Coding Project Final Report

Smart Grocery

Prepared by
Bilal Naseer
Umar Khan
Daniyal Khan

***for use in* CS 440**
at the
University of Illinois Chicago

November 2023

REMOVE OR REPLACE ALL TEXT IN RED ITALICS BEFORE SUBMITTING REPORT

How to Use This Document

[1], [2][3]

Table of Contents

REMOVE OR REPLACE ALL TEXT IN RED ITALICS BEFORE SUBMITTING REPORT 1
How to Use This Document 1
List of Figures 4
List of Tables 5

I Project Description 6

1 Project Overview 6

2 Project Domain 6

3 Relationship to Other Documents 6

4 Naming Conventions and Definitions 6

4a Definitions of Key Terms 6
4b UML and Other Notation Used in This Document 7
4c Data Dictionary for Any Included Models 8

II Project Deliverables 9

5 First Release 10

6 Second Release 10

7 Comparison with Original Project Design Document 11

III Testing 12

8 Items to be Tested 12

9 Test Specifications 12

10 Test Results 12

11 Regression Testing 13

IV Inspection 14

12 Items to be Inspected 14

13 Inspection Procedures 15

14 Inspection Results 16

V Reccomendations and Conclusions 17

VI Project Issues 17

15 Open Issues 17
16 Waiting Room 17
17 Ideas for Solutions 18
18 Project Retrospective 18
VII Glossary 18
VIII References / Bibliography 18
IX Index 19

List of Figures

Figure 1 – Sequence Diagram for viewing a recipe

Figure 2 – Sequence Diagram for adding item to pantry

Figure 3 – Sequence Diagram for buying from the store

Figure 4 – UML diagram for release 1

Figure 5 – UML diagram for release 2

Figure 2 - Sample Use Case Diagram from Bruegge & DuToit (modified)**Error! Bookmark not defined.**

Figure 3 - Sample Use Case Diagram from Robertson and Robertson**Error! Bookmark not defined.**

List of Tables

Table 2 - Requirements - Acceptance Tests Correspondence.....**Error! Bookmark not defined.**

I Project Description

1 Project Overview

Smart grocery is an application in which a user can find recipes, as there are about thousands of recipes in the application. Per recipe, there are set ingredients; popularly known as a cookbook. The user will also have a virtual pantry, in which they can keep track of what ingredients they have / are missing. If such ingredients are missing, the user will be linked to a supermarket with the ingredients available and can purchase it. This is a great platform when it comes to meal prepping.

2 Project Domain

Essentially, the functionality of this project is being tested and evaluated to create a pleasant experience for the user. The user will be able to scroll through recipes, favorite a recipe, update the ingredients in their virtual pantry, and be given recommendations of recipes based on the virtual pantry. If an ingredient is missing, the user will be linked to a supermarket of their choice, in which they can purchase the missing ingredient(s)

3 Relationship to Other Documents

There is a relationship between this document and the Smart Grocery created by the group 25 in Fall 22. (Folga, A., Nadkarni, A., Nicdao, G., & Patel, M. (2022, October). Smart grocery project report. University of Illinois Chicago.)

