# Grocery App Report

*-Submitted by PARUL SAHNI (parulsahni3282@gmail.com)*
*-https://g.dev/parul0621(GDG Profile)*

1 INTRODUCTION
1.1 Overview
A brief description about your project
1.2 Purpose
The use of this project. What can be achieved using this.

2 LITERATURE SURVEY
2.1 Existing problem
Existing approaches or method to solve this problem.
2.2 Proposed solution
What is the method or solution suggested by you?

3 THEORITICAL ANALYSIS
3.1 Block diagram
Diagrammatic overview of the project.
3.2 Hardware / Software
designing Hardware and software requirements of the project .

4 EXPERIMENTAL INVESTIGATIONS
Analysis or the investigation made while working on the solution.

5 FLOWCHART
Diagram showing the control flow of the solution

6 RESULT
Final findings (Output) of the project along with screenshots.

7 CONCLUSION
Conclusion summarizing the entire work and findings.

8 FUTURE SCOPE :Enhancements that can be made in the future.

# 1. Introduction

## 1.1 Overview

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

## 1.2 Purpose

We are humans and we cannot remember everything. 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 do not forget anything and also have a track of your expenditure for budget maintenance.

# 2. Literature Survey

## 2.1 Existing Problem

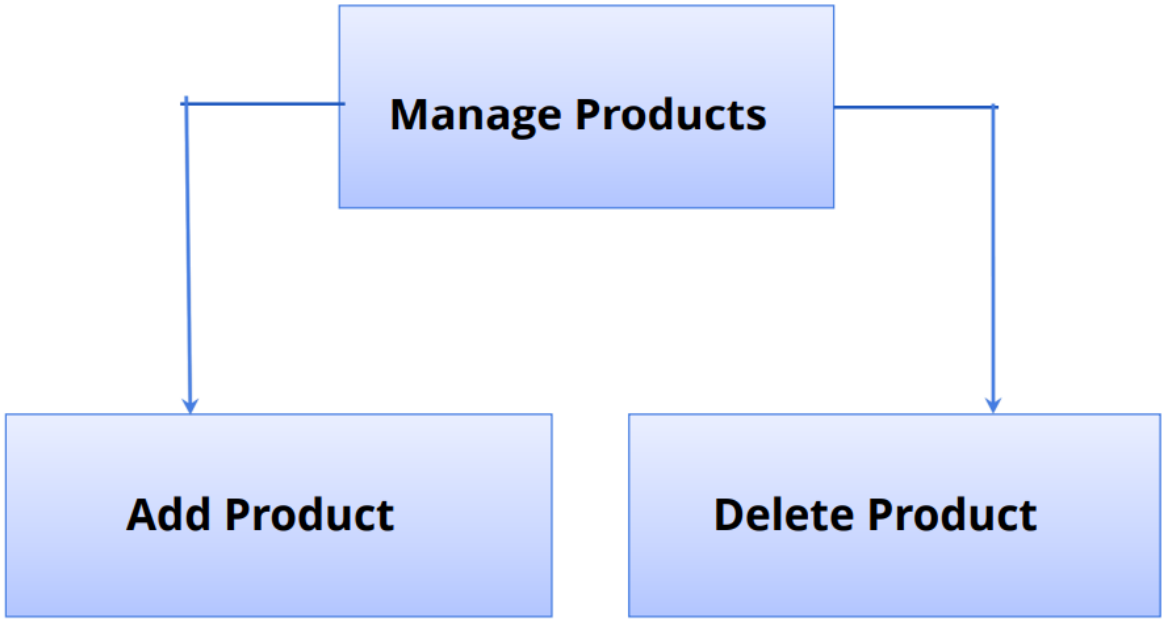
Users frequently forget items to buy because of which they have to run to shops again and again which is quite a frustrating and tiring situation and if our expenses crosses out budget while shopping that could be a matter of concern.

## 2.2 Proposed Solution

To overcome this problematic situation I built a grocery app. This 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

- Windows 10 OS
- Android Studio
- Min 4GB RAM


## 4. Experimental Investigations

In this project MVVM (Model View ViewModel) was used for architectural patterns, Room for database, Co-routines 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.

