Smart Internz

Grocery App Report

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 don't 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.

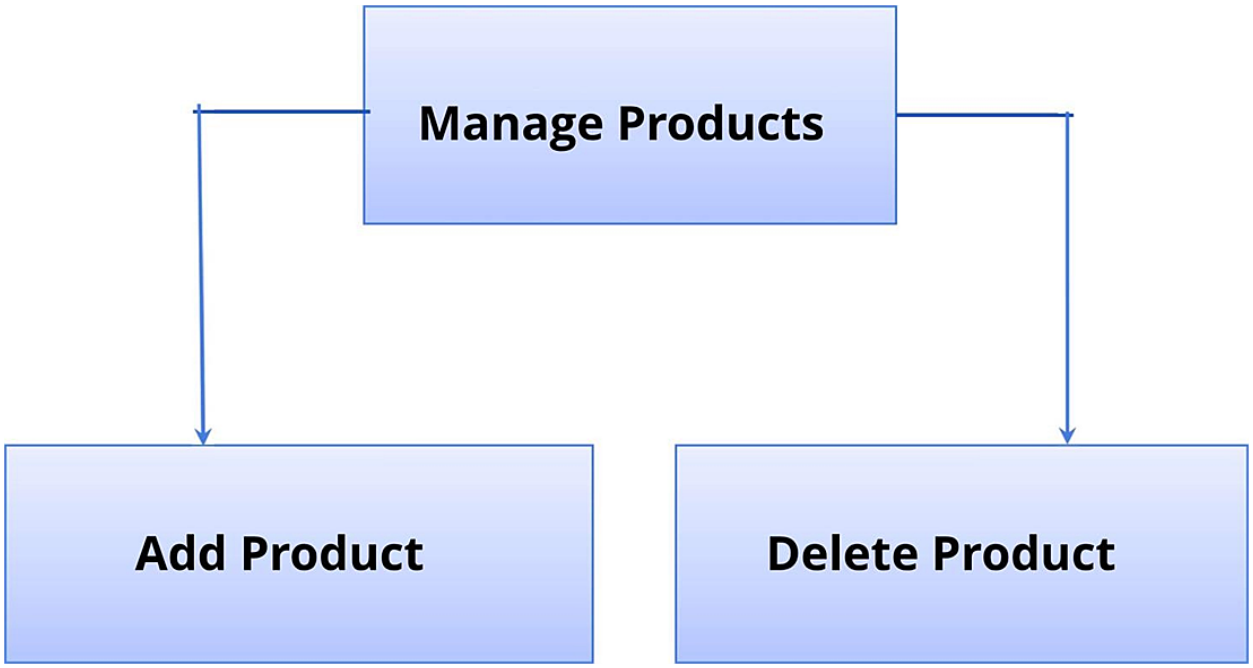
2.2 Proposed Solution

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

Smart Internz

3.Theoretical Analysis

3.1 Block Diagram



3.2 Hardware/Software designing

·Windows 10 OS

.Android Studio

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.

Smart Internz

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.