4 Naming Conventions and Definitions

4a Definitions of Key Terms

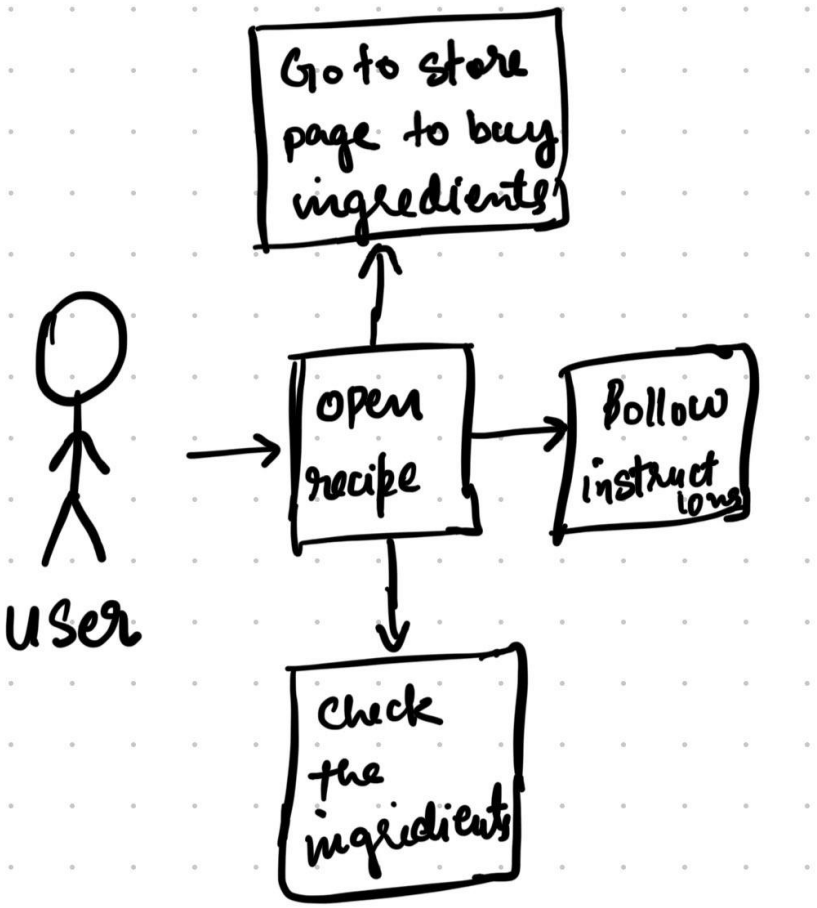
Pantry: A virtual list of items from your pantry

4b UML and Other Notation Used in This Document

[4]

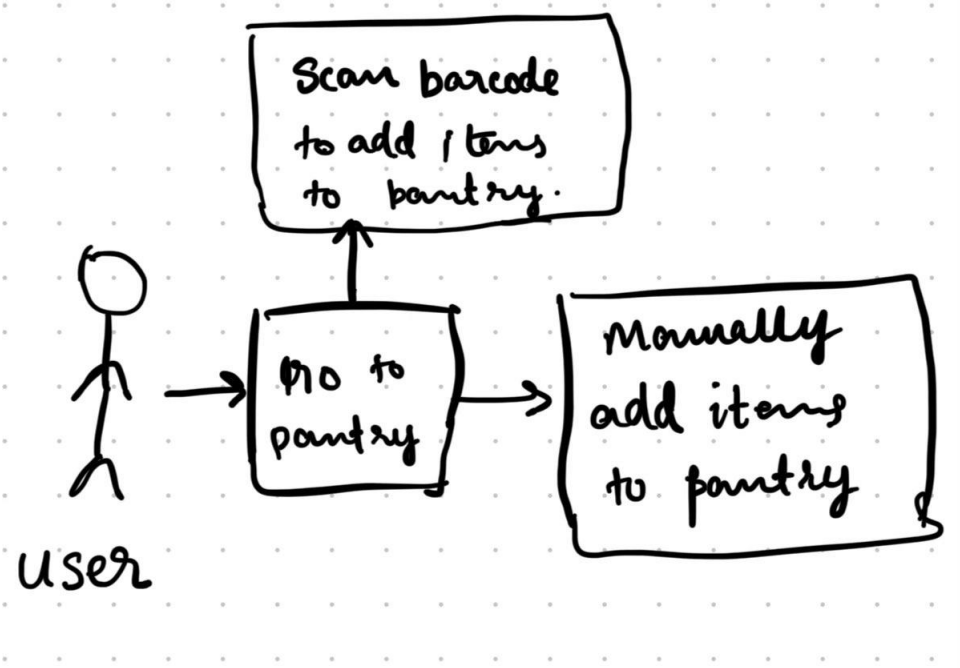
This document follows the Version 2.0 UML standard, as described by Fowler in [4].

