Virtual Internship - Android Application Development Using Kotlin

About: I'm Hareesh Narayana Naik. I recently participated Virtual Internship which is based on Android Application Development Using Kotlin. **SBID : SB20220205308**

Mobile Application:

Mobile application development is the set of processes and procedures involved in writing software for small, wireless computing devices, such as smartphones and other hand-held devices.

Like web application development, mobile application development has its roots in more traditional software development. One critical difference, however, is that mobile apps are often written specifically to take advantage of the unique features of a particular mobile device. For example, a gaming app might be written to take advantage of the iPhone's accelerometer or a mobile health app might be written to take advantage of a smartwatch's temperature sensor.

Google developer

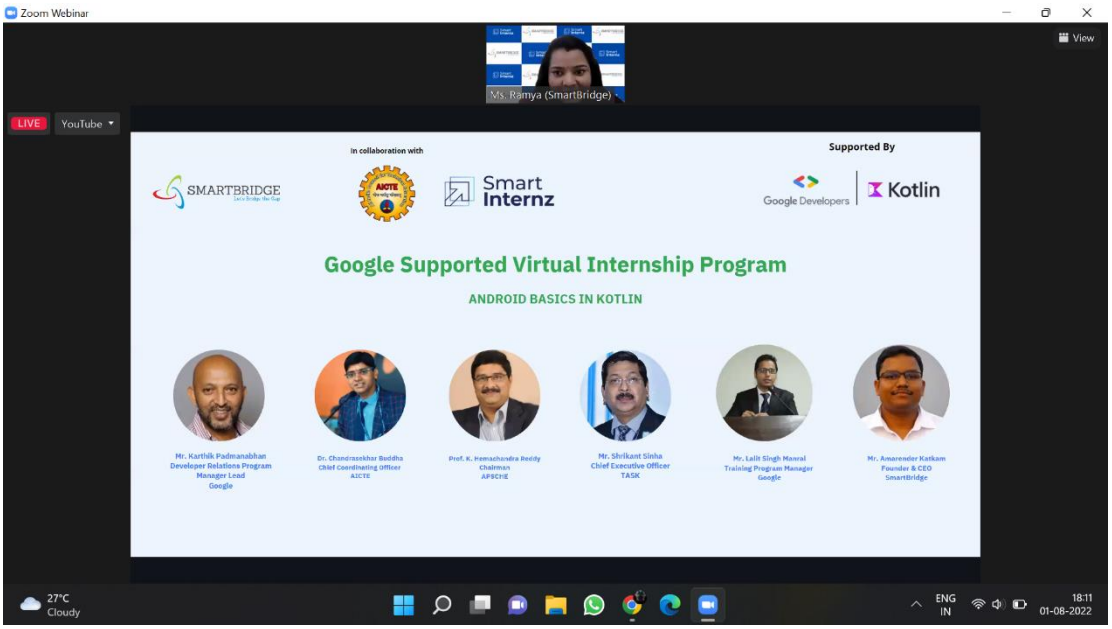
About Kotlin:

Kotlin is a general purpose, free, open source, statically typed "pragmatic" programming language initially designed for the JVM (Java Virtual Machine) and Android that combines object-oriented and functional programming features. It is focused on interoperability, safety, clarity, and tooling support. Versions of Kotlin targeting JavaScript ES5.1 and native code (using LLVM) for a number of processors are in production as well.

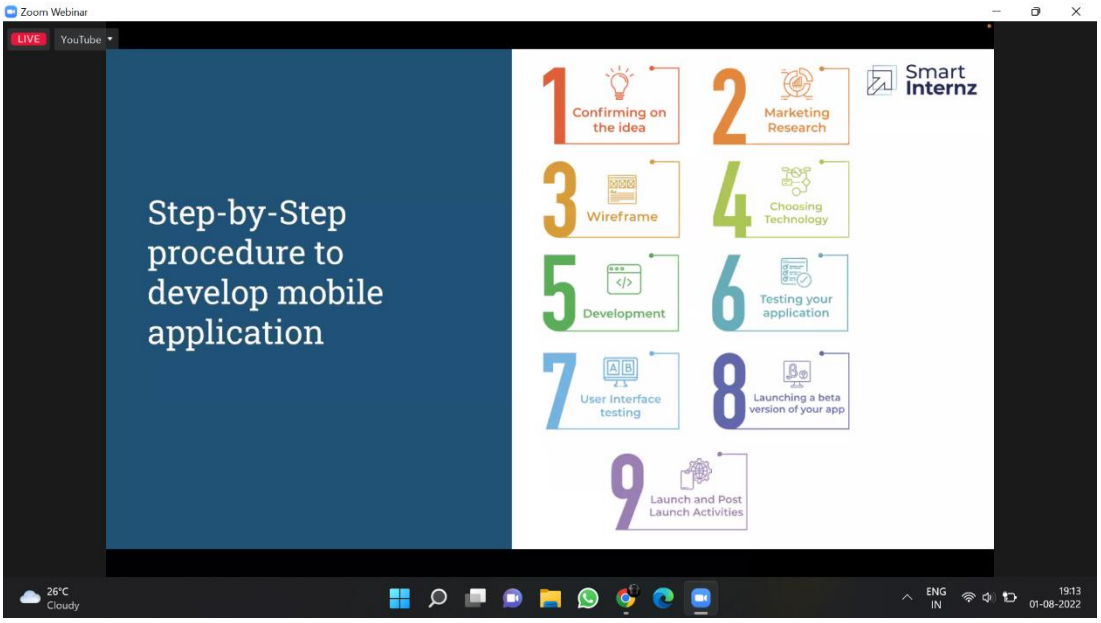
About the Program:

