

Do's 8 - Deliverable 3 - Report

Requirements

Scope of the project had to shift slightly for this phase - something that was not *explicitly* stated in the requirements document was that all of these components we mention have UIs and those UIs need to be built first and then “hooked up” to the actual data manipulation code. So a lot of the effort on this phase was getting the UIs built and the navigation between the different pages set up properly.

Some of the *actual* functionality was still included and where that is applicable, the requirement id from the SRS is recorded along with the requirement.

- Desktop Dashboard UI
 - Has view of recipe browser
 - Has view of meal plan
 - Has form for creating a new recipe
 - Has view of inventory
 - Has view of shopping list
- Desktop Meal Plan UI
 - Has view of public recipe browser
 - Has view of private recipe browser
 - Has view of meal plan
- Can select item in desktop recipe browser and it will populate in the meal plan view
- Can deselect item that is selected in the recipe browser and it will populate in the meal plan view
- Can deselect item from the meal plan browser - this will populate appropriately in the recipe browser
- Recipe views in dashboard and meal plan tab is linked
- Meal plan in dashboard and meal plan tab are linked
- Desktop Inventory :
 - Can add an item to the table.
 - Can delete an item from the table

→ Android Dashboard UI

- ◆ Instructions on how to use the application
- ◆ Button access for meal plan activity
- ◆ Button access for shopping list activity
- ◆ Button access for inventory activity
- ◆ Button access for recipe book activities

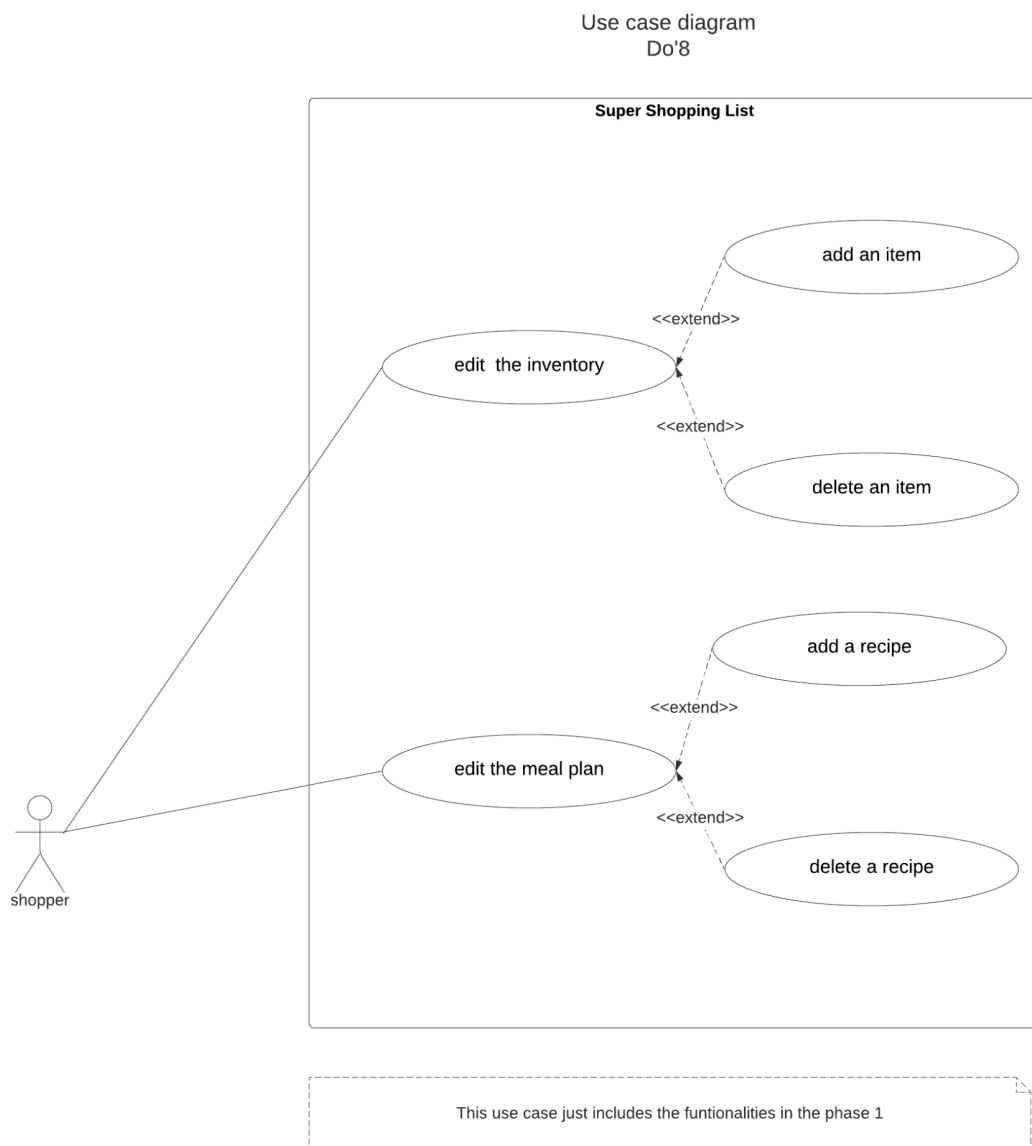
→ Android meal plan UI

- ◆ Displays sample data of how the meal plan activity would work once integrated with the database.