Sequence diagram for viewing a recipe. Figure 1



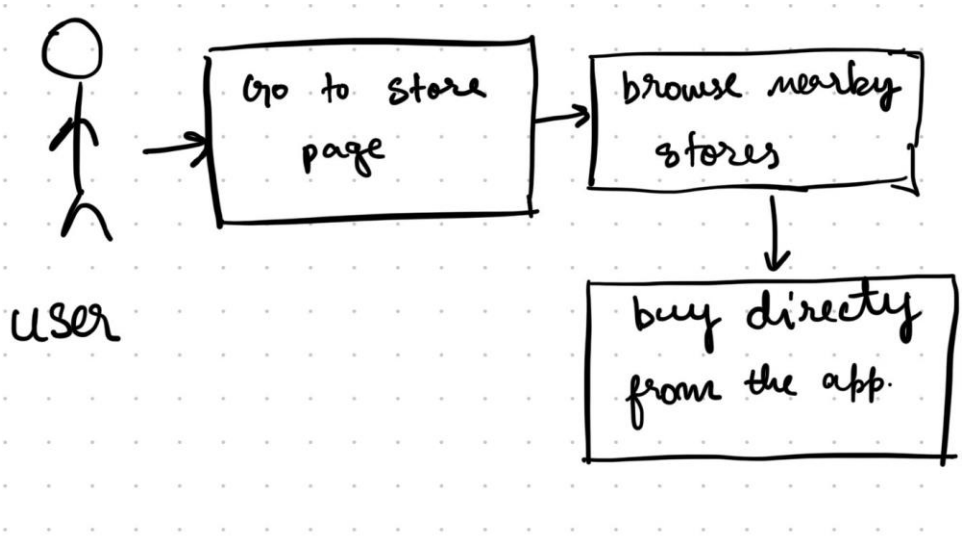
Sequence diagram for adding item to pantry.

Figure 2



Sequence diagram for buying from the store.

Figure 3



4c Data Dictionary for Any Included Models

1. **User Profile**
   - Attributes: User ID (unique alphanumeric string), Username, password).
   - Constraints: User ID must be unique; Email format validation.

2. Product Listing

- Attributes: ingredient ID, Name, cost, Availability, on sale
- Constraints: ingredient ID must be unique; cost in numerical format.

3. Recipe Details

- Attributes: Recipe ID, Title, Ingredient list (with quantities), Instructions, image
- Constraints: Recipe ID must be unique; Ingredients list linked to ingredient IDs where applicable.

4. Store

- Attributes: Store ID, Product Inventory (linked to Product IDs with quantities), Pricing.
- Constraints: Store ID unique; Inventory quantities in numerical format.

II Project Deliverables

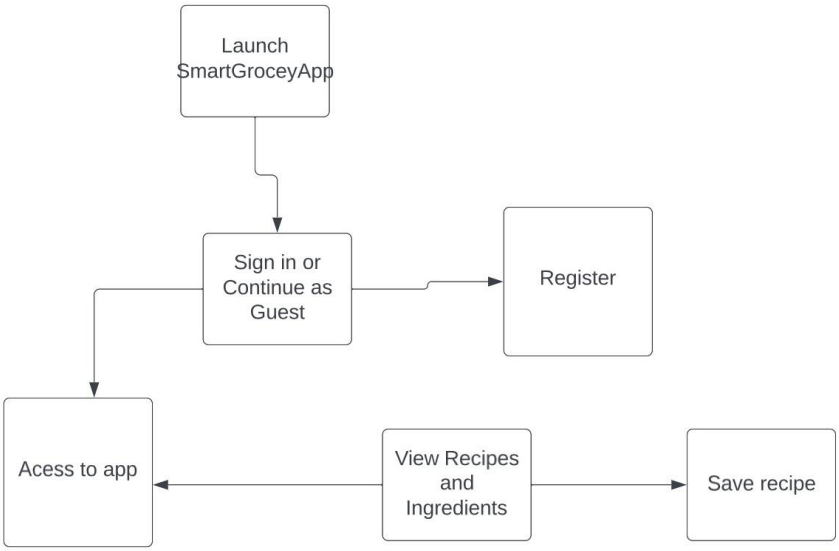
Smart Grocery is an app offering a vast collection of about a thousand recipes, functioning much like a digital cookbook. Each recipe comes with a specific set of ingredients. The app features a virtual pantry where users can monitor the ingredients they have or need. In cases where ingredients are missing, the app conveniently links users to the nearest supermarket where these items are available, and can purchase from the store within the app. This platform is particularly useful for planning and preparing meals efficiently. This application was made using dart, and SQLite for the databases.