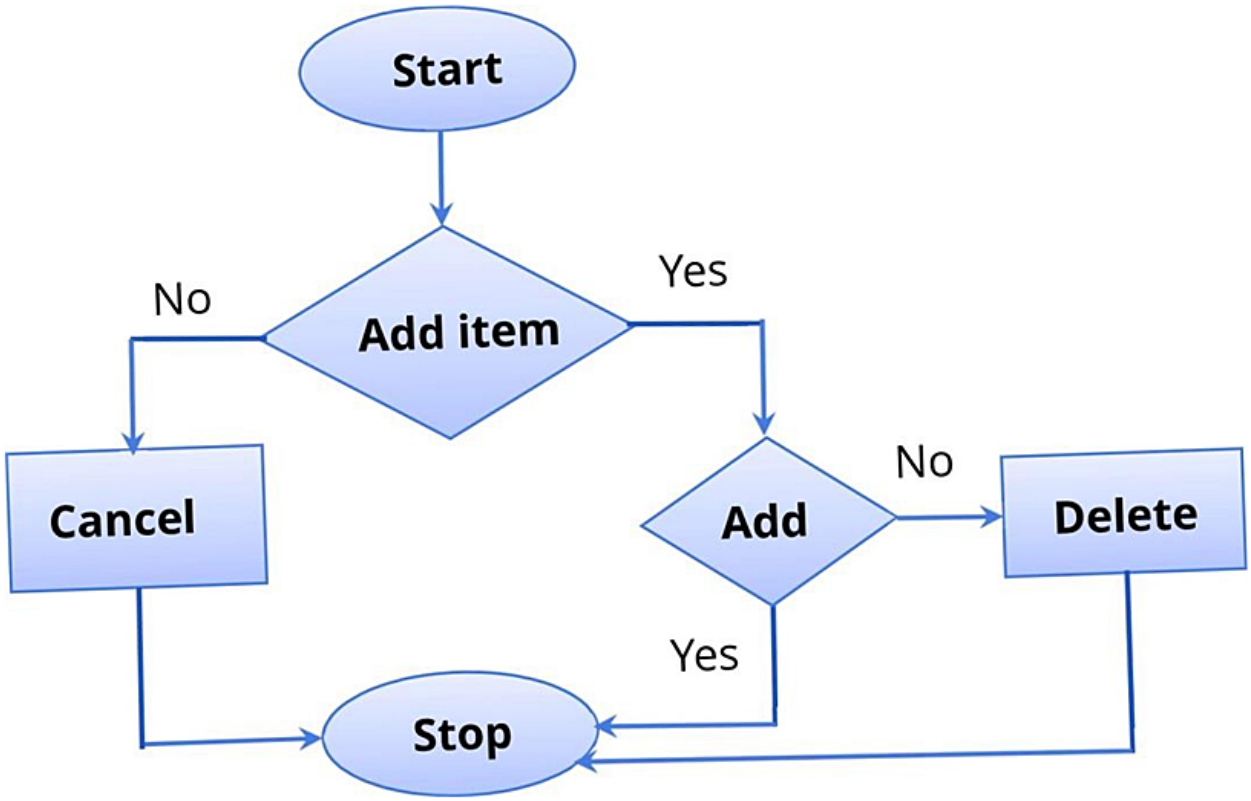
DAO: Data access object. A mapping of SQL queries to functions. When you use a DAO, you call the methods, and Room takes care of the rest.

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.

Smart Internz

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.