Before starting this program the instructors conducted webinar where they explained complete procedure of how we do this internship. I attended this webinar and understand the hole procedure.

The some pictures are given below,

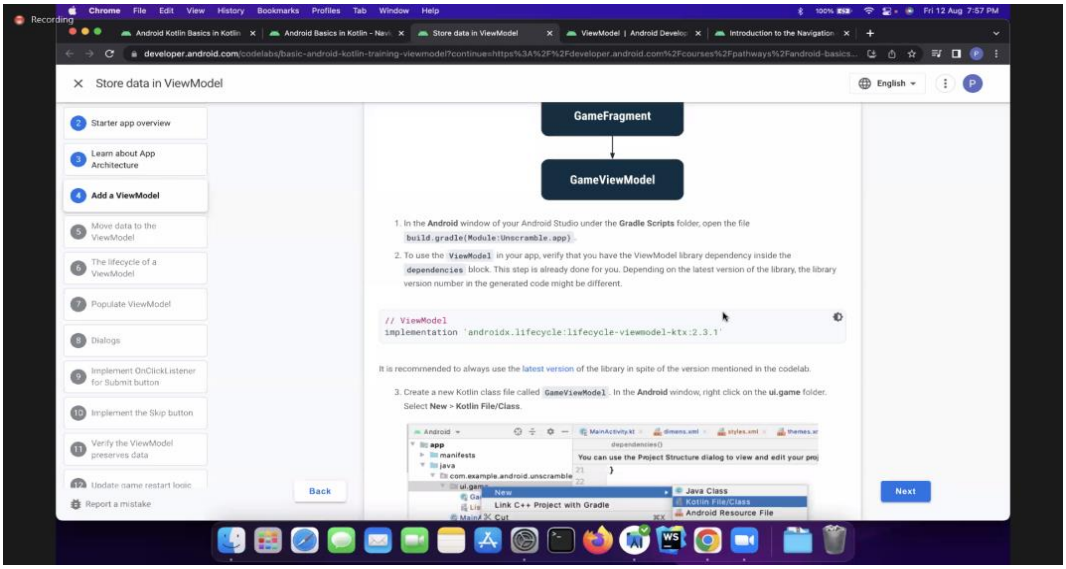


Google developer

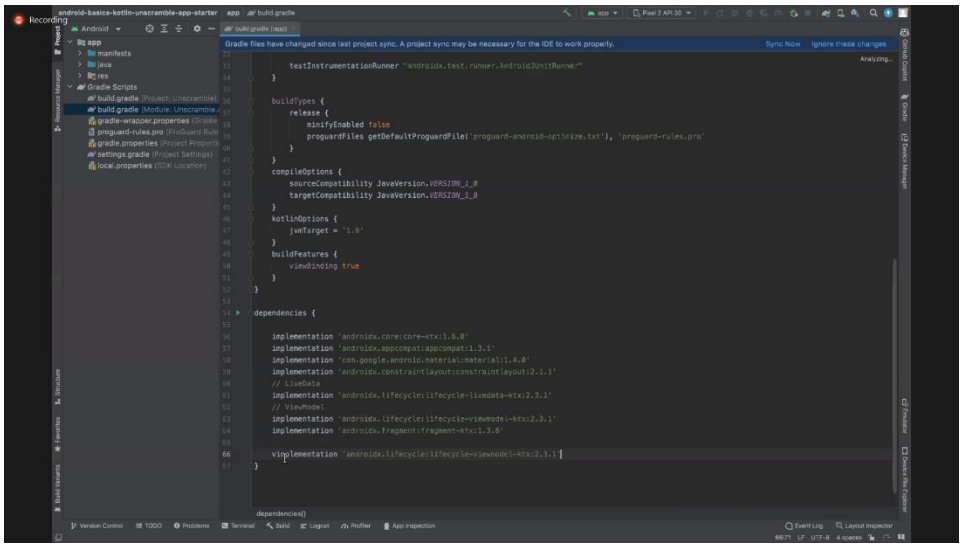


Then they will given the day wise calendar. In there we have live sessions and self based learning modules. The one of the well known instructor taken the live sessions and explained all about the concept which is there in self based modules and I followed all his sessions.

