SWE 3313

Software Design Document

Mobile Cooking Companion

Team 6
Jeremy Ray, Thomas Roberts, Vitor Santos, Joshua Schoen, and Katlin Scott

Version: (4) **Date: (11/07/2021)**

TABLE OF CONTENTS

1. INTRODUCTION page 3
   - 1.1 Purpose
   - 1.2 Scope
   - 1.3 Overview
   - 1.4 Reference Material
   - 1.5 Definitions and Acronyms

2. SYSTEM OVERVIEW page 4

3. SYSTEM ARCHITECTURE page 4-5
   - 3.1 Architectural Design
   - 3.2 Technical Design
   - 3.3 Design Rationale

4. DETAILED DESIGN page 5
   - 4.1 Class diagrams

5. DATABASE (DATA) DESIGN page 5-6

6. HUMAN INTERFACE DESIGN page 7-9
   - 6.1 UI design
   - 6.2 UX design

1. INTRODUCTION

1.1 Purpose

This software design document describes the architecture and system design of the Mobile Cooking Companion. The purpose of this software design document is to illustrate the features, and many uses of the Mobile Cooking Companion, and to make sure that the features meet user requirements, and all the intricacies of the application are discussed and understood.

1.2 Scope

The Mobile Cooking Companion is an application for people who have all distinct kinds of food and drinks in their home but do not know what to make from all of it. The app pulls information from a whole database of recipes and will take different food and drink inputs from the user to decide which recipe they receive.

Using this app, a user will receive benefits like creating new meals they never think of with the food they regularly buy at the supermarket. This can improve the quality of the food they eat by limiting their need to go out and get fast food. It can also help people who struggle with getting a meal plan going.

The goal of this app is to help people who have excess food and drinks in their home but never know what new creations they could make. This helps save money by budgeting and get the most out of each meal without all the waste.

1.3 Overview

The Mobile Cooking Companion is going to help a ton of people around the world who struggle with meal prep, and eating healthy food, and it will help reduce the amount of people who get fast food for convenience and reduce overall food waste.

Other businesses can use this application to create new entrées quickly and efficiently by fitting this application into the new computerized technology of the average restaurants.

Average users who need help with meal prepping, people who are into fitness and counting their macros, and restaurants who are looking to create new recipes to sell to their customers will all benefit using the Mobile Cooking Companion.

1.4 Reference Material

Documents referenced within this SDD document:

- Project Plan, submitted October 3rd, 2021,
    - PDF found at: https://tinyurl.com/MCC-Project-Docs

1.5 Definitions and Acronyms

MCC: Acronym for “Mobile Cooking Companion”.
Pantry: A list of cooking ingredients that the user has on hand.
Recipes: A list of cooking recipes that the user has input.

2. SYSTEM OVERVIEW

The environment of the app is going to be an Android phone. The application will be compatible with Android 9 Pie. You will be able to find it on through a local downloadable file (potentially the Google Play Store). The functionality of the app is that the app is going to generate recipes from a database when a user inputs an X number of items. For example, the user inputs: Ground beef, sour cream, cheese, and the app generates recipes for unique styles of burritos. The design is going to look like an interface with categories of food and drinks like "dairy, vegetables, fruit" and a search feature, like what one would see at a Walmart self-checkout station.

The user will choose all the items they have in their pantry and find or search for them in the app. Then after an X number of items is entered (the minimum amount for a recipe in the database) there will be a generate button to generate recipes that contain the items that were input by the user.

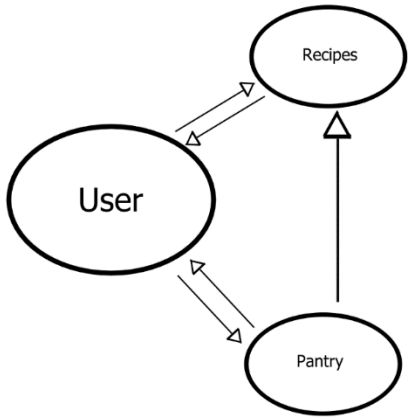
3. SYSTEM ARCHITECTURE

3.1 Architectural Design

The high-level aspects of our app revolve around three main things: The person interfacing with the application, the User; the recipes that the User inputs, Recipes; the ingredients the User has on hand, Pantry. The relationships between each aspect are shown by the arrows.