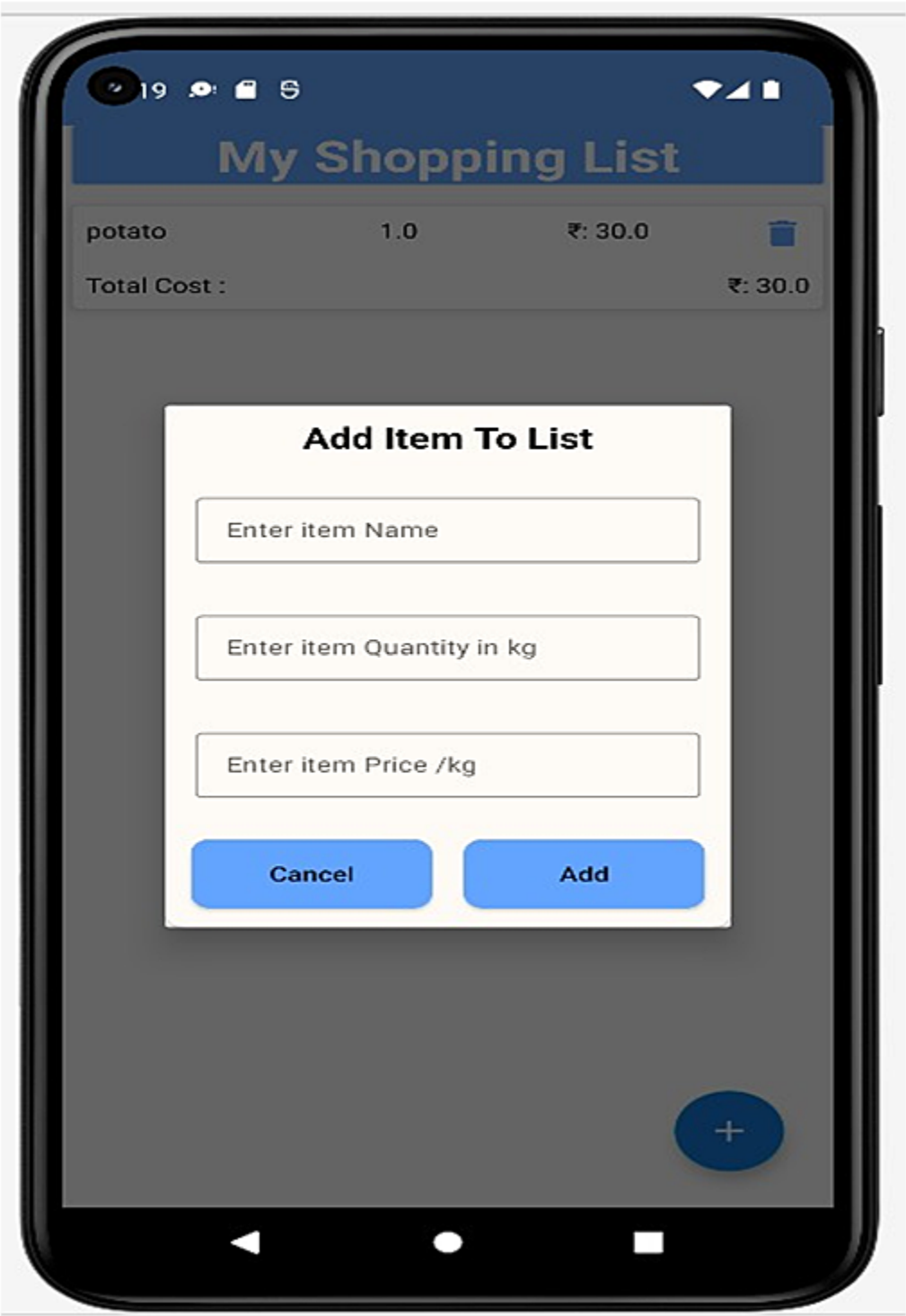
5. Flowchart



6.Result

Smart Internz

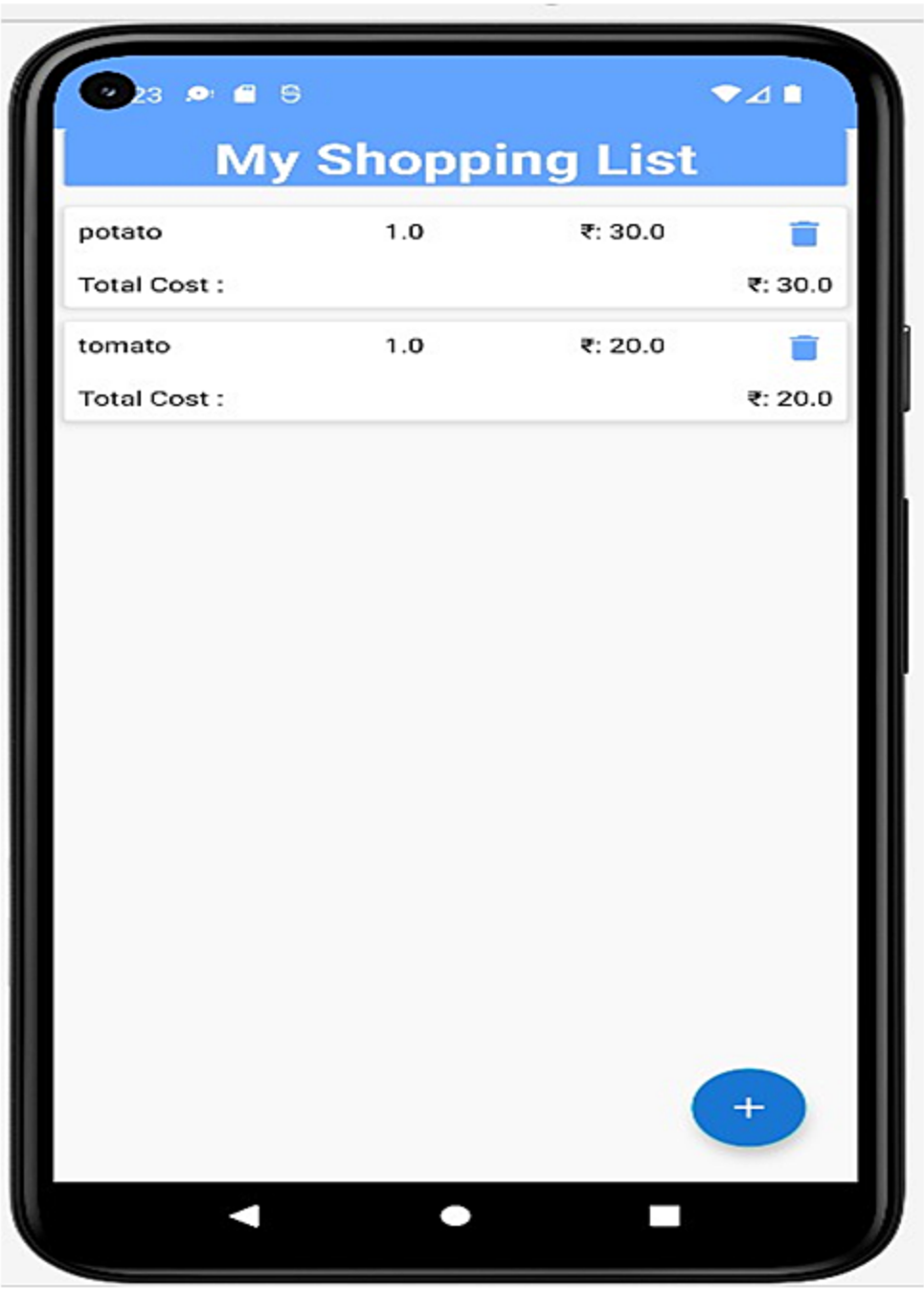
Emulator: 2 API 30



Event Log Layout inspector

Smart Internz

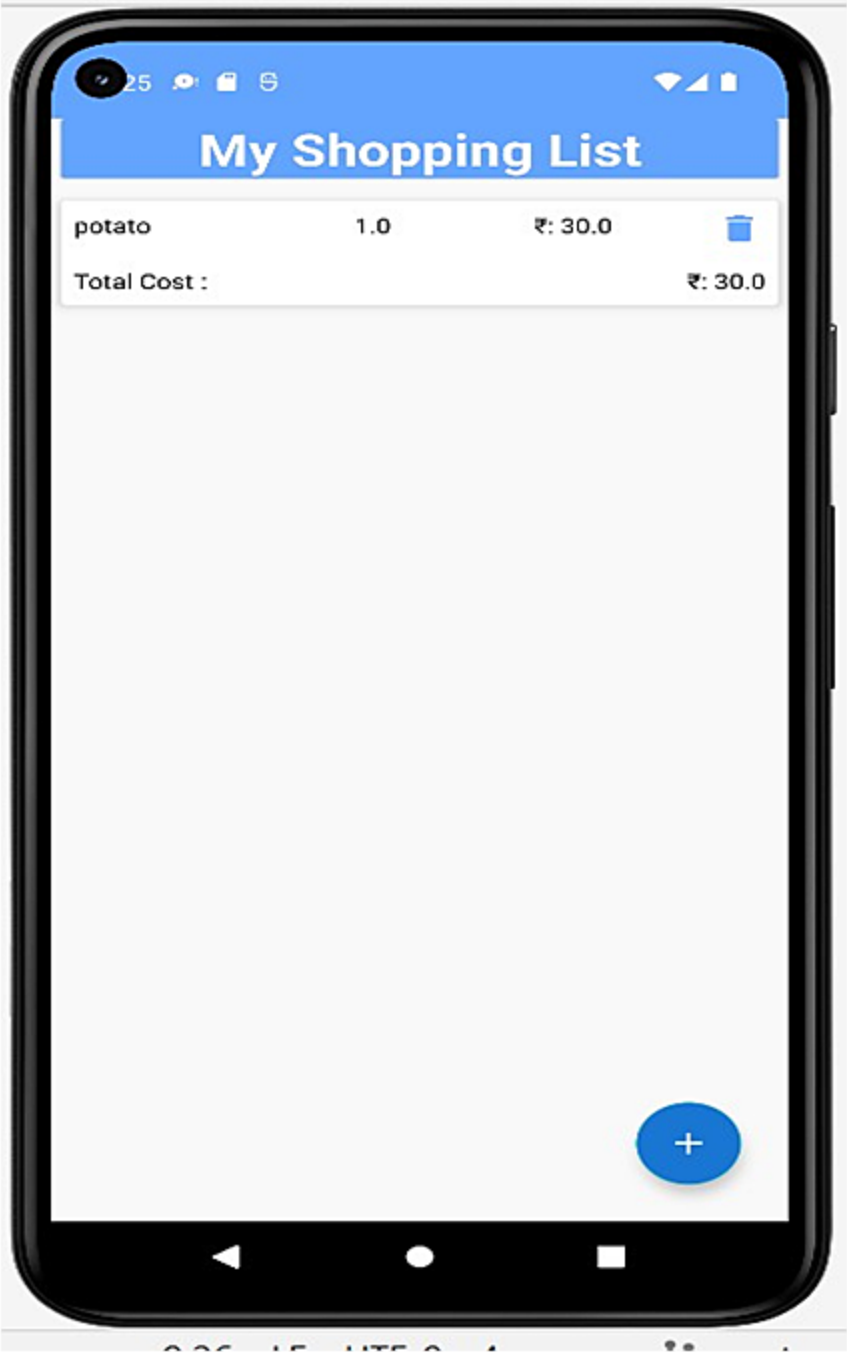
Emulator: 2 API 30



Event Log Layout inspector

Smart Internz

Emulator: 2 API 30



Event Log Layout inspector

Smart Internz

7. Conclusion

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.

