but having to sift through everything is difficult especially if you're trying to plan meals for the week.

Planning and cooking meals can take up a lot of time during the week, even if you already know a bunch of recipes. Recipes can be scattered across cookbooks, blogs, youtube videos, and phone notes. Planning grocery lists around them can also be time consuming as you have to reference what you already have and if there are any overlaps with the recipes.

Product:

The product is a software application that will be a one stop shop for compiling, searching, and filtering recipes. You can upload recipes in a uniform format and search for them based on the name, region, ingredients, etc. You can also store a list of what's currently in the pantry and based on the recipes that are selected, compile a grocery list.

Process:

An Agile process will be used to complete this software design project. Agile values working software over comprehensive documentation, and individuals and interactions over processes and tools. The agile methodology encourages setting intermediary, small goals in a project to achieve them faster, as well as testing and review during development. Usage of agile will ensure that progress on the project is made rapidly and consistently.

Requirements and Analysis Models

Requirements:

Functional

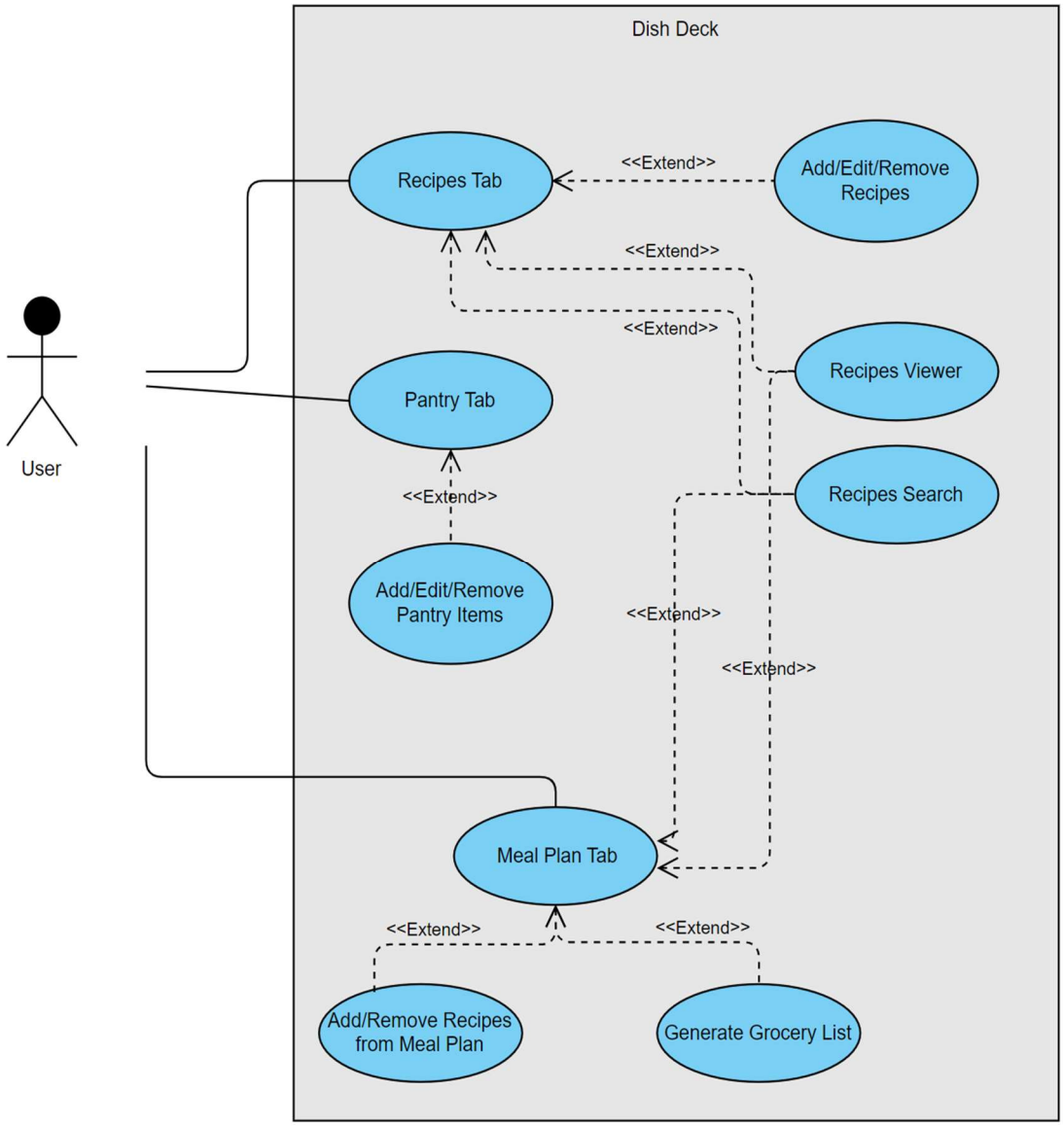
1. Add and store recipes which include:
    a. Ingredients
    b. Description
    c. Instructions
    d. Tags
    e. Pictures
    f. Source
2. Store currently owned ingredients in a ‘pantry’
3. Editing and removal of recipes and pantry items.
4. Recipes are searchable by:
    a. Name
    b. Ingredients
    c. Tags
5. Make a ‘grocery list’ based on recipes selected and current items in ‘pantry’.