The User's relationship with the Recipes aspect is two-fold in that the User is the one who inputs new recipes into the system/database and the User is the one that requests a specific recipe from said database. Likewise, The User's relationship with the pantry is inputting ingredients into the database while being able to search through the ingredients that they have in the database.

The Recipe's relationship with the User has already been explained, but the relationship with the Pantry database is that of requesting ingredients that are in the Pantry database to accurately show what recipes are currently available for the User to create.



3.2 Technical design

The Recipes will be written as objects that will be stored in a SQL database for ease of accessing different parts throughout the program. Pantry items are stored as strings that are sanitized to account for user input and will also be stored in a SQL database.

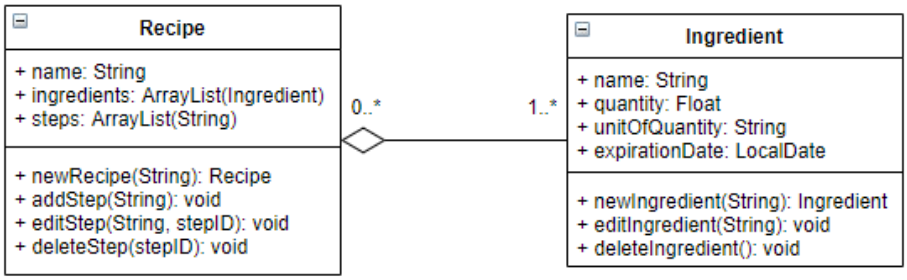
3.3 Design Rationale

We went with the architecture that we selected due to time and scope constraints. We found that the simplicity of using objects and basic data types would allow us to focus more on the structure and production of the product instead of having to learn a new language or new techniques that might suit the application better. Also, using simpler techniques would allow for easier restructuring down the road if we were to come back to the project or found that

one aspect is impossible with our timeframe and would need to scrap it entirely and slot something else in its place.

4. DETAILED DESIGN

4.1 Class diagrams



5. DATA (DATABASE) DESIGN

Using an SQL database management system, the information provided by the user will be broken down and stored in the relations listed below. Note that Recipes are broken into 3 relations, where the ingredients & instructions have their own table, and the recipe name is stored with a corresponding ID. For the Recipe_Instructions relation, each entry is an individual step of the instructions for a recipe.

PANTRY

Attribute	Key	Type	Required	Notes
ingredientName	Primary	Char(50)	Yes	
quantity		Float	Yes	
unitOfQuantity		Char(15)	Yes	
expirationDate		Date	No	

RECIPES

Attribute	Key	Type	Required	Notes
recipeID	Primary	Int	Yes	
recipeName		Char(50)	Yes	
numberOfSteps		Int	No	

RECIPE_INSTRUCTIONS

Attribute	Key	Type	Required	Notes
stepID	Primary	Int	Yes	Composite Primary Key
recipeID	Primary Foreign	Int	Yes	Composite Primary key
instruction		String	Yes	