The some pictures of live session is given below,



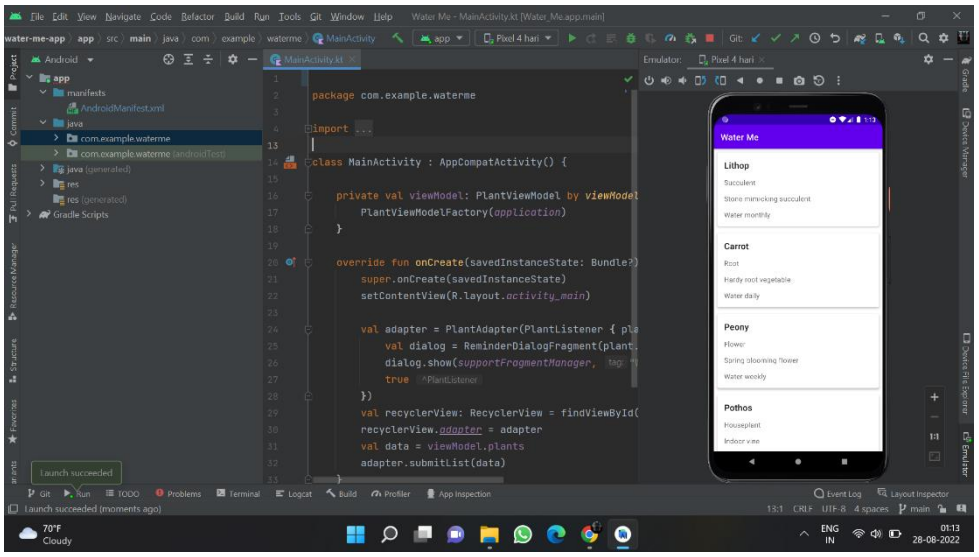
Google developer



We have to complete all 6 self based learning modules first then they will assign the project which is we have to do.

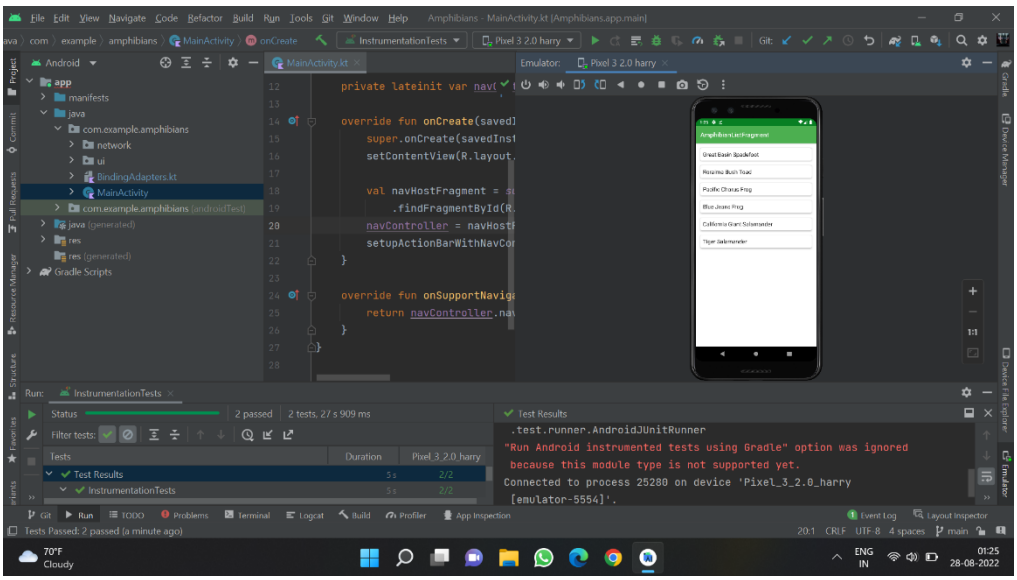
The all 6 modules has a project and I completed all the projects which is there in self based modules. Pictures are given below,

1.Water_Me app:

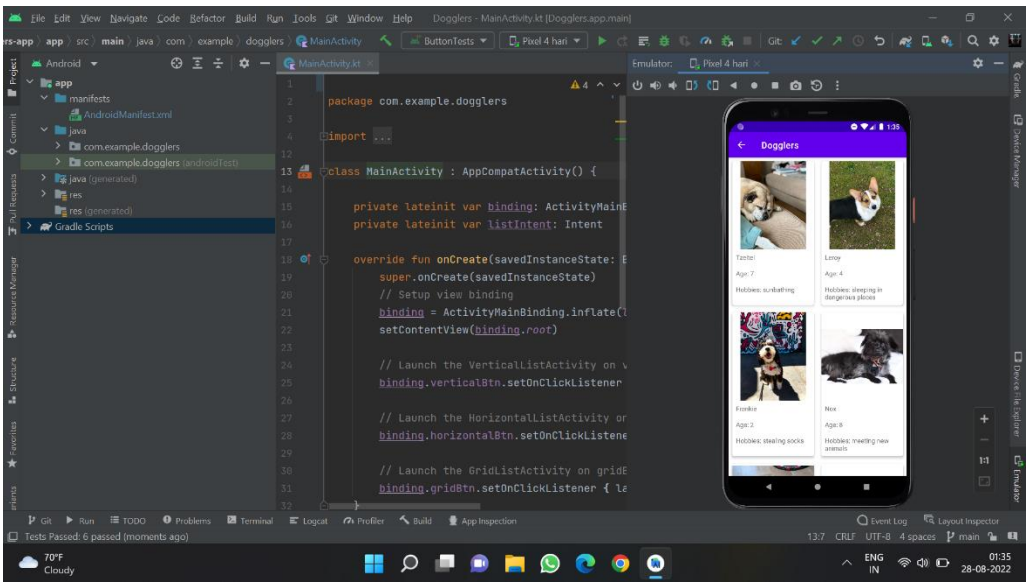


Google developer

2. Amphibians app:



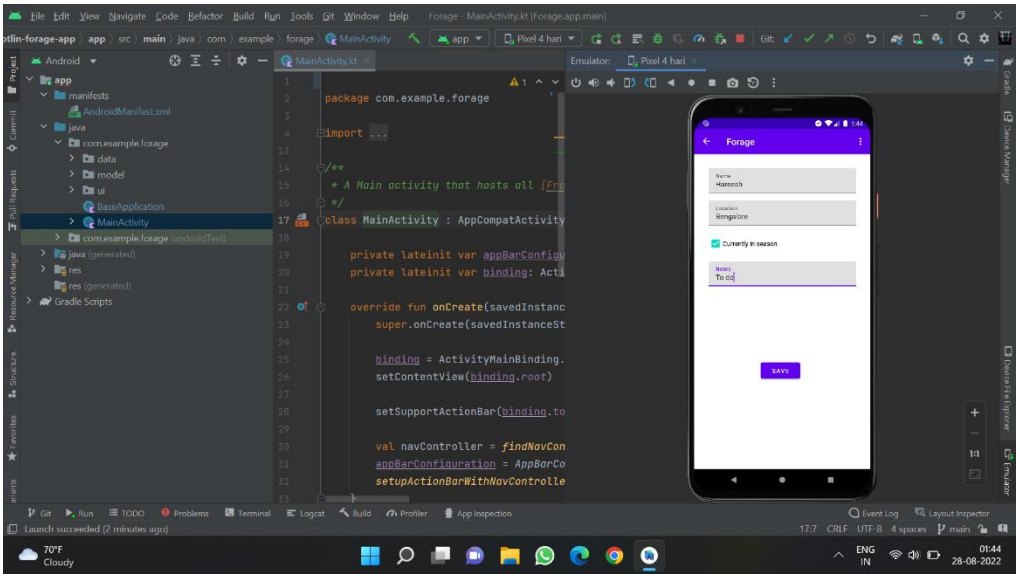
3. **Dogglers app:**



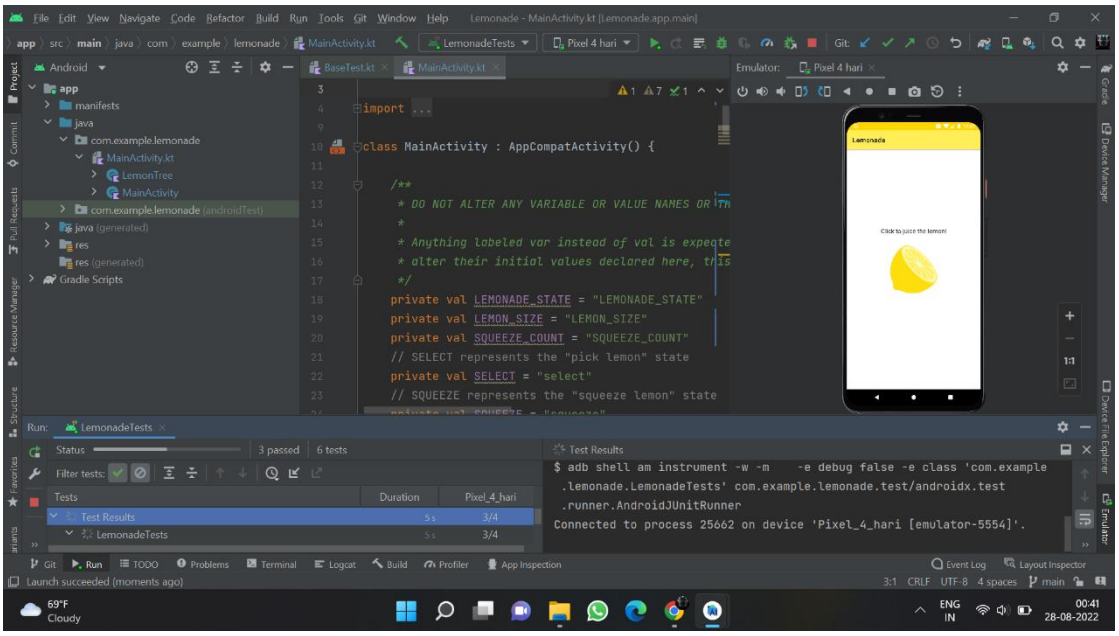
Google developer

Smart Internz

4. Forage app:

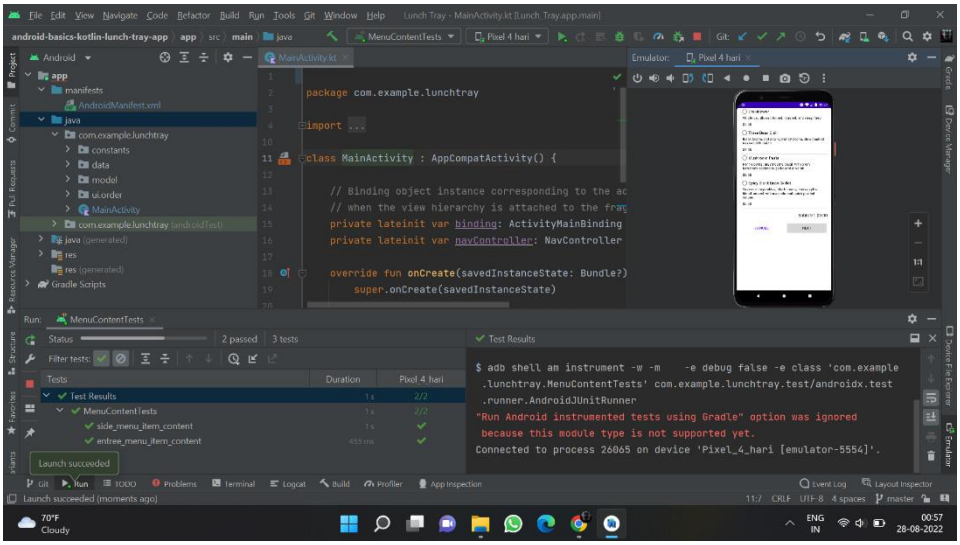


5. Lemonade app:



Google developer

6. Lunch_Tray_app:



Then they assigned the project which is I have to do. The given project name is Grocery app.

Project: Grocery Application Report

This is the simple android application where we shop our groceries. And here we used the MVVM and room database to store the added groceries.

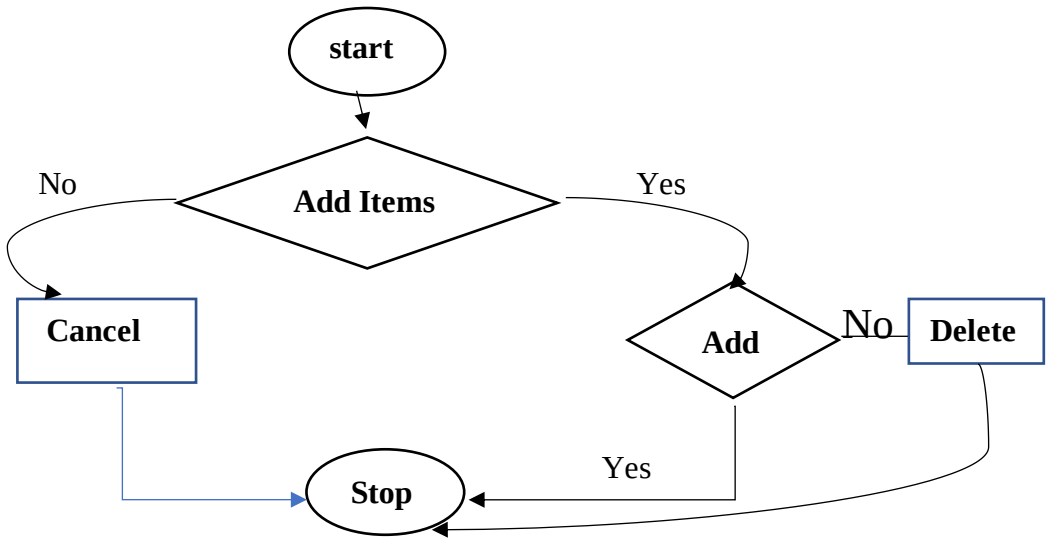
Application Features:

1.Adding different grocery items

2.Delete the grocery items

Google developer

Flowchart:



About:

Once we created the new project we have two files, they are activity_main file and MainActivity.kt file. After this we need to add the dependences to gradle file. Then we have to write the code to create our Application as given in video. We need to create some classes which is helps to add the items and delete the items.

Here we given more values they are Quality of the items, Name of the items and item price. And there is methods also created to communicate with each other files.

We can make the design pattern by using layout design. Once I created the all classed that is next thing is to run the application.