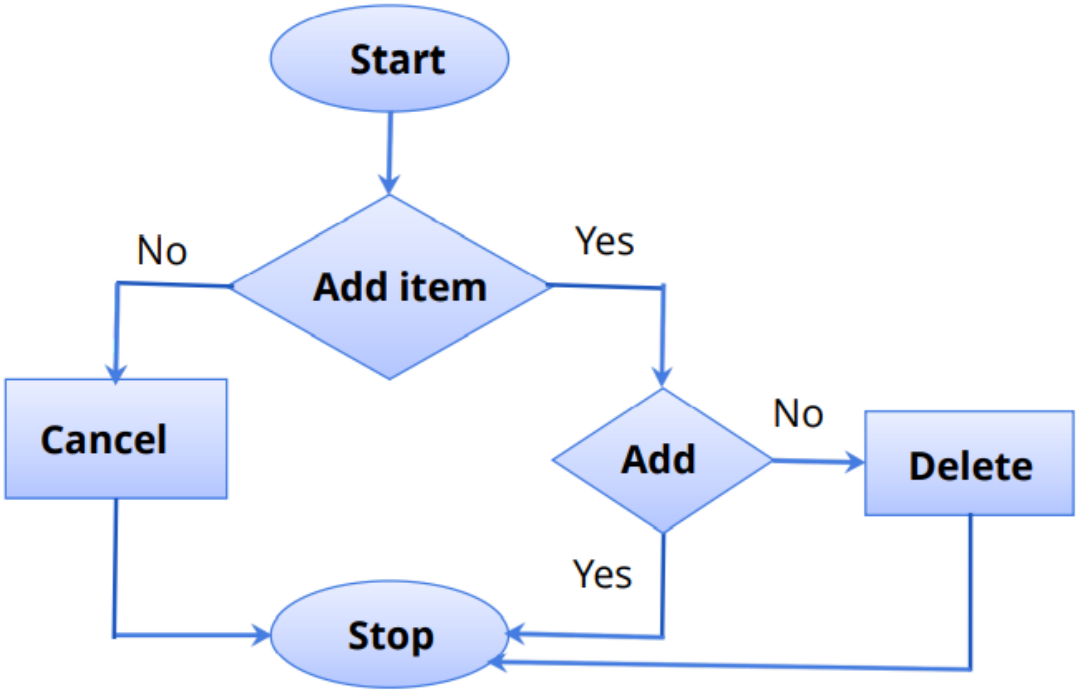
**DAO:** Data access object. A mapping of SQL queries to functions. When you use a DAO, you call the methods, and Room takes care of the rest.

**RecyclerView :** It is a container and is used to display the collection of data in a large amount of dataset and can be scrolled very effectively by maintaining a limited number of views.

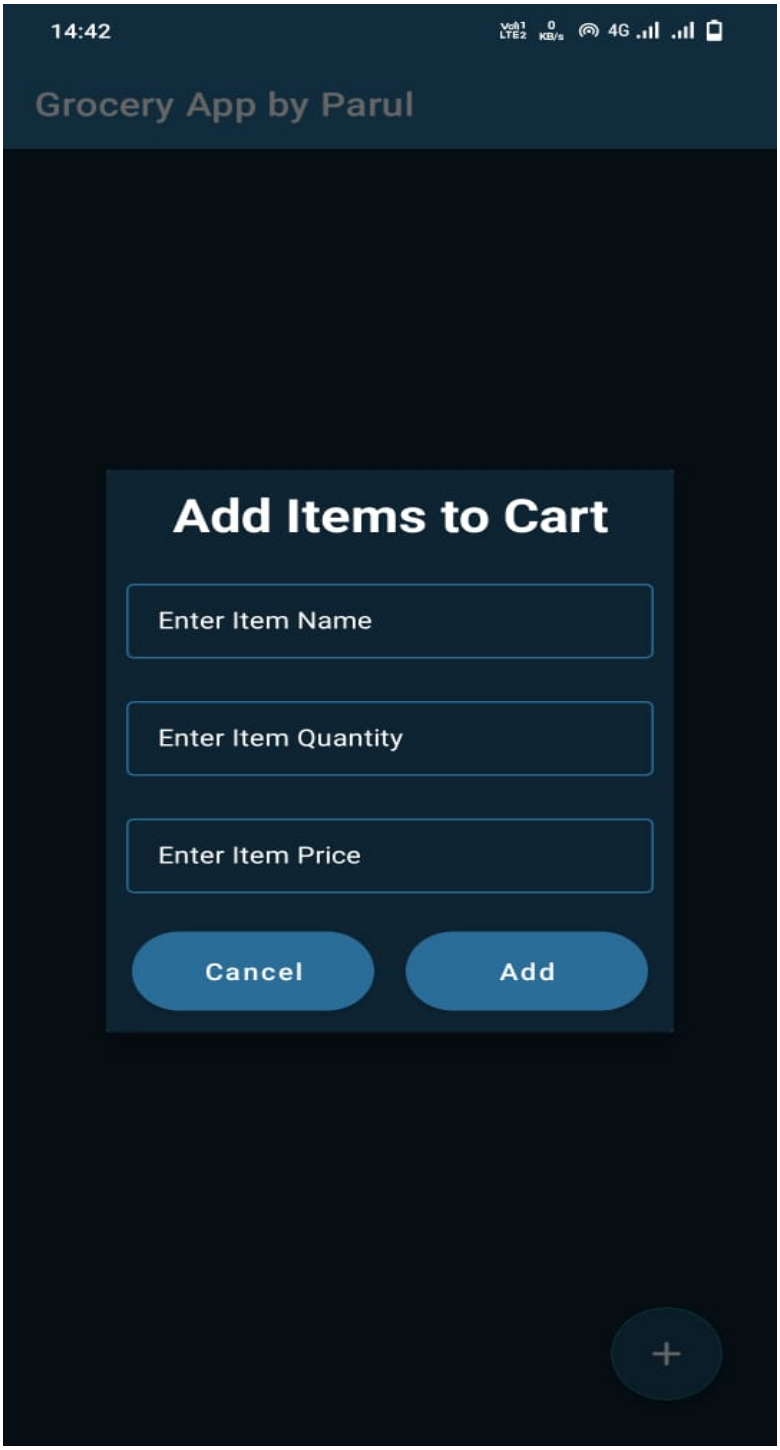
**Co-routines:** Coroutines are lightweight thread, we use a coroutine to perform an operation on other threads, by this our main thread doesn't block and our app doesn't crash.

