**PROJECT REPORT**

**BUILD A GROCERY ANDROID APP – PROJECT**

***Submitted by***

**KAVIARASAN M.S**

**Problem Statement :**

Giving individuals the ability to keep track of the groceries they purchase at markets is the goal of this initiative. For the majority of people, buying groceries is a daily or weekly activity. The issue is that individuals frequently forget or lose track of what they are purchasing, so they utilise old-fashioned methods like writing down their shopping lists on paper. I have addressed the issue by giving them access to a digital inventory of goods from which to select before visiting a store and by enabling them to keep track of prior purchases while on the go via an Android smartphone. People will be more productive as a result and save time.

**Introduction :**

  Grocery shopping is a regular chore that must be done on a daily, weekly. For that i create a user-friendly interface that allows users to build shopping lists and keep track of their purchases inside of an Android app, our project aids in making these tasks easier. Used MVVM (Model View View Model) architectural patterns, room database , coroutines, and Recycler View for the list of objects to be displayed.

**Technologies-Used :**

1. MVVM (Model View ViewModel)

MVVM architecture in android is used to give structure to the project’s code and understand code easily. MVVM is an architectural design pattern in android. MVVM treat Activity classes and XML files as View. This design pattern separates UI from its logic. Here is an image to quickly understand MVVM.

2) ROOM Database

Room persistence library is a database management library and it is used to store the data of apps like grocery item name, grocery item quantity, and grocery item price. Room is a cover layer on SQLite which helps to perform the operation on the database easily.

3)Recycler View

Recycler View is a container and it is used to display the collection of data in a large amount of data set that can be scrolled very effectively by maintaining a limited number of views.

4)Coroutines

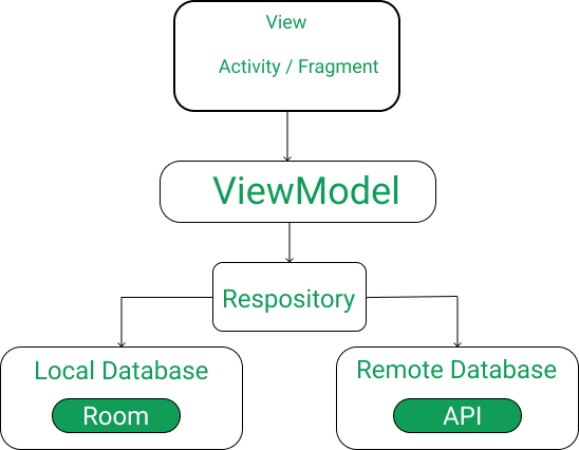
Coroutines are a lightweight thread, we use a coroutine to perform an operation on other threads, by this our main thread doesn’t block and our app doesn’t crash.

**LITERATURE SURVEY**

* **Low-profit margins-eating away profits:** Customers always prefer **Online Grocery app solution** as they expect good deals with less price. But after spending so much on shifting the store to online grocery app development but it is a too tough task to keep less cost for the products and sell them to customers which will ultimately bring loss to the company. Also, many people would like to test the vegetables physically rather than seeing them on the screen, which is impossible in online.
  + **Solution:** By including non-perishable items like packaged foods and personal care item etc. whose storage and delivery cost are less, you can save some expenses as well as satisfy the customer. You can merge the offline and online business where people can order online and collect them whenever possible from the store. This will minimise the operation cost to some extent.
* **Disorganised and Inefficient Delivery System:** Customers will never plan before they order from a grocery store, they order it when they like to eat something or need any fruits and vegetables at that moment and expect it to reach them without any delay. The customers who order from online grocery app solution will compromise with the spontaneous buying experience. Many people still prefer to visit the store rather than ordering things online. As most of the people are working, they are not available to collect their stuff when it is delivered home.
  + **Solution:** This can be solved only if the companies follow some delivery schedules. A buyer should be able to receive his order within 30 mins, which must be implemented by **Online grocery app development.** By this, the buyer can get a clear picture of the delivery so that they can manage their time.

## **THEORITICAL ANALYSIS :**

* Block diagram: Diagrammatic overview of the project.



**HARDWARE / SOFTWARE DESIGNING HARDWARE AND SOFTWARE REQUIREMENTS OF THE PROJECT.**

**HARDWARE USED:**

1. Laptop or PC with Android studio installed into it along with a good internet connection.
2. Android mobile phone for running our app(if not we can use Emulator as well.)

