Project Title

Build A Grocery Android App - Project

Problem Statement

As we can't remember everything, users frequently forget to buy the things they want to buy. However, with the assistance of this app, you can make a list of the groceries you intend to buy so that you don't forget anything

Introduction

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

ABSTRACT

Shopping is one of the activities that some people consider part of their life, while others do not even think of it. This comparison makes us discover people's problems with shopping. People have shopping problems such as limited time, expats in foreign countries without cars, a transportation issue, people consider physical shopping as a waste of time, health issues, longdistance to market. And the difficulty in obtaining some items. As the problems mentioned above, we have explored our idea, which is related to personal shopping. Therefore, we have built an application that combines different market shops, i.e. (Malls, supermarkets, and pharmacies). This personal grocery shopping is an innovative app that allows the customers to get all their needs and suggest items based on previous history. Then deliver items to their doorstep and can facilitate online shopping procedure where customers can browse unlimited products all at one time. This work supports people in exploiting their time to be safer and more accessible than wasting it physically. Moreover, people can order the product from home instead of going around for long distances for shopping. In addition, this app could help people who are facing health problems and unable to buy something

physically to avoid future problems. Finally, some people do not have transportation methods for shopping, and they should keep pace with the evolution.

OBJECTIVE

The main aim of this project is to list the items so that whenever users go to grocery stores, users will not be able to forget their items and this grocery application helps the users to tackle their day to day chaos more effortlessly.

Technologies-Use

1) MVVM (Model View View Model) :

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 Databas:

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) RecycleView:

RecyclerView 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) Corountines:

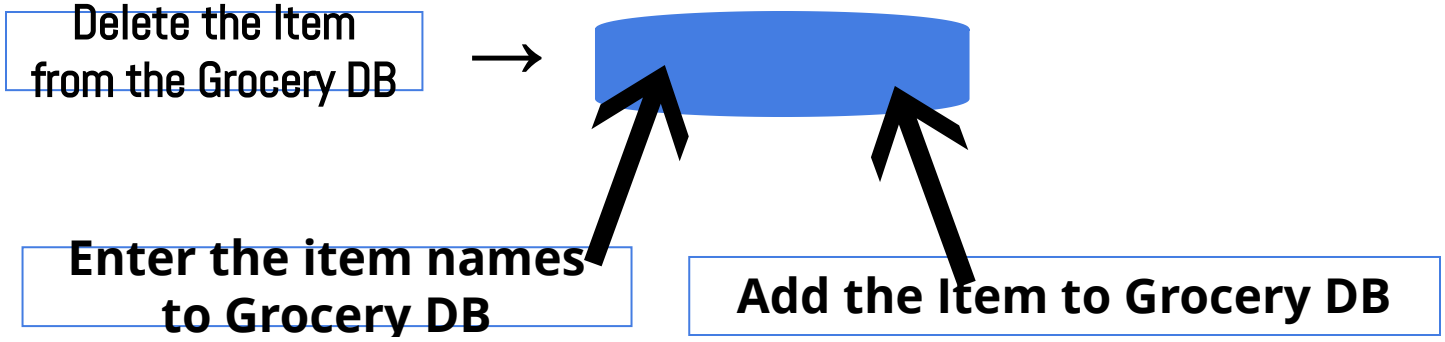
Coroutines are a lightweight thread, we use a coroutine to perform an operation on other threads, by this our main thread doesn't block and our app doesn't crash.

BACKGROUND

The grocery cart application project will help the user or admin to store the list of items in proper sequence. User/Admin can add and remove the items in the list according to his/her will.

- UI DESIGN IN THE ANDROID PLATFORM
- ANDROID APPLICATION DEVELOPMENT
- DATABASE CONNECTION TO STORE USER DATA

CONTEXT DIAGRAM



SOFTWARE

The Software Package is developed using Kotlin and Android Studio, basic SQL commands are used to store the database. Operating System: Windows 11 Software: Kotlin and Java Emulator: Pixel 4 API 30 3.2

HARDWARE RAM:

16GB RAM

Step By Step Process

Step 1:
Create a New Project To create a new project in Android Studio please refer to How to Create/Start a New Project in Android Studio. Note that select Kotlin as the programming language

Step 2:
Before going to the coding section first you have to do some pre-task 6 Before going to the coding part first add these libraries in your gradle file and also apply the plugin as 'kotlin-kapt'. To add these library go to Gradle Scripts > build.gradle (Module: app).

Step 3:
Implement Room Database a) Entities class The entities class contains all the columns in the database and it should be annotated with @Entity (tablename = "Name of table"). Entity class is a data class. And @Column info annotation is used to enter column variable name and datatype. We will also add Primary Key for auto-increment. Go to app > java > com.example.application-name. Right-click on com.example.application-name go to new and create Kotlin file/class and name the file as GroceryEntities. See the code below to completely understand and implement. b) DAO Interface The DAO is an interface in which we create all the functions that we want to implement on the database. This interface also annotated with @Dao. Now we will create a function using suspend function which is a coroutines function. Here we create three functions, First is the insert function to insert items in the database and annotated with @Insert, Second is for deleting items from the database annotated with @Delete and Third is for getting all items annotated with @Query. Go to the app > java > com.example.application-name. Rightclick on com.example.application-name go to new and create Kotlin file/class and name the file as GroceryDao. See the code below to implement. c) Database class Database class annotated with @Database(entities = [Name of Entity class.class], version = 1) these entities are the entities array list all the data entities associating with the database and version shows the current version of the database. This database class inherits from the Room Database class. In GroceryDatabase class we will make an abstract method to get an instance of DAO and further use this method from the DAO instance to interact with the database. Go to the app > java > com.example.application-name. Right-click on com.example.application-name go to new and create Kotlin file/class as GroceryDatabase.