5 First Release

Demo: 10-6-23

In this first scenario, we had created a simple functionality of being able to login or access the application as a guest and be able to view recipes and see the ingredients per recipe.

Figure 4



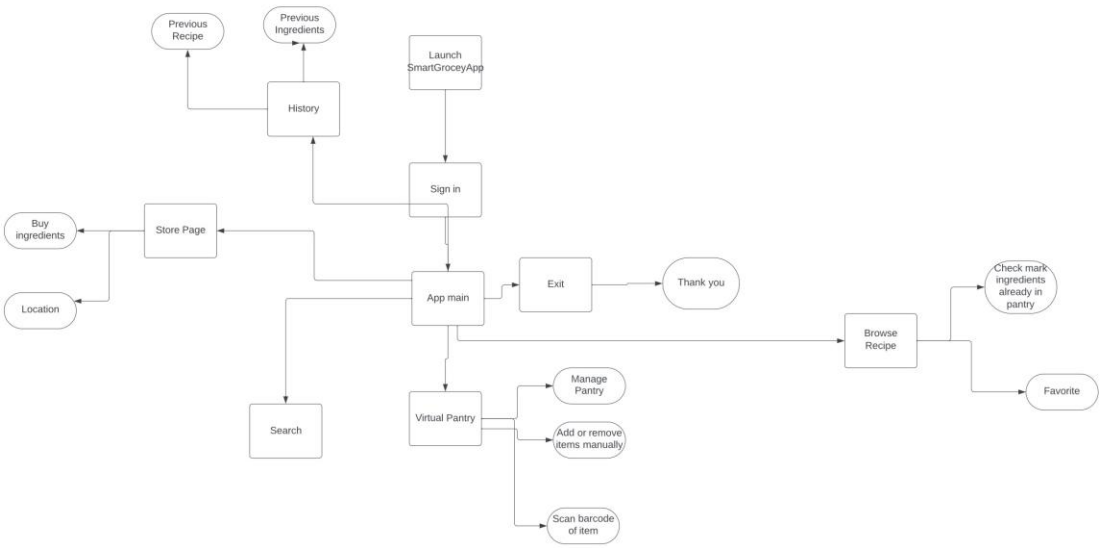
1 Second Release

Demo: November 4th

In this 2nd release, this application also consisted of a virtual pantry, where the user can add or remove items from the pantry (based on what they have in real life), and the

application will recommend recipes as well. Another feature added was the ability to display supermarkets and their pertaining addresses in Chicago, in case ingredients were missing.

Figure 5



6 Comparison with Original Project Design Document

There are tons of similarities and differences between the protype, and the project described by group25 of Fall 2022. The similarities consist of browsing through

recipes, see in the ingredients and instructions per recipe, and being linked to grocery stores to purchase ingredients that are missing.

The differences are as follows: The previous group had the idea of turning this app into a platform as well, where chefs can have their own account, which will allow them to post recipes and their cooking tutorials and interact with users. When the user creates an account, they would also be able to add dietary restrictions in their profile.

III Testing

7 Items to be Tested

User interface and actions

- clicking on recipe
- checking ingredients
- favoriting recipes
- Pantry tab & updating virtual pantry
- purchasing missing ingredients
- Store page
- Camera- to scan barcode of item to add to pantry

8 Test Specifications

9 Test Results

ID#1 - User Interface Test

- **Description**: Testing the ease of navigation and clarity of instructions within the app.
- **Items covered by this test**: Application interface elements.
- **Requirements addressed by this test**: Usability requirements.
- **Environmental needs**: Test environment mirroring the app's production environment.
- **Intercase Dependencies**: NA
- **Test Procedures**: Manual testing by navigating through all app screens.
- **Input Specification**: User actions (clicking on recipe, checking ingredients, favoriting recipes, updating virtual pantry, and purchasing missing ingredients
- **Output Specifications**: Application responses to user inputs.
- **Pass/Fail Criteria**: Interface should be intuitive and responsive without bugs.