RECIPE_INGREDIENTS

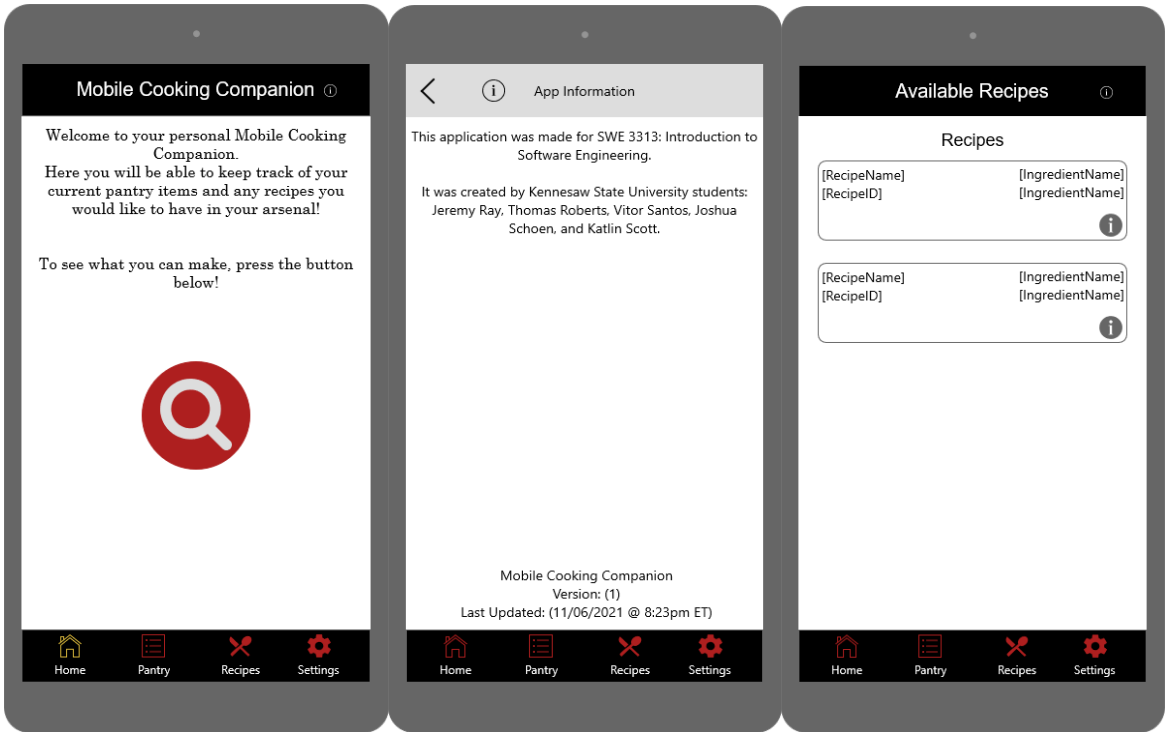
Attribute	Key	Type	Required	Notes
ingredientName	Primary	Char(50)	Yes	Composite Primary key
recipeID	Primary Foreign	Int	Yes	Composite Primary key
quantity		Float	Yes	
unitOfQuantity		Char(15)	Yes	

The information stored in these tables will be accessed and managed through SQL queries.

