Shrey Agarwal

Grocery Application Report

1. Introduction

1.1 Overview

This is an android app that helps you to make a list of grocery items which you might buy, along with its price and quantity.

1.1 Purpose

The core purpose of the application is to keep track of the grocery items so that we do not forget. We sometimes forget the things that we want to buy. However, with the assistance of this app you can make a list of grocery items you intend to buy so that you don't forget anything and also have a track of your expenditure for budget maintenance. Also it will be easy to keep track of items.

## 2. Literature Survey

2.1 Existing Problem

Users frequently forget items to buy because of which they have to run to shops again and again which is quite a frustrating and tiring situation and if our expenses crosses out budget while shopping that could be a matter of concern.

2.2 Proposed Solution

      To overcome this problematic situation I built a grocery app which helps you to list down all the item that you need to buy along with its price.

3. Theoretical Analysis

3.1 Block Diagram
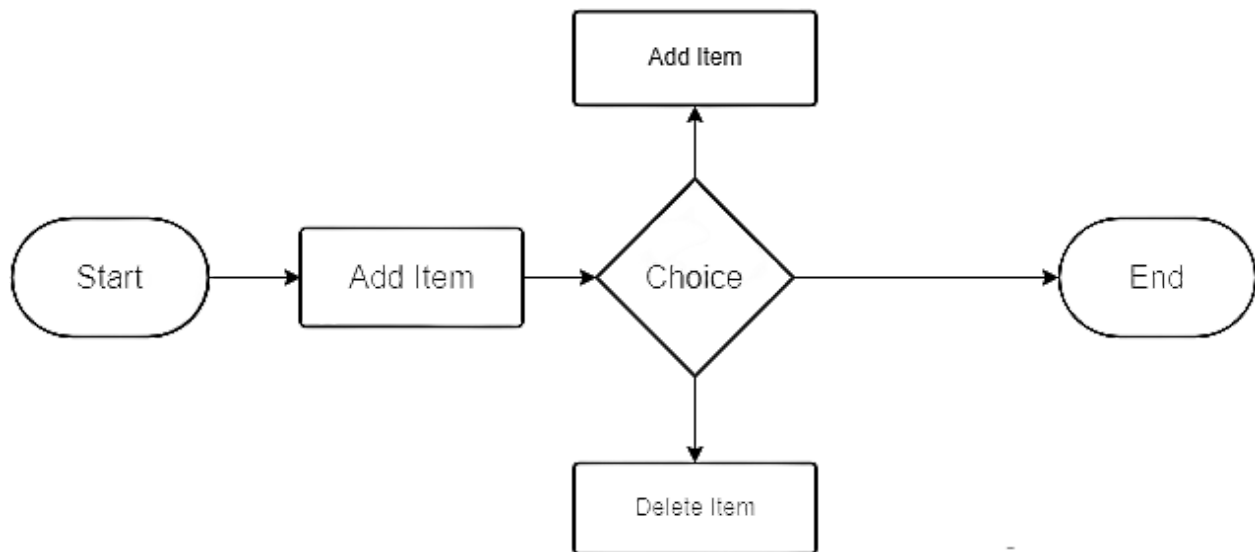
3.2 Hardware/Software designing

1. Android Studio
2. Kotlin
3. Windows 11

4. Experimental Investigations

      In this project MVVM (Model View ViewModel) was used for architectural patterns, Room for database, Coroutines and RecyclerView to display the list of items.

LiveData: A data holder class that can be observed. Always holds/caches the latest version of data, and notifies its observers when data has changed. LiveData is lifecycle aware. UI components just observe relevant data and don't stop or resume observation. LiveData automatically manages all of this since it's aware of the relevant lifecycle status changes while observing.

ViewModel: Acts as a communication center between the Repository (data) and the UI. The
UI no longer needs to worry about the origin of the data. ViewModel instances survive Activity/Fragment recreation.