I done complete project which is assigned me and given all document to the mentor. Now I jest To know that all the things are good or not. The some of the output shots are given below.

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.

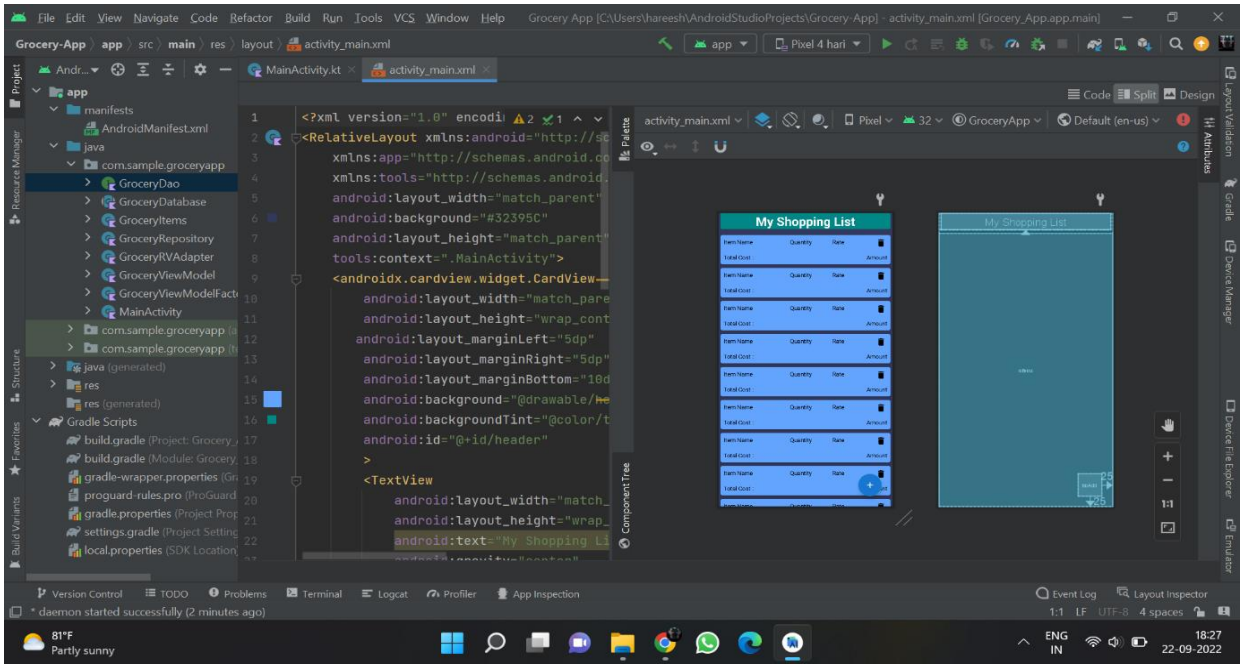
DAO: Data access object. A mapping of SQL queries to functions. When you use a DAO, you call the methods, and Room takes care of the rest.

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.