8. Reference

Google: https://www.google.com/

Geeksforgeeks: https://www.geeksforgeeks.org/how-tobuild-a-grocery-androidapp-using-mvvm-and-roomdatabase/

Android Developer: hÜps://developer.android.com/codelabs/android-roomwitha-view-kotlin#0

YouTube: hÜps://www.youtube.com/watch?v=vdcLb_Y71 lc

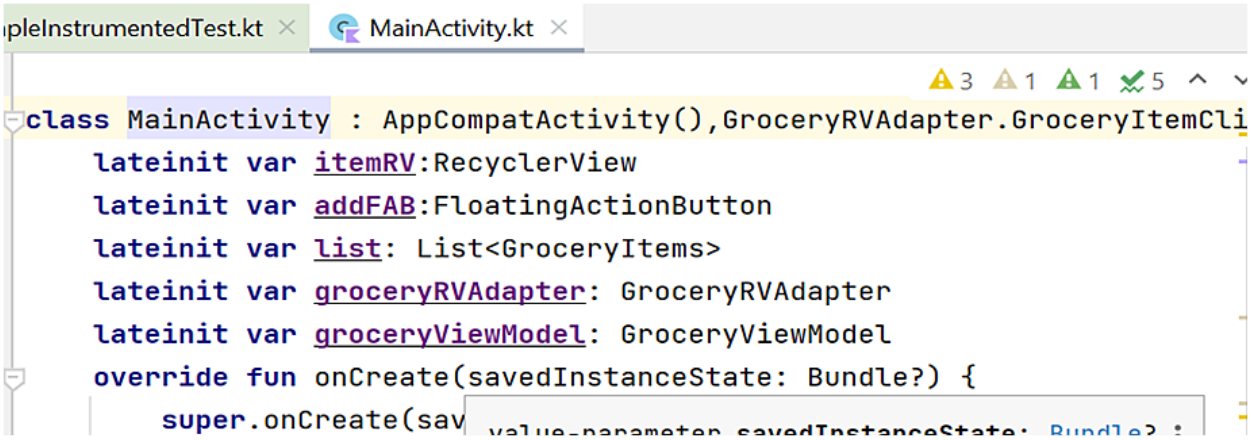
SmartInternz: hÜps://smartinternz.com/Student/guided_project_workspac e/55908

9. Appendix

9.1 Source Code

MainActivity.kt

Smart Internz



t11po arid r Oidx.11 recycle . Observer
rT.

rnpo androidx . li fecycle . ViewMode1Provider
rt

impo androidx . recyclerview . widget .
rt LinearLayoutManager

impo androidx . recyclerview . widget .
rt RecyclerView

impo com.google . android . material . float on
rt ingactionbutton .FloatingAct

•lass MainActivity : AppCompatActivity ( ) ,
GroceryRVAdapter . GroceryTtemC1ickTnterface {