ID#2 - Database Integration Test

- **Description**: Ensuring seamless interaction between the app and the database.
- **Items covered by this test**: Database queries, data retrieval, and update operations.
- **Requirements addressed by this test**: Database functionality requirements.
- **Environmental needs**: Access to the development database.
- **Intercase Dependencies**: Depends on successful user interface testing.
- **Test Procedures**: Automated scripts to test database operations.
- **Input Specification**: Sample database queries.
- **Output Specifications**: Expected database responses.
- **Pass/Fail Criteria**: Accurate and timely database operations.

Test Results

ID#1 - User Interface Test

- **Date(s) of Execution**: 10/4/23, 11/2/23, 11/19/23
- **Staff conducting tests**: Bilal, Umar, and Daniyal
- **Expected Results**: Responsive and intuitive interface.
- **Actual Results**: The application is functional
- **Test Status**: [Pass]

ID#2 - Database Integration Test

- **Date(s) of Execution**: 10/3/23, 11/1/23, 11/18/23
- **Staff conducting tests**: Bilal, Umar, and Daniyal
- **Expected Results**: Accurate and efficient database interactions.
- **Actual Results**: The database interaction is efficient

- **Test Status**: [Pass]

4b Regression Testing

User Interface Update Regression Test

- **Re-test Date(s)**: [November 4 2023]
- **Changes Made**: Updated the user interface to enhance responsiveness on mobile devices.
- **New Test Results**: The interface showed improved performance on various mobile devices

Database Query Optimization Regression Test

- **Re-test Date(s)**: [November 3rd, 2023]
- **Changes Made**: Optimized database queries for faster data retrieval.
- **New Test Results**: Queries executed 30% faster on average with no loss in data accuracy or integrity.

Payment Gateway Security Enhancement Regression Test

- **Re-test Date(s)**: [November 19 2023]
- **Changes Made**: Implemented additional security protocols for the payment gateway.
- **New Test Results**: Payment transactions remained secure with no significant impact on transaction speed or user experience.