Nonfunctional

1. Easily navigable UI (tabs, scroll bar, search bar)
2. Quick way to import recipes.
3. Blog-like viewing page for recipes.

Uses Cases:

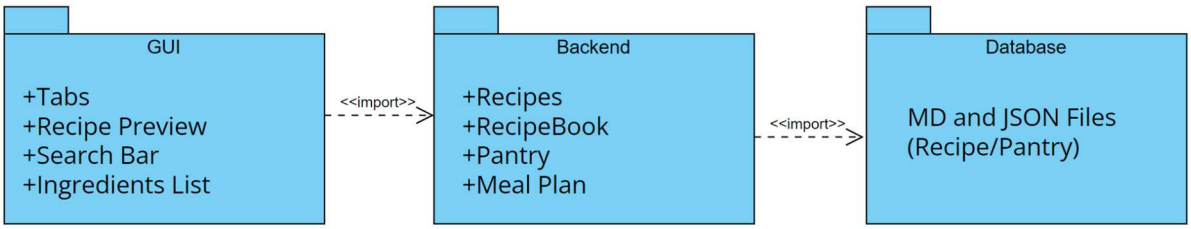
1. User adds/removes/edits recipes
2. User adds/removes/edits pantry items
3. Recipe Search
4. User add/remove recipe to meal plan
5. Generate grocery list
6. View recipe



Dish Deck Use Case UML Diagram

Architecture Design:

Dish Deck is an interactive system and uses an N-tier architecture style.



Architecture Diagram

Component Design:

Recipe

Attributes: [Name, Description, Ingredients, Instructions, Tags, Picture, Source]

Methods:

- __str__()

RecipeBook

Attributes: [recipe, number of recipes]

Methods:

- importRecipe(submethod of loadRecipes)
- loadRecipes() #stores the data in a dictionary
- addRecipes()
- removeRecipes()
- search(field, query) (uses dictionary)

Pantry

Attributes: [Item, Quantity]

Methods:

- load_ingredients()
- add_ingredient(item, quantity, unit)
- remove_ingredient()
- edit_ingredient()
- search()