Repository: A class that you create that is primarily used to manage multiple data sources.

Entity: Annotated class that describes a database table when working with Room.
Room database: Simplifies database work and serves as an access point to the underlying SQLite database (hides SQLiteOpenHelper). The Room database uses the DAO to issue queries to the
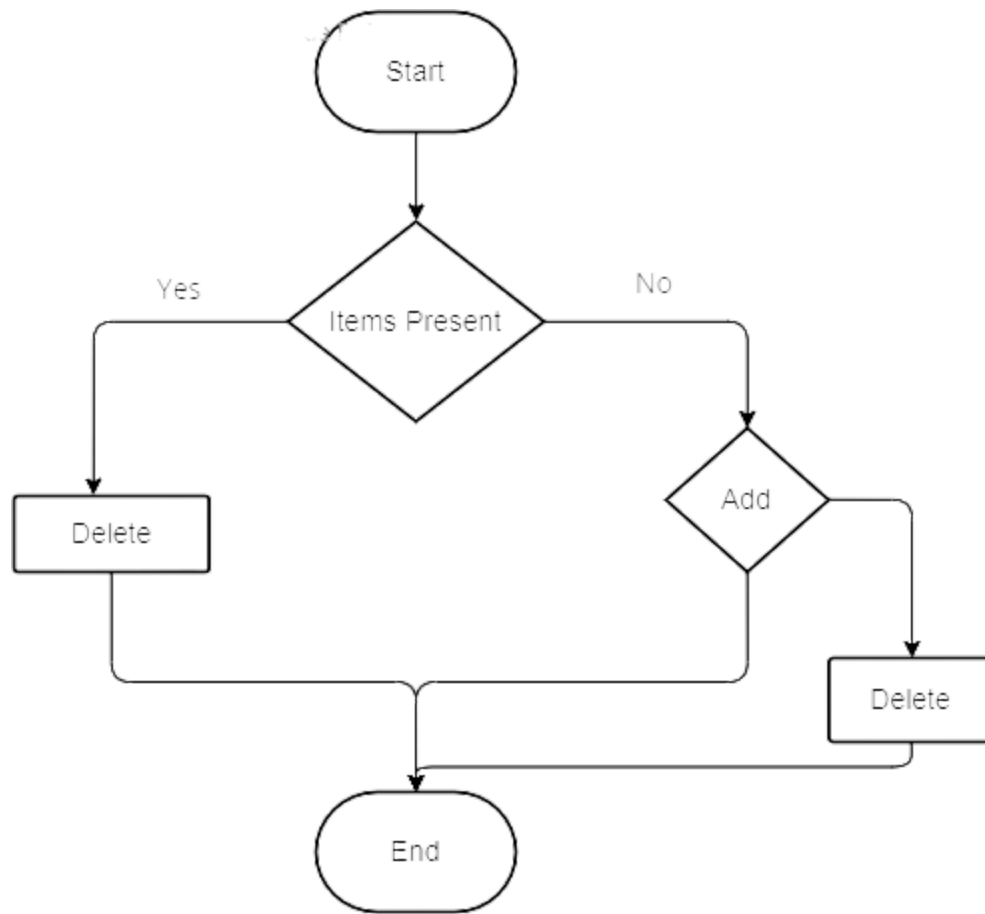SQLite database.

SQLite database: On device storage. The Room persistence library creates and maintains this database for you.