latein v itemsRV: RecyclerView
it ar

latein v adclFAB:
lt ar FloatingActionButton

latein v list : List
lt ar

latein v groceryRVAdapter:GroceryRV
lt ar Adapter

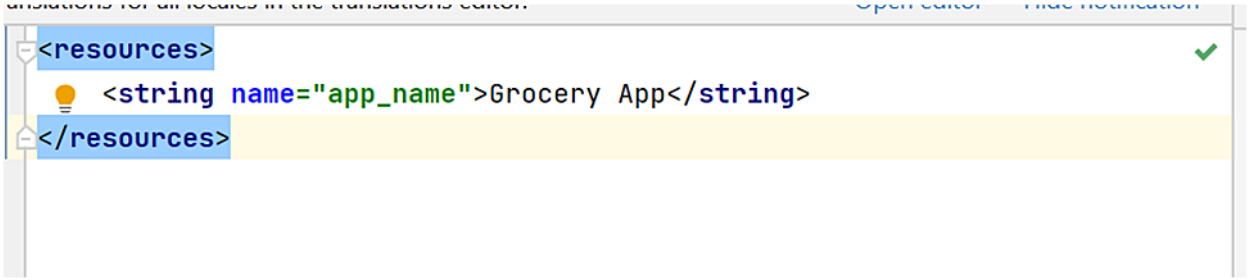
```
latein v groceryViewModal:
    it ar GroceryViewMode1
          onCreate
overri f (savedInstanceState:
    de un Bundle?)
    super . onCreate
    (savedInstanceState)
    setContentView (R.
    layout . acti vity_main)

    itemsRV =
    findViewById (R.id.
    idRVItems) addFAB =
    findViewByTd (R. id.
    idFABAdd) list -
    ( )
    graceryRVAdapter = GroceryRVAdapter
(1 i st, this) itemsRV . layoutManager -
LinearLayoutManager (this) itemsRV.
adapter- groceryRVAdapter val
groceryRepository - GroceryRepository
(GroceryDatabase (this) ) val factory =
GroceryViewMode1Factory
(groceryRepository) groceryViewMoclal =
ViewMode1Provider (this, factory) . get
(GroceryViewModeI : :class . java)
groceryvlewmodal . getA11GroceryIterns (
) . observe (this, Observer {
groceryRVAdapter.1ist - groceryRVAdapter
. notifyDataSetChanged ( )
    addFAB .
       set
```

```
        OnC1ickListen
        er openDia10g
        ( )



    fun   openDiaIog
        (    )    val
        dialog
        Dialog
        (this)
        dialog      .
        setConter,tV
        iew       (R.
        layout .
          cancelBtndialog .findViewById<Button> (R. id.
          idBtnCance1)
        val addBtn = dialog . (R. id. idBtnAdd)val
        itemEdt - dialog
        .   (R. id. idEdt Tternmame)
        dxalog .   (R.    ce) val
        itemQuantityEdt = dialog .
        findViewById<EditText > (R . icl.
        idEdtItemQuanti ty) cancelBtn .
        setOnC1ickListener { dialog
        . dismiss ()


        addBtn  .  setOnC1ickListener  val
itemName:  String  =  itemEdt.  text  .
tostring  (  )  val  itemPrice:  String  -
itemPriceEdt  .  text  .  toString  (  )  val
```

```
itemQuantity: String = itemQuantityEdt .
text . toString ( ) val qty: Int =
itemQuantity . tolnt ( ) v al pr: Int
itemPrice . tornt ( ) i f (itemName .
isNotEmpty() S S itemPrice. isNotEmpty()
itemQuantity . isNotEmpty() ) val items
= Groceryltems (itemName, qty, pr)
groceryViewModa1. insert (items)
        Toast . makeText
(applicationCont ext,Item Inserted. .
roast.LENGTH_SHORT) . show ( )
groceryRVAdapter .
          notifyDataSetCh
          anged ( )
          dialog.dismiss
          ( )

          Toast .
             makeTe
             xt (
             applica
             tionCon
             text,
            'l Please
             Enter all
             data. . '1 ,
             Toast .
             LENGTH-SHORT
          ) . show ( )
```

```
      dialog . show ()


   override  fun  onltemC1ick
      (groceryltems:  Grocery
      Items)groceryV1ewModa1 .
      delete (groceryltems
      ) groceryRVAdapter . not
    ifyDat aSetChanged ( ) Toast .
    makeText(applicationContext,It
    em Deleted. .

roast.LENGTH-SHORT) . show ( )
```

activity_main.xml



Note :- Since the page limit is exceeding I can't put the whole source code here. Pleasecheck the drive link or the github link below for full code.

Drive link:
https://drive.google.com/file/d/1JbBKwASgGicXqYgrzSk1Jbf4swBvvBpV/view?usp=sharing

Github link:https://github.com/smartinternz02/SPSGP-54938-Virtual-Internship---Android-Application-Development-Using-Kotlin