Ingredients

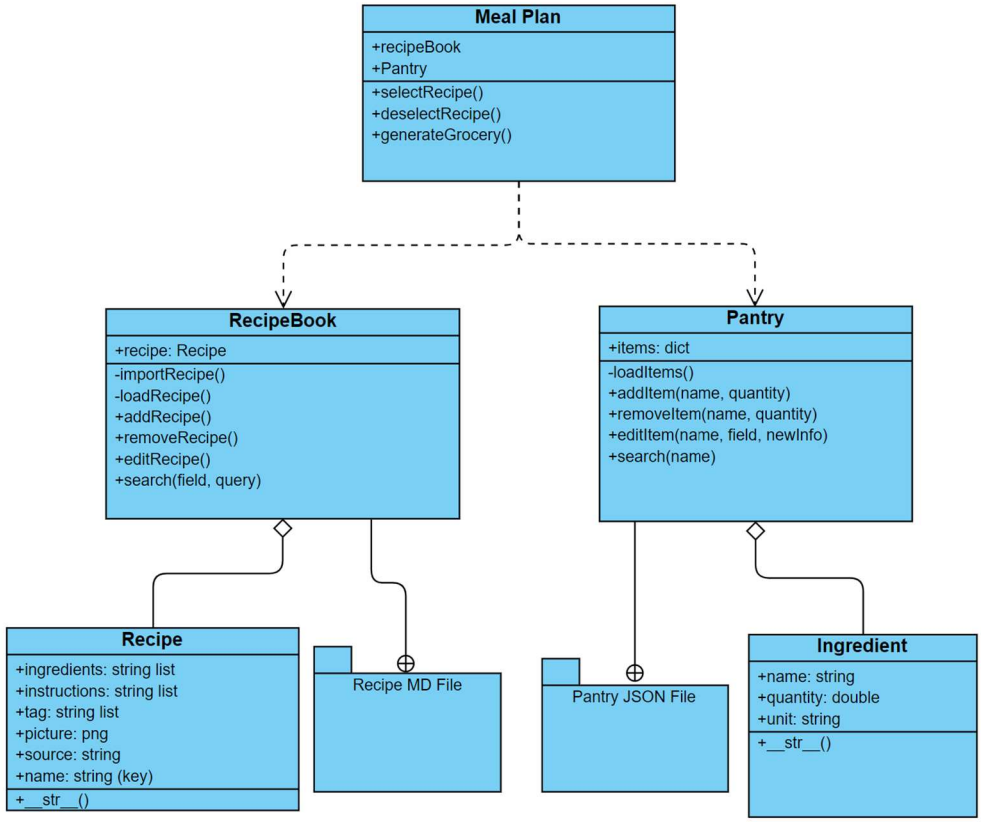
Attributes: [name, quantity, unit]

Methods:

-__str__()

MealPlan
Attributes: [recipeBook, pantry]
Methods:

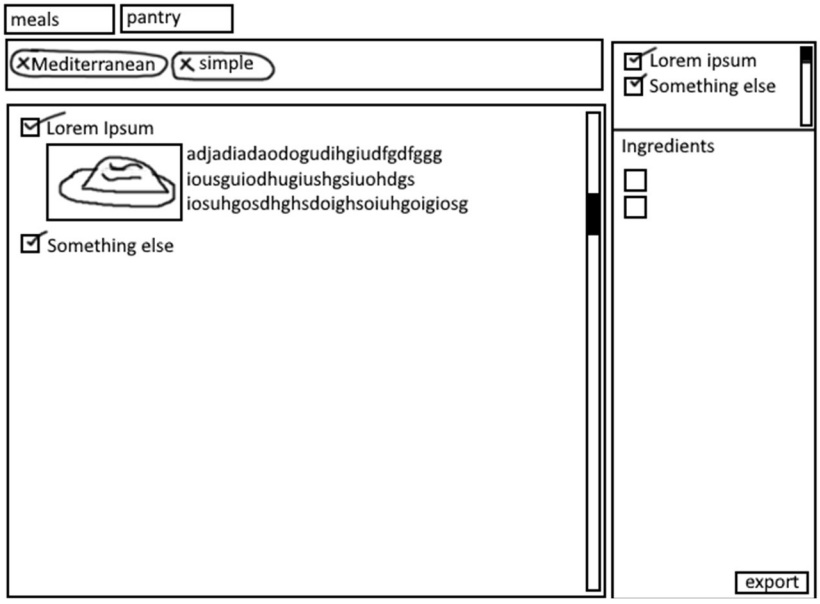
- selectRecipe()
- deselectRecipe()
- generateGrocery()