DAO: Data access object. A mapping of SQL queries to functions. When you use a DAO, you call the methods, and Room takes care of the rest.
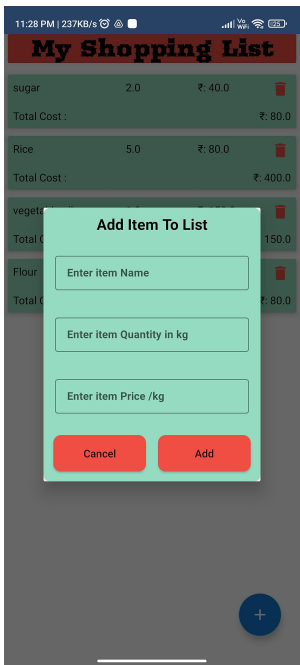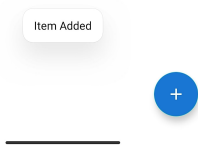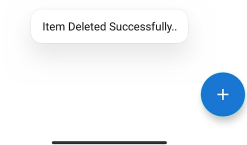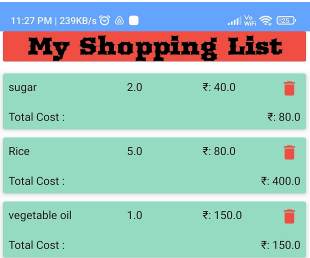
RecyclerView: It is a container and is used to display the collection of data in a large amount of dataset that can be scrolled very effectively by maintaining a limited number of views.

Coroutines: Coroutines are lightweight thread, we use a coroutine to perform an operation on other threads, by this our main thread doesn't block and our app doesn't crash.

5. Flowchart

6. Result

My Shopping List

sugar	2.0	₹: 40.0	🗑
Total Cost :			₹: 80.0
Rice	5.0	₹: 80.0	🗑
Total Cost :			₹: 400.0
vegetable oil	1.0	₹: 150.0	🗑
Total Cost :			₹: 150.0

Item Deleted Successfully..

+

My Shopping List

sugar	2.0	₹: 40.0	🗑
Total Cost :			₹: 80.0
Rice	5.0	₹: 80.0	🗑
Total Cost :			₹: 400.0
vegetable oil	1.0	₹: 150.0	🗑
Total Cost :			₹: 150.0
Flour	2.0	₹: 40.0	🗑
Total Cost :			₹: 80.0

Item Added

+

My Shopping List

sugar	2.0	₹: 40.0	🗑
Total Cost :			₹: 80.0
Rice	5.0	₹: 80.0	🗑
Total Cost :			₹: 400.0
vegeta			🗑
Total C			150.0
Flour			🗑
Total C			₹: 80.0

Add Item To List

Enter item Name

Enter item Quantity in kg

Enter item Price /kg

Cancel	Add

+

My Shopping List

Item Deleted Successfully..

+

7. ADVANTAGES & DISADVANTAGES

- **Advantages:**
    - ■ **Easy to use and understand.**
    - ■ **Requires minimum device specifications.**
- **Disadvantages:**
    - ■ **Lacks various features.**

8. APPLICATIONS

Although application is simple and lacks various functionalities it can be really useful in real life during shopping you won't be needing to memorize the items, you can simply type and add item to your grocery list.

9. Conclusion

The project was a great learning opportunity and to brush up the basics withe the android application developing. This project helped me to clear my concepts on Room Database, Coroutines, MVVM, etc. This project would help me not just as a developer to learn new and interesting things but also as a user we generally forgets items to purchase while shopping. Working on this project made me confident enough to apply my knowledge on android app development and create such an app. I have used Kotlin to build this application. All the functionality is coded in the classes and interfaces created and the layout is designed using xml.

10. Future Scope

The application has a lot of scope in the near future just we need to improve various things in the application and also there is need to add various features, for instance if we add the buying option or the map which shows the location of the shops which has those items.

11. Bibilography

1. Google: https://www.google.com/
2. Geeksforgeeks: https://www.geeksforgeeks.org/how-to-build-a-grocery-android app-usingmvvm-and-room-database/
3. Android Developer: Android Basics in Kotlin course | Android Developers
4. YouTube: https://www.youtube.com/watch?v=vdcLb_Y71Ic

Appendix

Source Code:

Due to a lot of code the link to the project repository is been given:

https://github.com/smartinternz02/SI-GuidedProject-56830-1660823790/tree/main/Grocery%20App(Final%20Project)