**SOFTWARE AND TECHNOLOGY REQUIREMENTS**:

* 1. Andorid studio.
  2. Github.
  3. Emulator. etc
  4. Kotlin.



package com.example.grocerylist.UI  
  
import android.os.Bundle  
import androidx.appcompat.app.AppCompatActivity  
import androidx.lifecycle.Observer  
import androidx.lifecycle.ViewModelProvider  
import androidx.recyclerview.widget.LinearLayoutManager  
import com.example.grocerylist.Adapter.GroceryAdapter  
import com.example.grocerylist.Database.Entity.GroceryItems  
import com.example.grocerylist.Database.GroceryDatabase  
import com.example.grocerylist.Database.GroceryRepository  
import com.example.grocerylist.R  
import kotlinx.android.synthetic.main.activity\_main.\*  
  
class MainActivity : AppCompatActivity() {  
  
  
 lateinit var ViewModel: GroceryViewModel  
 lateinit var list: List<GroceryItems>  
  
  
  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_main*)  
  
  
  
 val groceryRepository = GroceryRepository(GroceryDatabase(this))  
  
 val factory =  
 GroceryViewModelFactory(groceryRepository)  
  
// Initialised View Model  
 ViewModel = ViewModelProvider(this,factory).get(GroceryViewModel::class.*java*)  
  
 val groceryAdapter =  
 GroceryAdapter(*listOf*(), ViewModel)  
  
  
 rvList.*layoutManager* = LinearLayoutManager(this)  
 rvList.*adapter* = groceryAdapter  
  
// To display all items in recycler view  
 ViewModel.allGroceryItems().observe(this, *Observer***{** groceryAdapter.list = **it** groceryAdapter.notifyDataSetChanged()  
  
  
 **}**)  
  
// on ClickListener on button to open dialog box  
 btnAdd.setOnClickListener **{** GroceryItemDialog(this,object : DialogListener{  
 override fun onAddButtonClicked(item: GroceryItems) {  
 ViewModel.insert(item)  
 }  
 }).show()  
 **}** }  
  
  
}

**Activit\_main.xml:**

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="#ffffff"  
 android:orientation="vertical"  
 tools:context=".UI.MainActivity">  
  
  
 <Button  
 android:id="@+id/btnAdd"  
 android:layout\_width="67dp"  
 android:layout\_height="62dp"  
  
 android:layout\_marginBottom="32dp"  
 android:background="@drawable/free\_green\_add\_button\_icon\_12023\_thumb"  
 app:layout\_constraintBottom\_toTopOf="@+id/rvList"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.619"  
 app:layout\_constraintStart\_toEndOf="@+id/imageView" />  
  
 <ImageView  
 android:id="@+id/imageView5"  
 android:layout\_width="475dp"  
 android:layout\_height="772dp"  
 app:srcCompat="@drawable/rm222\_mind\_14"  
 tools:layout\_editor\_absoluteX="-32dp"  
 tools:layout\_editor\_absoluteY="45dp" />  
  
 <ImageView  
 android:id="@+id/imageView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="35dp"  
 android:src="@drawable/title"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.406"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/imageView6"  
 app:layout\_constraintVertical\_bias="0.159" />  
  
 <androidx.recyclerview.widget.RecyclerView  
 android:id="@+id/rvList"  
 android:layout\_width="match\_parent"  
 android:layout\_height="472dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.0"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/imageView"  
 app:layout\_constraintVertical\_bias="1.0">  
  
  
 </androidx.recyclerview.widget.RecyclerView>  
  
 <ImageView  
 android:id="@+id/imageView6"  
 android:layout\_width="392dp"  
 android:layout\_height="116dp"  
 app:srcCompat="@drawable/coollogo\_com\_27713699"  
 tools:layout\_editor\_absoluteX="3dp"  
 tools:layout\_editor\_absoluteY="5dp" />  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="394dp"  
 android:layout\_height="39dp"  
 android:background="@color/blue"  
 android:text=" ITEM QUANTITY PRICE"  
 android:textStyle="bold"  
 app:layout\_constraintBottom\_toTopOf="@+id/rvList"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/imageView" />  
  