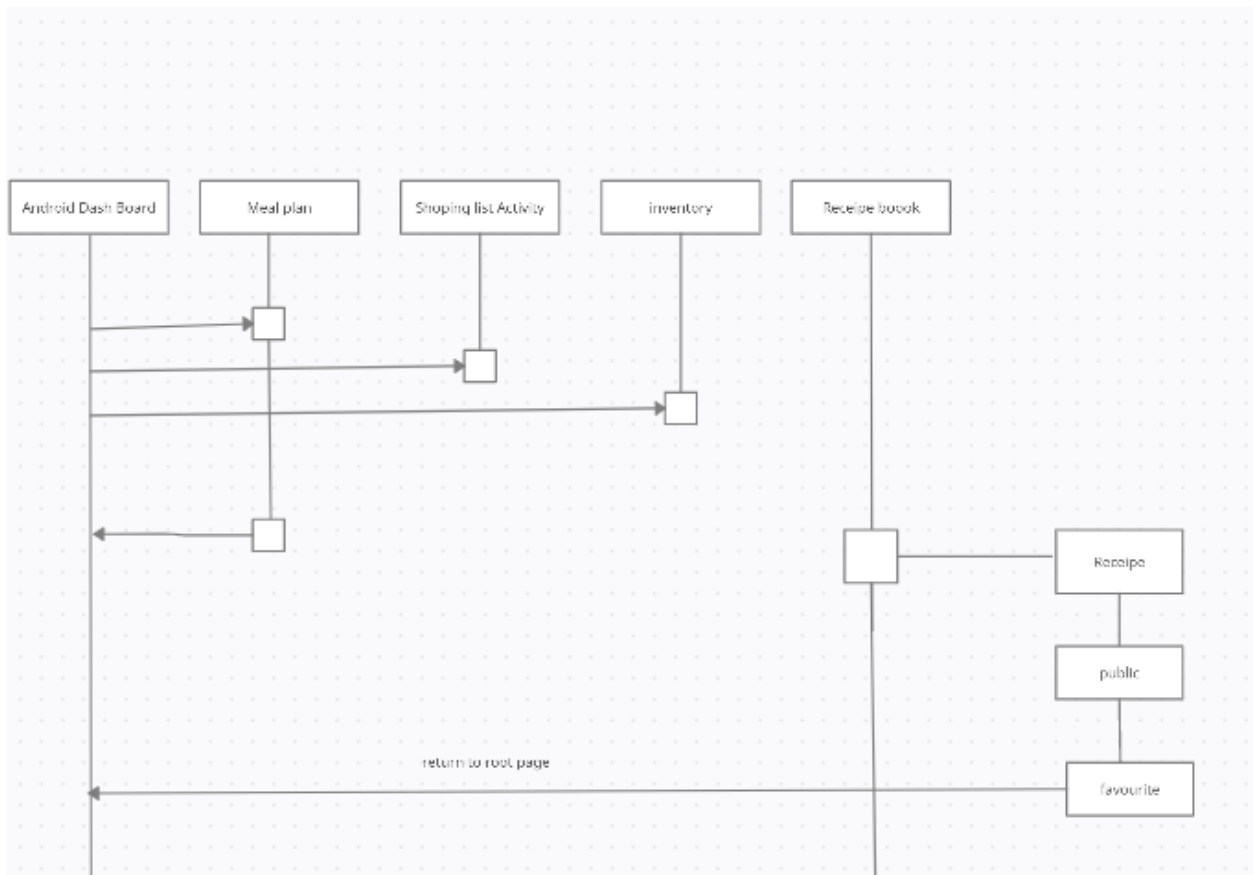
- Android shopping list activity
 - ◆ Displays sample ingredients for shopping list - actual coding to be integrated in phase 2.
- Inventory contains ingredients that are available in stock at home.
- Recipe books contain public recipes, favorite recipes and a root page.