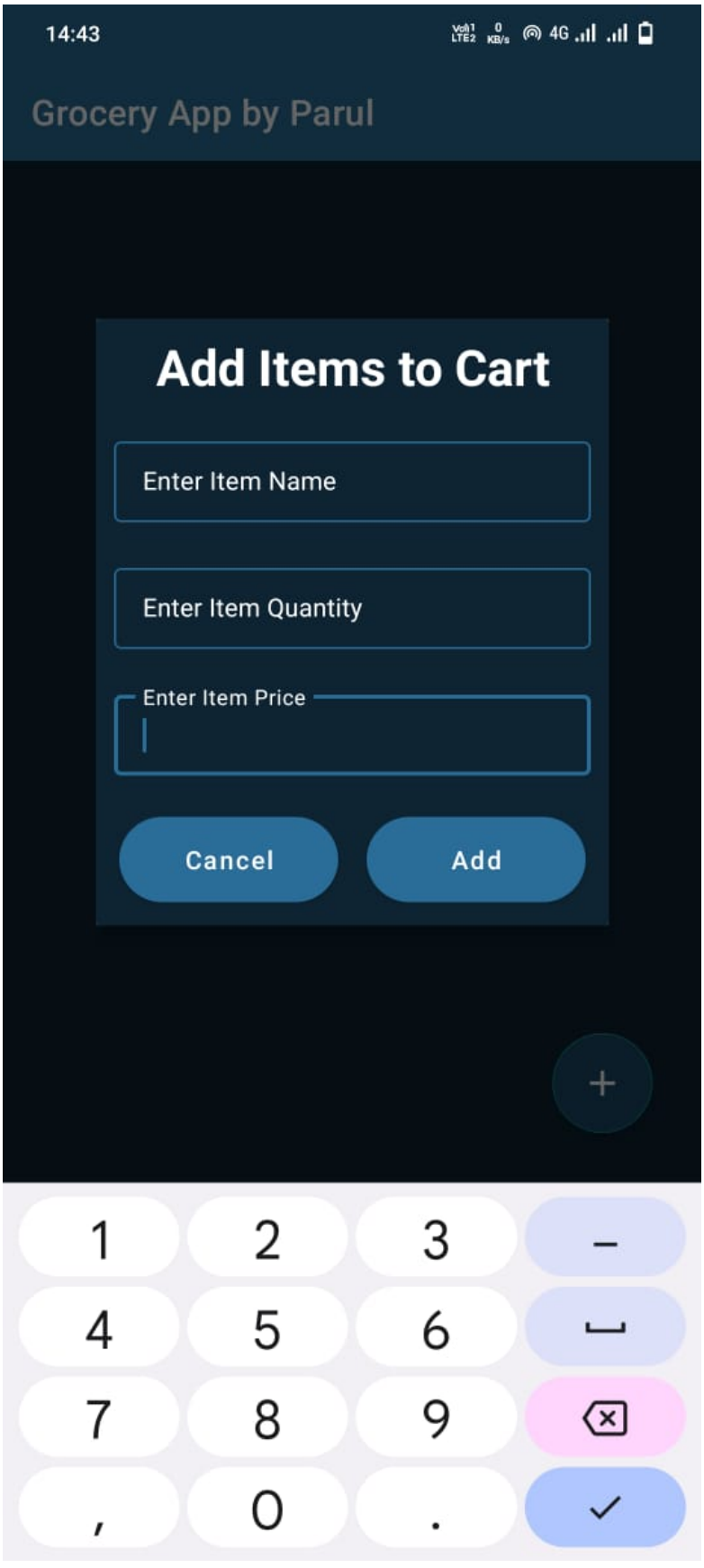
## 5. Flowchart

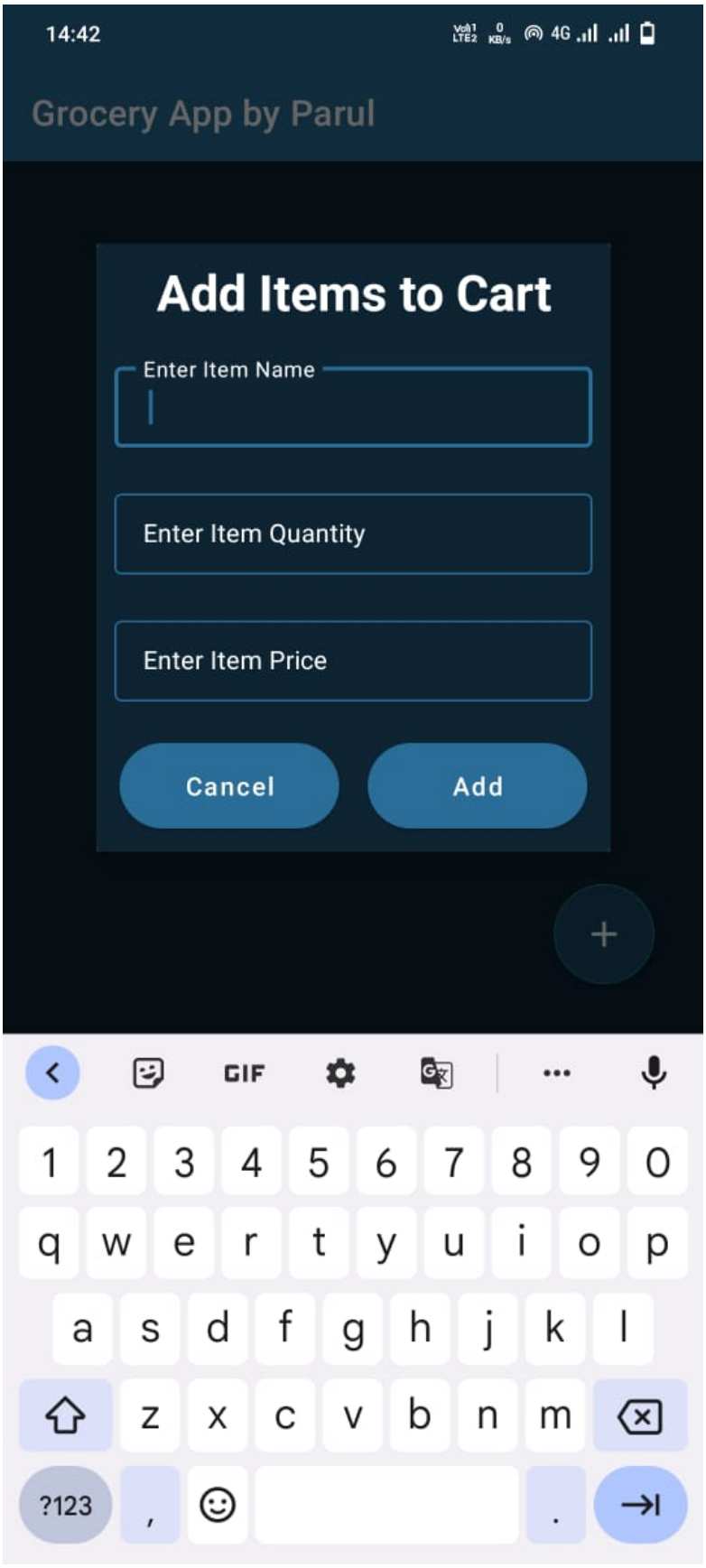
Below is a diagrammatic step by step representation in form of a flowchart for the project given to me.