  
</androidx.constraintlayout.widget.ConstraintLayout>

**Note :-** Since the page limit is exceeding I can't put the whole source code here

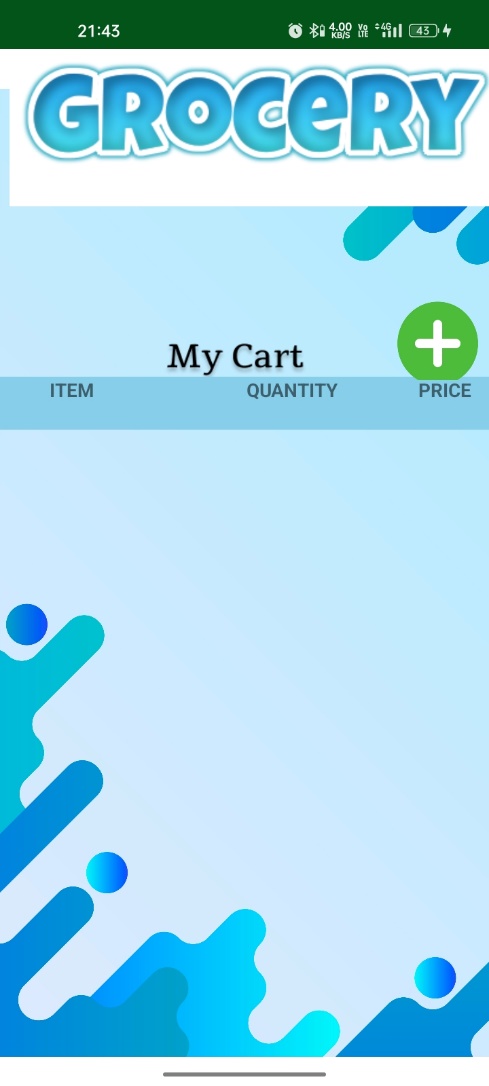
. Please check the drive link or the github link below for full code.

**RESULT:**

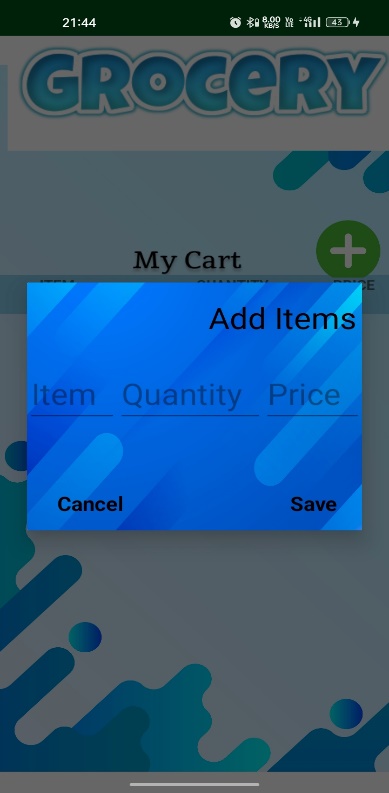
1. Logo of Grocery application.



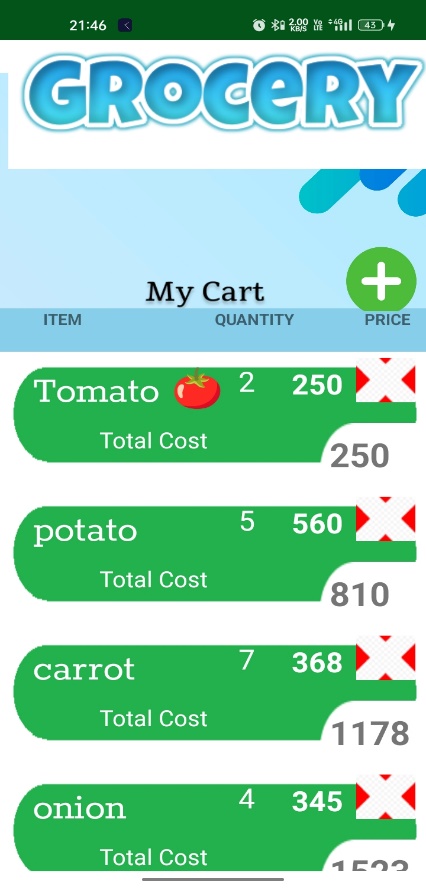
b) Layout of Home screen of Grocery Application.



1. Adding items to Cart.



1. Cart after adding Items in it.



**ADVANTAGES:**

* + User can purchase grocery products through his mobile phones that support android.
  + User does not have to wait in long queue and does not have to struggle with trolleys.
  + User can coolly sit at home and purchase the products according to his like.

**DISADVANTAGES:**

* + This system won’t work in mobile phone that does not support android.
  + Product quality would differ from manually purchasing product in a supermarket.

**APPLICATIONS:**

* + This application can be used by any user who loves to shop and this application can be used by many house wives.

**GitHub URL**

<https://github.com/smartinternz02/SPSGP-104032-Virtual-Internship---Android-Application-Development-Using-Kotlin>

* Google Developer Profile:

<https://g.dev/mskavi>

ACCOUNT ID’s

GitHub id:

<https://github.com/kavi3210>

SmartInternz id:

<https://smartinternz.com/student-profile/feed/U0IyMDIxMDE0NzIyOQ==>

SmartInternz registered email id: [sit20cs031@sairamtap.edu.in](mailto:sit20cs031@sairamtap.edu.in)

**ACKNOWLEGEMENT**

I want to express my sincere gratitude to Mr. Sandeep Doodigani for his excellent guidance and support in seeing my project through to completion. Additionally, I want to thank him for giving me the chance to work on a project with the theme "Grocery App." Without their assistance and suggestions, this project could not have been completed.

Kaviarasan.