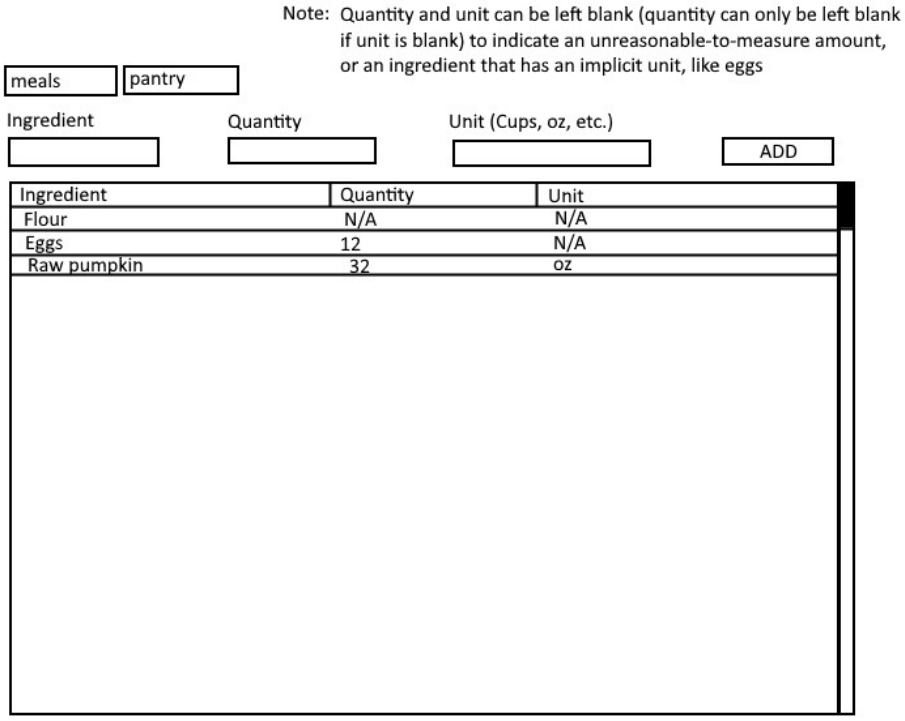
UML Class Diagram

***Note:** Meal Plan class has been allocated to the UI event handling

UX Design:



Meal tab mockup



Pantry tab mockup

Patterns Applied:

Within the context of the UIUX file, which will be our main program file, we apply patterns to the behavior of certain classes. The singleton pattern will be used for the Pantry and Recipebook, as only one instance of those classes will exist throughout runtime.

Plan Project

Estimate Timelines & Milestones

1. Create Github repository
2. Create software configuration
3. Outline of all the classes with parameters and methods
4. Working GUI
5. Functioning backend operations

Analyze Risk

Risks in this project are losing source code and encountering code errors that delay our code completion beyond the due date. To mitigate the risk of losing source code, it will be hosted on a GitHub repository, with protection applied to the main branch. If severe code errors are encountered, agile methodology will be applied to find alternative ways to achieve the desired result.

Quality Management

Software will be reviewed and tested by multiple members of the team. Quality will refer to usability and intuitiveness of the user interface and user experience.

Test Methods

Classes and methods will be tested individually and in specific combinations using the pytest library to ensure functionality. Constant example data is used to test against the software.

Design Reviews

Meetings will be held on design elements. Discussions regarding practicality, effectiveness, and efficiency will be had about each design element. A singleton pattern will be used for the Pantry class and Recipe class, meaning there will be only one instance of each class at runtime.

Security Design, Implementation, and Monitoring

This application poses a minimal security threat to the end user. We will not monitor or store any user's personal information, all user-provided information will be stored on the end user's local

machine. One warning will be included in the product documentation, which is for the user not to include any sensitive information in recipes or ingredients.

Configuration Management

Recipes and pantry files will be within the data folder for the project. Software documentation will be under the doc folder. Source code will be under the src folder. If any changes need to be made to this configuration, approval is needed from the entire team.

System Support

The application will receive updates and new features after production. Feedback from the user will be used to update and fix issues with the UI and usability.

Process Improvement

Our process improvement process involves starting with a small software development organization. By focusing on small parts of the overall system, we're able to avoid the unnecessary complexity of tackling a large problem all at once. Over time the goals and objectives can become more complex, but the goals must be specifically defined.
Currently there are no national standards that need to be upheld for this project. But in the future, as more features are added and the user base expands, adopting these standards will be an important step for process improvement. Agile allows quick development and changes to our process.