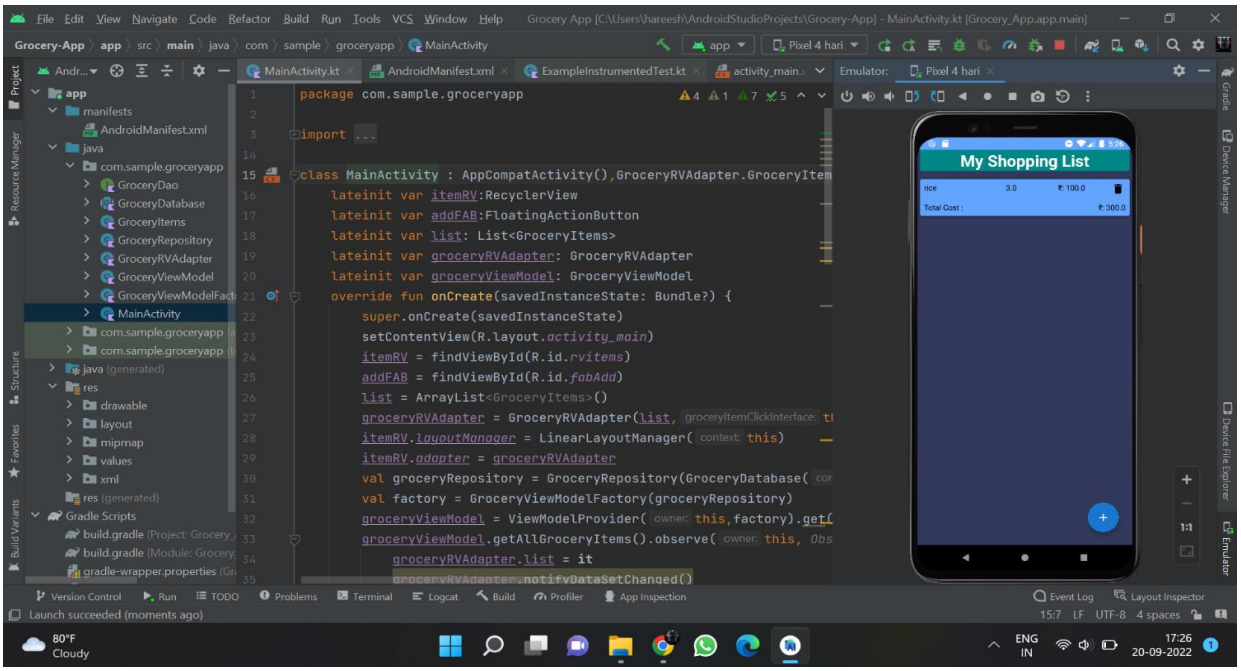
Google developer

Smart Internz

The Interface :