IV Inspection

10 Items to be Inspected

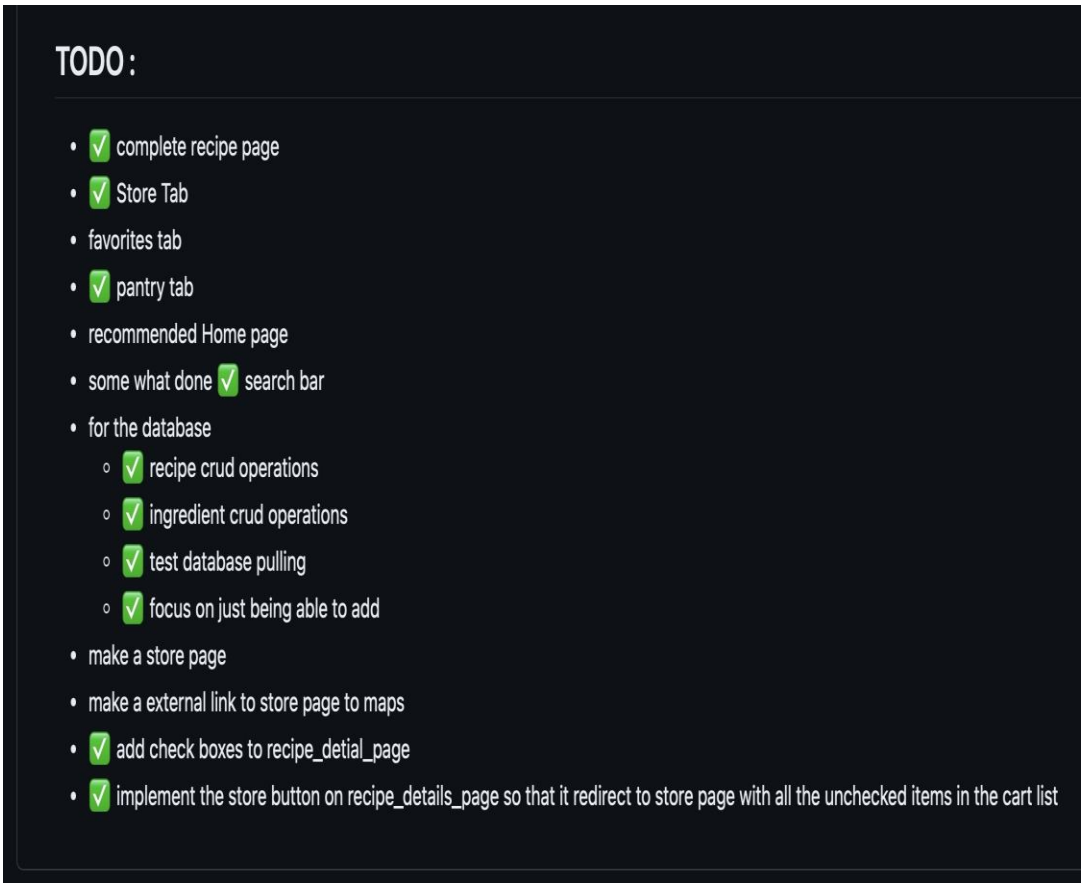
User interface and actions

- clicking on recipe
- checking ingredients
- favoriting recipes,
- Pantry tab & updating virtual pantry
- purchasing missing ingredients
- Store page
- Camera

Database

- *Database pulling*
- *Crud operations for recipe and ingredients*

11 Inspection Procedures



12 Inspection Results

Bilal and Umar inspected the user interface and actions on 3 different occasions: 10/4/23, 11/2/23 and 11/19/23 at 4:00 pm.

Flaw that occurred: Camera was not working in the first release

Daniyal inspected the database on 3 different occasions: 10/3/23, 11/1/23 & 11/17/23 at 2:00 pm

No major flaws had occurred

V Recommendations and Conclusions

All items that were being inspected passed the inspection testing

VI Project Issues

13 Open Issues

Need to implement the store Api for easier checkout which will the user experience more streamlined and make the user more engaging.

14 Waiting Room

Since this is a prototype, we did not dive deep into implementing the machine learning algorithm which would be used to provide recommendations based on the user's ingredients in the pantry, and previous recipes the users have utilized in the app. As a group, we wanted to focus on other important functionalities in the application. Applying the machine learning algorithm would be quite extensive and would take some time, but it is not impossible. Another feature we may add later is for users is being able to add dietary restrictions: Allergies, vegetarian, halal, and kosher. Based on these restrictions the user chooses, it will filter out recipes. The recommendations feature would take this into consideration as well.

15 Ideas for Solutions

A nice feature for this project would be to add the ability to share recipes, and create recipes as a chef, just as the original creators of this project had in mind. If that was to be done, this would be a very versatile and complete app, like popular apps like TASTY.

16 Project Retrospective

Your text goes here . . .

For the most part, this project went well, and we are happy with what we have come up with. We may however implement more features in the future, along with the machine learning algorithm due to the great potential of this app

VII Glossary

VIII References / Bibliography

Folga, A., Nadkarni, A., Nicdao, G., & Patel, M. (2022, October). Smart grocery project report. University of Illinois Chicago.

[1] Robertson and Robertson, Mastering the Requirements Process.

[2] A. Silberschatz, P. B. Galvin and G. Gagne, Operating System Concepts, Ninth ed., Wiley, 2013.

[3] J. Bell, "Underwater Archaeological Survey Report Template: A Sample Document for Generating Consistent Professional Reports," Underwater Archaeological Society of Chicago, Chicago, 2012.

[4] M. Fowler, UML Distilled, Third Edition, Boston: Pearson Education, 2004.

IX Index