UML Diagram

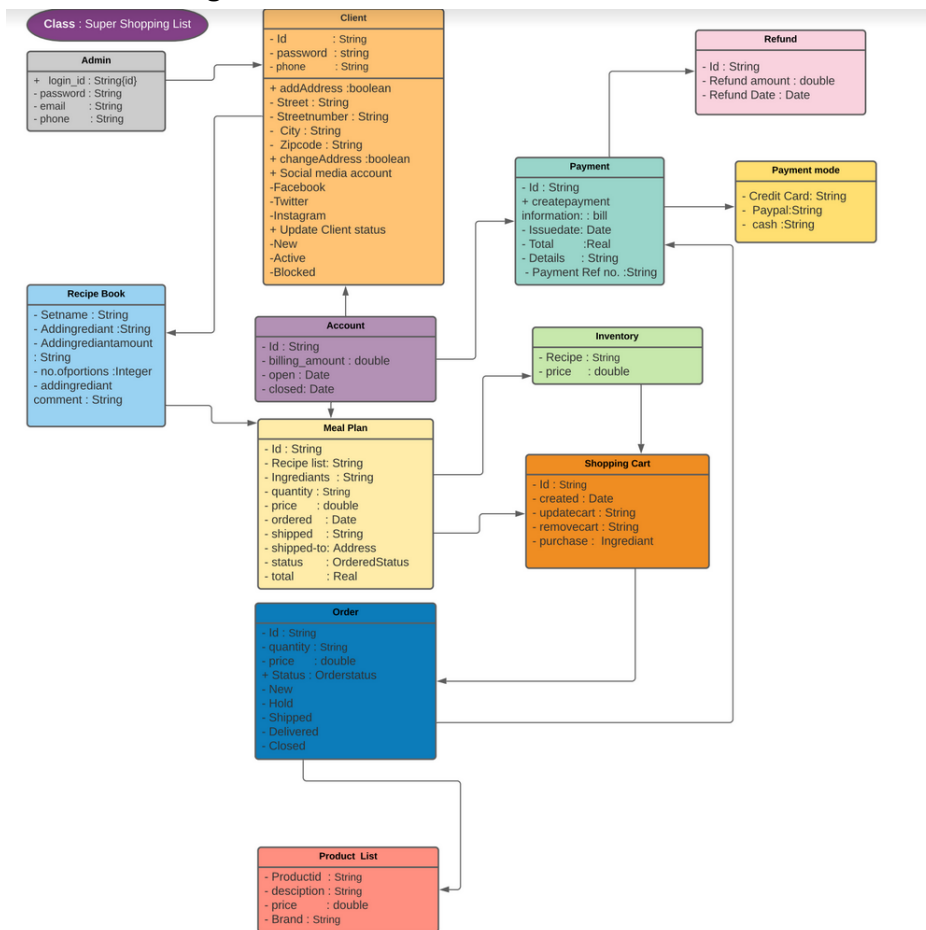
Use Case Diagram



Sequence Diagram (naveen)



UML Class Diagram



Tests

We will be using JUnits to test the app code. For Phase 1, only the U.I. pages have been created for the app therefore JUnits have not yet been created. However, the way the android app can be tested is to test the buttons in the main activity in order to determine if those buttons take you to the intended activity and that all the activities are included in the AndroidManifest.xml.

Manual

Mobile

Once you install the mobile application on an android device, the first page you see is the dashboard. From there you can read the instructions of how you can use the application and what functionalities the application provides. From the dashboard, you are able to navigate to

one of the six pages which include inventory, meal plan, recipe book root, public recipes, favorite recipes and the shopping list.

For phase one, there is no actual functionality in the app, for now you can access different activities and the layouts consist of data that is not pulled from the database (will be implemented in phase 2). The inventory activity helps the user to keep track of their inventory for different ingredients. In this activity, there are three sections being fridge, pantry and spice cabinet. Secondly, the meal plan activity consists of meals planned for different days for example, Cake for Tuesday and Rice and beans for Thursday. Thirdly, the recipe book root allows you to access the recipe book pages which are the public recipes and the favorite recipes. In the public recipes, you have access to all the different recipes uploaded by different users. And from there you can select recipes to be your favorite and they will appear in the favorite recipes activity. While the shopping list activity pulls ingredients from the recipes of the meals you have in your meal plan and suggests you to shop for them. The shopping list can be manually edited.

Desktop

The desktop app has a tab layout to navigate through the different views, these are:

- Dashboard
- Meal plan
- Shopping list
- Recipe
- Inventory
- User Profile

The dash board has a list of recipes that you can browse through, collapse with the arrows on the left, and select for your meal plan with the select box on the right. When you select a recipe, you will see it populate in the meal plan window in that same tab.

If you change tabs with the tab buttons at the top, you can change to the meal plan view. The meal plan view shows the same recipe browser and a meal plan browser that are editable in the same way. If you edit from here, you will see the changes populate in the other tab.