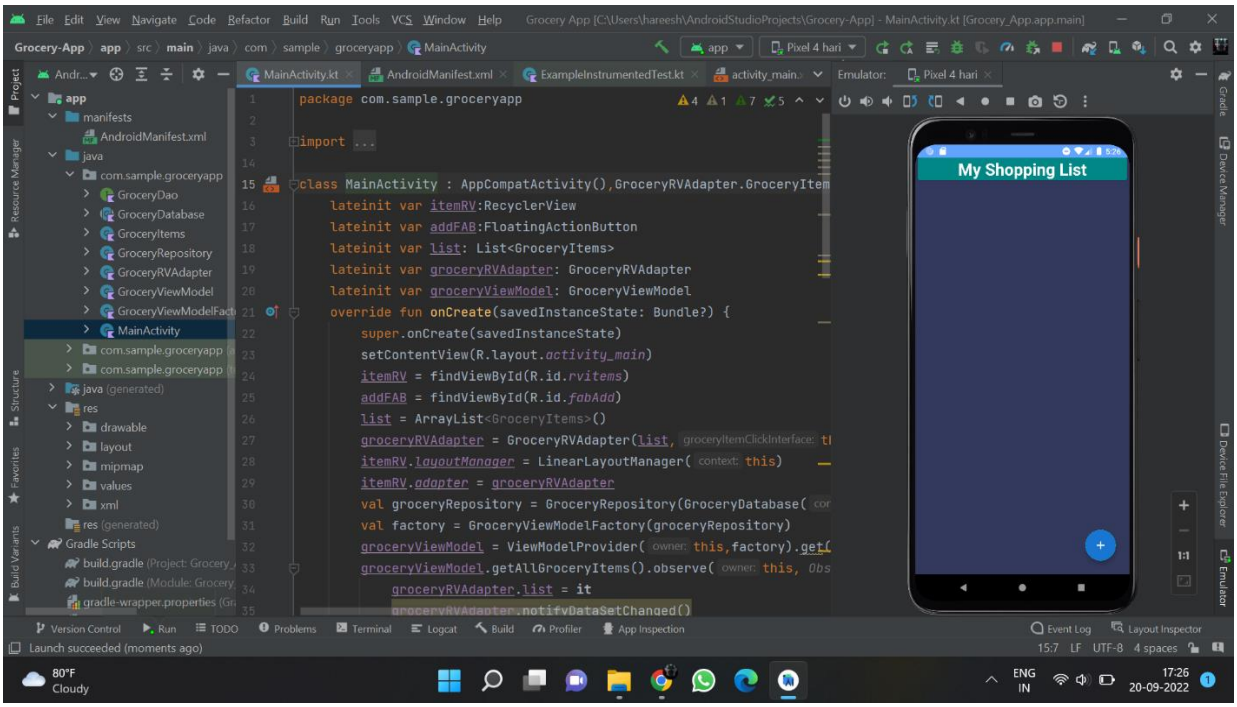


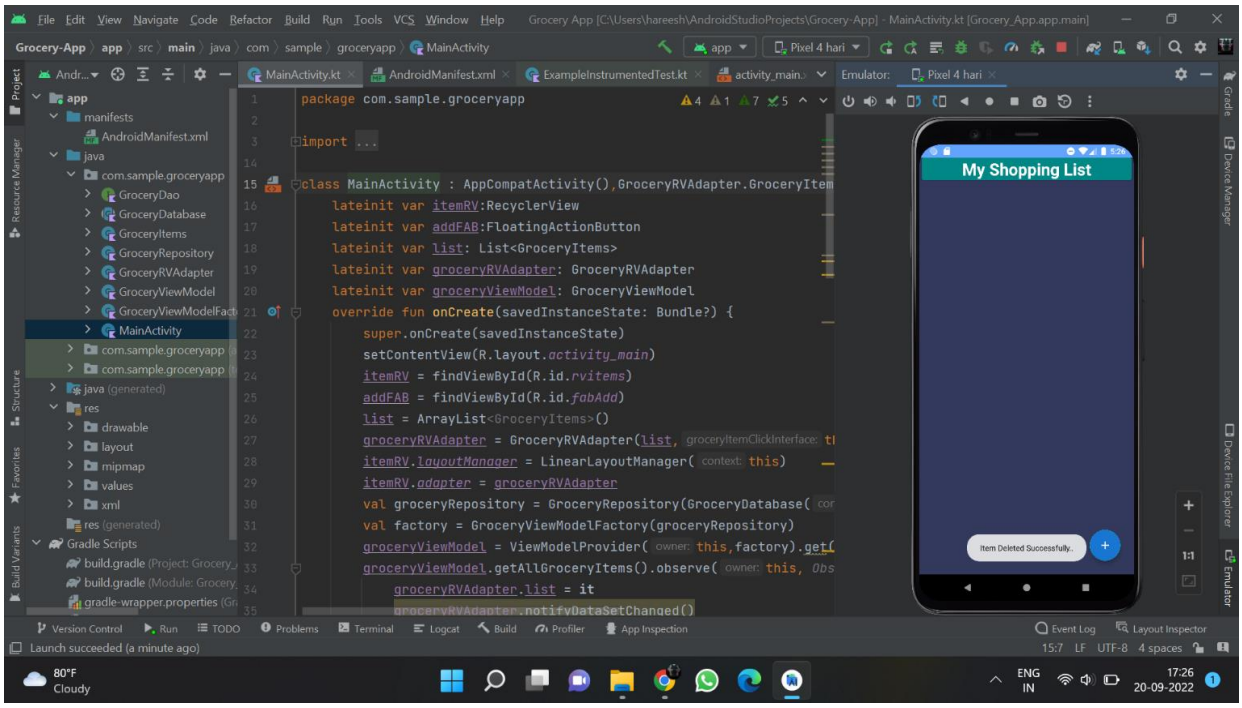
Result with Code:



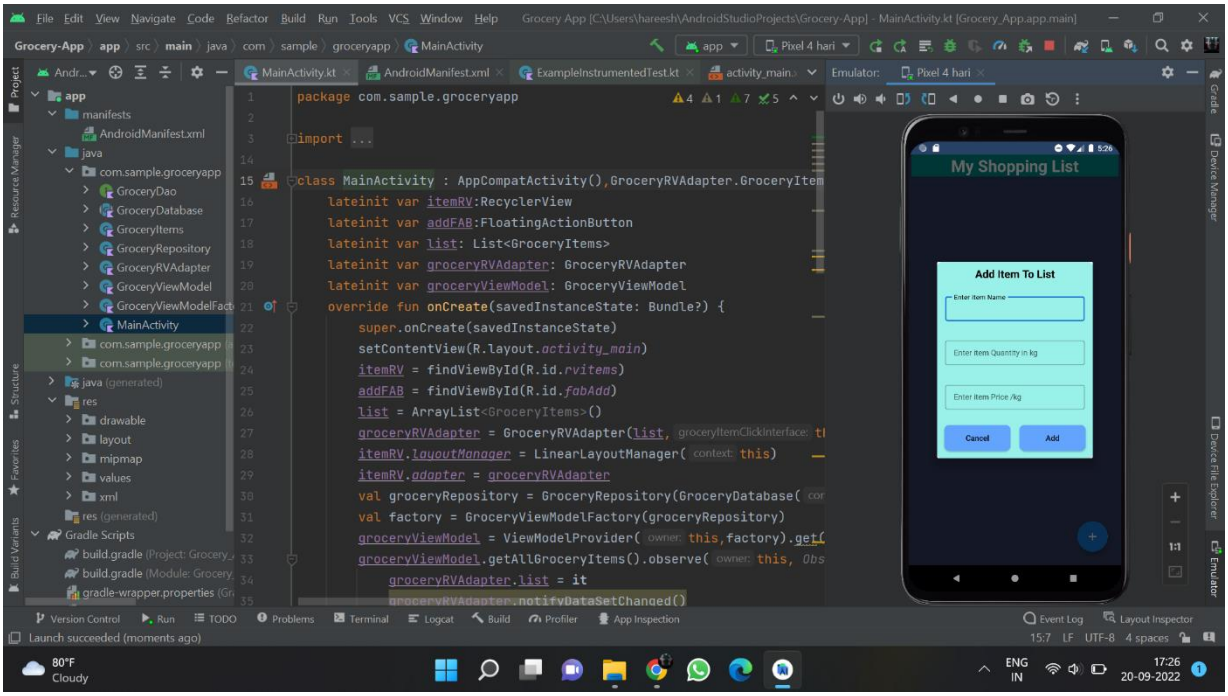
Google developer

Smart Internz





Google developer



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 forget 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.

Google developer

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.

Github link: https://github.com/smartinternz02/SI-GuidedProject-57409-1661140305

Demo Video: https://www.youtube.com/watch?v=6XgcWVoMZFQ

Developer Link: https://g.dev/hareeshnaik061

Google developer