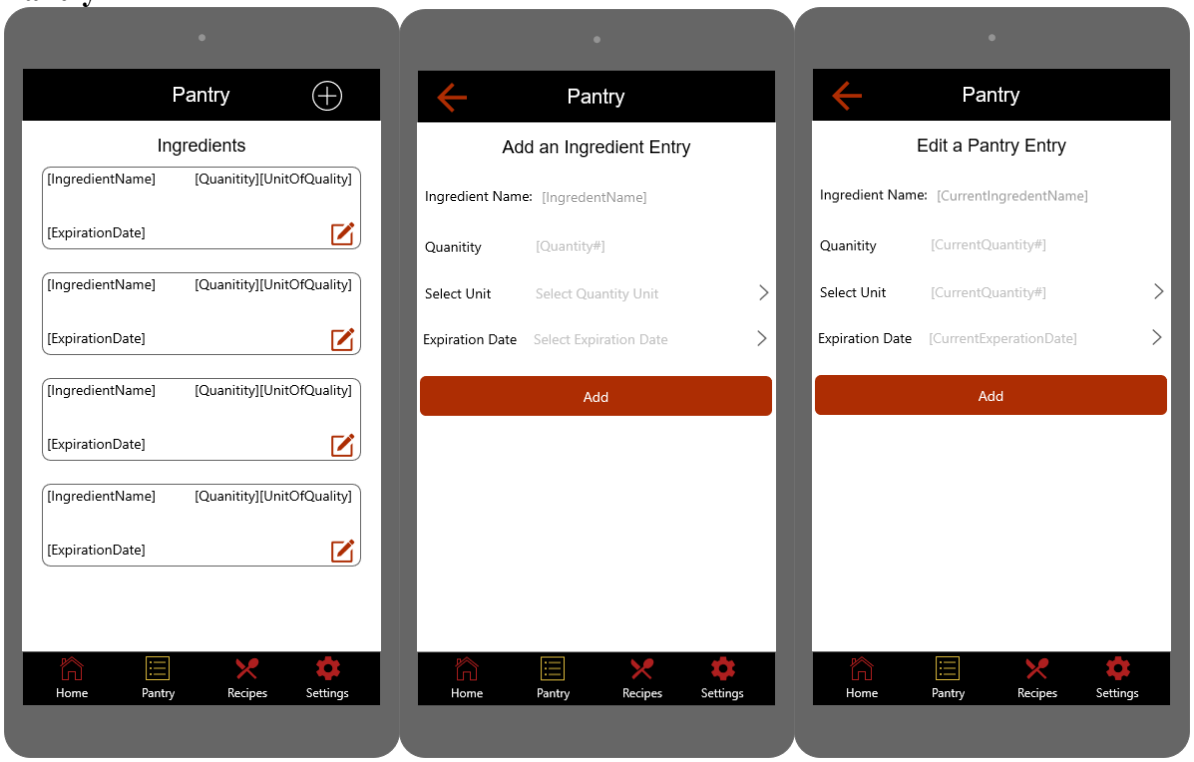
6. Human Interface Design

6.1 UI design

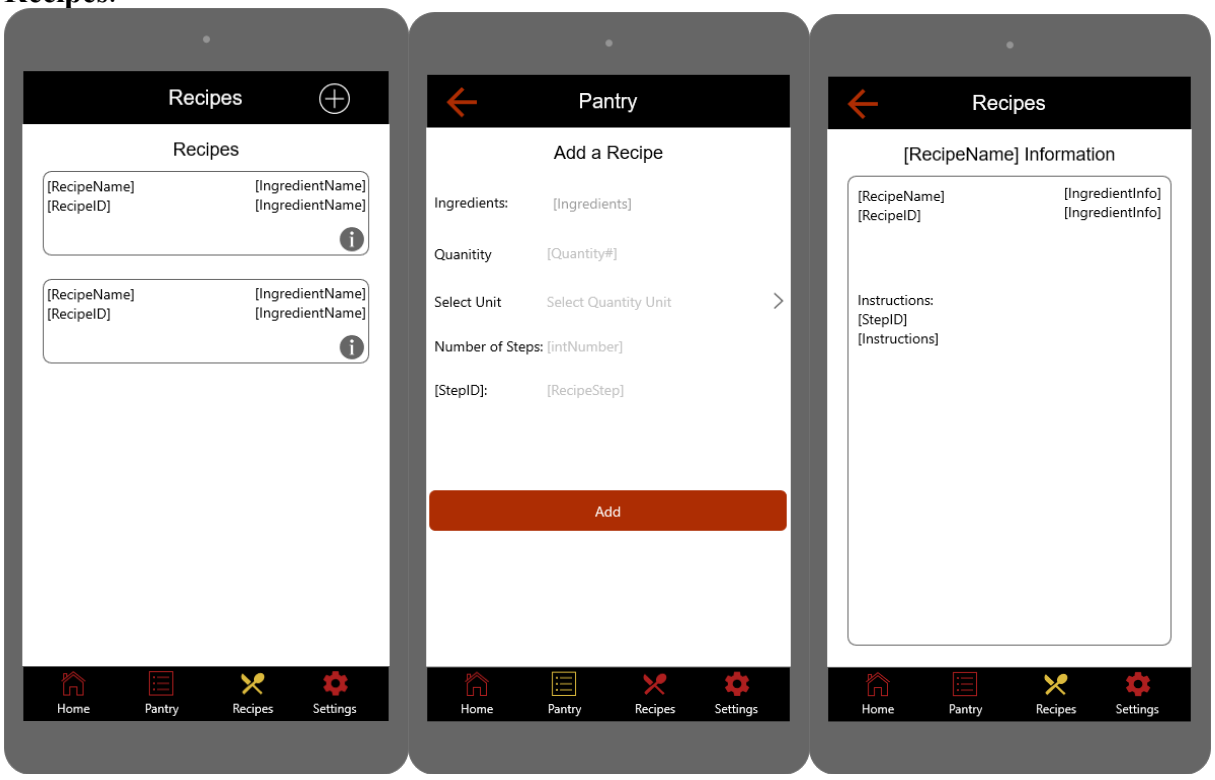
Home:



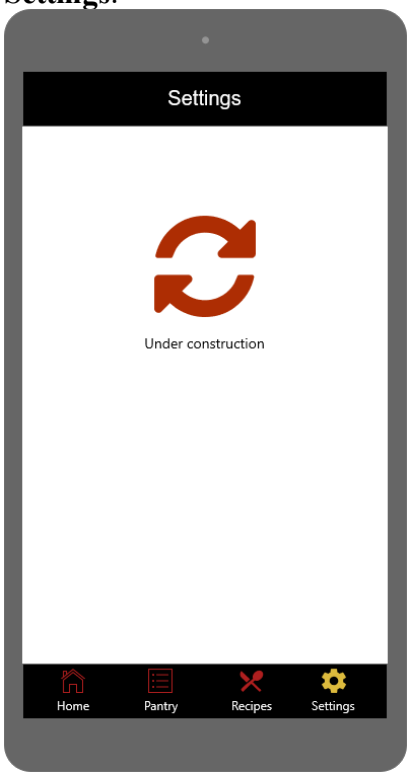
Pantry:



Recipes:

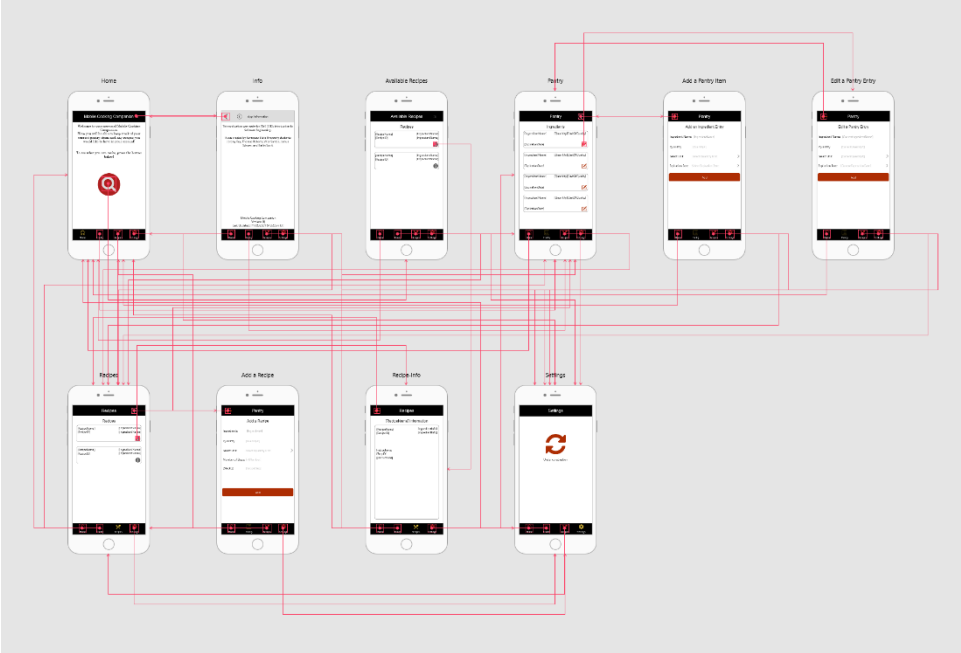


Settings:



6.2 UX design

User Experience Design Diagram:



Linked below is the webpage where one can view and interact the UI and UX Design emulation for the Mobile Cooking Companion app:
http://run.mockplus.com/oWJ2YaTSf32Bnonf1rpD/index.html