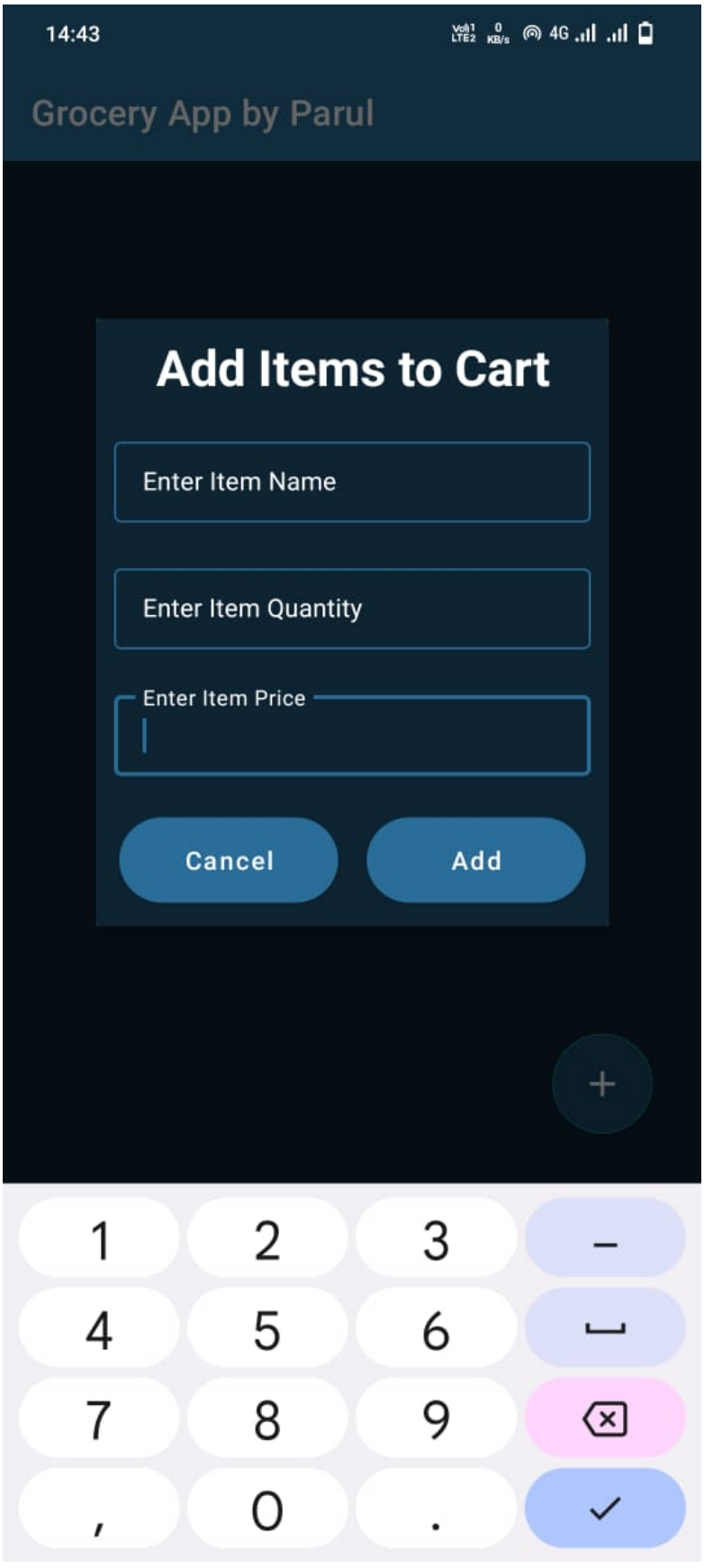


## 6. Result


14:42
Grocery App by Parul
Add Items to Cart
Enter Item Name
Enter Item Quantity
Enter Item Price
Cancel
Add


14:43
Grocery App by Parul
Add Items to Cart
Enter Item Name
Enter Item Quantity
Enter Item Price
Cancel
Add


14:42
Grocery App by Parul
Add Items to Cart
Enter Item Name
Enter Item Quantity
Enter Item Price
Cancel
Add
GIF


14:43
Grocery App by Parul
Add Items to Cart
Enter Item Name
Enter Item Quantity
Enter Item Price
Cancel
Add

## 7. Conclusion

This project helped me to clear my concepts on Room Database, Co-routines, 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.

## 8. Reference


- Google: https://www.google.com/
- GeeksforGeeks: https://www.geeksforgeeks.org/how-to-build-a-grocery-androidapp-using-mvvm-and-room-database/
- Android Developer: https://developer.android.com/codelabs/android-room-witha-view-kotlin#0
- YouTube: https://www.youtube.com/watch?v=vdcLb_Y71Ic
- SmartInternz: https://smartinternz.com/Student/guided_project_workspace/56568


## 9. Future Scope

Android has a vast scope in future world alone as well as combined with all other techs to make world more productive and useful along eith tech.