Instructions on How To Compile

Mobile

This is an Android app written with Java. To run it you will need to install the latest version of Android Studio, which can be found on developer.android.com. Once the android studio is opened, you have to open the project with the source code of this project. Make sure to separate the mobile app folder from the git repository so that android can run it. Once the app source code is opened in Android Studio, you can hit run. If all dependencies are up-to-date, an Android phone emulator will run the app. One thing to always make sure of is to have all activities stated in the AndroidManifest.xml. If not, the respective activity would not open once

directed to from within the emulator. If something fails, the most likely culprit is Gradle. Deleting the .gradle file and rebuilding the app may help. Also make sure that Gradle is up to date as well.

Desktop

This is a Qt application written with qt for python - to run it you will need to install pyside6 and qt creator.

Getting qt-creator will require an account (it's free).

- Go to qt.io
- Click download try
- Go open source
- Scroll to and click download the qt installer
- Run the installer
- Install qt 5.15
 - Include qtcore
 - Qt designer
 - You can exclude anything you already have (like visual studio or mingw)
 - You can disregard anything like mobile / android / web /etc (this is a qt app made with python - let that guide what you install to keep the install to keep it as small as possible)

Dependencies

- Python 3.8 or later
- Install (using pip) pyside6
- Ensure that your environment is set up correctly such that when you run a python command it is indeed the python for which pyside was installed
- If you are not using python3.9 you will need to pip install dataclasses (they are built into 3.9)

Running

- You can run the program by using `python mainwindow.py` or by opening the project in an ide and running / pointing config at mainwindow.py.
- Using the build.bat script - this will compile all the ui files into python code (these files are committed to the git so you shouldn't *need* to do this, but you can)
 - For linux systems, you can use a shell script (just remove that "@echo off")
- If you prefer to use an ide to run your code rather than the command line, you can have your ide of choice point its run configuration towards that build script.

Test

Desktop

- Test Meal Plan to Shopping List Converter

- Empty meal plan case
- One ingredient recipe
- Case 1 with same ingredient into two different recipes (failure - fix for phase 2)
- Case 2 with same ingredient into two different recipes (failure - fix for phase 2)
- Test adding recipes to ingredient tree
 - Empty tree
 - Tree with items

Feedback-based Details

Feedback summary and detailed discussion with Eagle 8 Team

- Can manager (admin) access users account
Yes, we have only admin and user accessible account but no separate interface for manager so, need to update the document accordingly
- Can admin delete users' account
Yes, admin has the privileges to delete any unauthorised user
- Does editing a recipe change public recipe
Yes, it will, the owner of the recipe has the privilege to edit the recipe and the changes will get reflected to the users.
- Public recipes don't change but users can save and edit their own copy - how public vs private recipes are saved
We have different fields (public and private) in the same table just we need access these pages as per our requirement
- How many users are authorized
As of now we have a group of 8 people authorised, but we can support more users.
- In the use case diagram what is the difference between a book of recipes and add recipes to book
Adding recipe is to just create a new recipe whereas book of recipes is the collection of recipes which is easily accessible to the users
- Is the Database in the cloud?
Yes, it's in the cloud, it's a browser-based system database that needs to be hosted on a separate server.

Reflection

Everything seems to be moving along well. Interfaces were the main focus for our Phase 1 as a foundation to add functionality to in the following phases.

On the desktop side, with how fast things are moving for the interfaces and linking of data between them, it might be that some of the features designated as “fancy details” that were likely to get cut might actually stay in.

For the mobile app, we were able to successfully create User Interfaces for all the activities. The main activity allows the users to direct themselves to different activities. The results were satisfactory considering we had some members not available to contribute to the project. Like mentioned above, the actual backend will be integrated in the next phase of this project.

Contributions

Names	Contribution Description	Contribution (%)	Notes
Mustafa Memon	Android App: UI for dashboard, meal plan. Java code for all activities (dashboard, meal plan, inventory, recipe root, shopping list & 2 recipe book pages), requirements, User Manual, android tests, reflection, code inspection file.	20%	
Brice Brosig	Qt crash course for those working on desktop app. Built the main window, recipe generator for testing, and supporting functions for sharing data with different views in different tabs. Built dashboard page. Built meal plan page. Built shopping list page. Contributed to the requirement sections, desktop manual, desktop running instructions, and the inspection code file.	20%	
Mica	Built private recipe book page. Built public recipe book page. Built create a recipe form. Contributed to requirements.	13%	
Nestor Molina	Created some of the Android layouts. Created and implemented the database in Azure. Created the necessary tables using SQL. Wrote the mobile 'How to Compile' section.	14%	
Tam Doan		14%	
Vandana Sinha	Completed the Android layout for one of the module. Connected database instances via aws to the	13%	

	local database server section. Document feedback details from Eagle 8's team.		
Nikhil Gaur		0%	Is currently out of country for a family emergency - he will contribute to future phases of the project.
Naveen	Public recipe and favorite recipe UI.	6%	