Step 4:
Now we will implement the Architectural Structure in the App a) Repository Class The

repository is one of the design structures. The repository class gives the data to the ViewModel class and then the ViewModel class uses that data for Views. The repository will choose the appropriate data locally or on the network. Here in our Grocery Repository class data fetch locally from the Room database. We will add constructor value by creating an instance of the database and stored in the db variable in the Grocery Repository class. Go to the app > java > com.example.application-name. Right-click on com.example.application-name go to new and create Kotlin file/class as GroceryRepository. Go to app > java > com.example.application-name. Right-click on com.example.application-name go to new and create a new Package called UI and then right-click on UI package and create a Kotlin file/class. b) ViewModel Class ViewModel class used as an interface between View and Data. Grocery View Model class inherit from View Model class and we will pass constructor value by creating instance variable of Repository class and stored in repository variable. As we pass the constructor in View Model we have to create another class which is a Factory View Model class. Go to app > java > com.example.application-name > UI. Right-click on the UI package and create a Kotlin file/class and name the file as GroceryViewModel. 7 c) FactoryViewModel Class We will inherit the Grocery ViewModel Factory class from ViewModelProvider. NewInstanceFactory and again pass constructor value by creating instance variable of GroceryRepository and return GroceryViewModel (repository). Go to the app > java > com.example.application-name > UI. Right-click on the UI package and create a Kotlin file/class name it GroceryViewModelFactory.

Step 5:
Now let's jump into the UI part In the activity_main.xml file, we will add two ImageView, RecyclerView, and Button after clicking this button a DialogBox open and in that dialog box user can enter the item name, item quantity, and item price.

Step 6:
Let's implement RecyclerView. Now we will code the UI part of the row in the list. Go to app > res > layout. Right-click on layout, go to new, and then add a Layout Resource File and name it as GroceryAdapter.We will code adapter class for recycler view. In the GroceryAdapter class, we will add constructor value by storing entities class as a list in list variable and create an instance of the view model. In Grocery Adapter we will override three functions: onCreateViewHolder, getItemCount, and onbindViewHolder, we will also create an inner class called grocery view holder. Go to the app > java > com.example.applicationname. Right-click on com.example.application-name go to new and create a new Package called Adapter and then right-click on Adapter package and

create a Kotlin file/class name it GroceryAdapter.

Step 7:
To enter grocery item, quantity, and price from the user we have to create an interface. To implement this interface we will use DialogBox. First create UI of dialog box. In this dialog box we will add three edit text and two text view. Three edit text to enter grocery item name, quantity and price. Two text view one for save and other for cancel. After clicking the save text all data saved into the database and by clicking on the cancel text dialog box closes. Go to the app > res > layout. Right-click on layout, go to new and then add a Layout Resource File and name it as GroceryDialog. To add a clicklistener on save text we have to create an interface first in which we create a function. Go to the app > java > com.example.applicationname > UI. Right-click on the UI package and create a Kotlin file/class and create an interface name it as DialogListener.

Step 8:
In this final step we will code in our MainActivity. In our MainActivity, we have to set up the recycler view and add click listener on add button to open the dialog box.

Conclusion & Future Scope

This grocery application will help to store the list of data items include name of item, price and quantity required. Admins store his/her data in the list, the grocery application very helpful to users

Future Scope:

This application helps to store the list of items by Admin. In Future we can also add scheduled addition of items according to requirement of user. The Features are:
✓ Add User Panel
✓ Add Admin Panel
✓ Provide Login Authentication
✓ Add Image to user Product and Rating

URL’s

- **GitHub Code URL**

https://github.com/smartinternz02/SPSGP-80910-Virtual-Internship---Android-Application-Development-Using-Kotlin.git

- **Demo URL**

https://drive.google.com/file/d/1lGHHvJDTp_i9injGUF9Kzn_j2_k7HW4y/view?usp=drivesdk

- **Google Developer profile**

https://g.dev/madhasaikumar

ACCOUNT ID’s

- **GitHub Code URL**

https://github.com/Madhasai21

- **smartIntern id:**

https://smartinternz.com/student-profile/feed/U0IyMDIyMDE3MzQ5MQ==

- **SmartInternz registered email id:**

madhasaikumar2230@gmail.com

ACKNOWLEGEMENT

I would like to convey my heartfelt gratitude to Mr Sandeep Doodigani for his tremendous direction and assistance in the completion of my project.

I would also like to thank him for providing me with this wonderful opportunity to work on a project with the topic Grocery App. This project would not have been accomplished without their help and insights.

MADHA SAIKUMAR