Name : Kavya Para

Student ID : 16326415

Mail : kpkc8@umkc.edu

GitHub Link : https://github.com/KavyaPara/Web_Development_Course/tree/main/Mobile-Development/ICP10

Name : Manikanta Kavitapu

Student ID : 16322502

Mail : mkmdy@umkc.edu

Github Link : https://github.com/Manikantakavitapu/Web-Development-Course/tree/main/Mobile-Development/ICP10

Mobile App Development

Tasks:

Retrofit:

- Retrofit is basically is a type-safe HTTP client which is used for Java and Android. It will turn our HTTP API into a Java interface. It will allow us to retrieve and upload JSON using a REST based. We can configure which converters are used for the data serialization(the process of converting an object into stream of bytes). Retrofit also manages the process of creating, sending and receiving the HTTP requests & responses. When a connection to a web service fails, it switches IP addresses. To avoid sending duplicate requests, it caches responses. It has underlying GSON converter that can naturally parse HTTP response into an Object or some other sorts in Java that can be utilized in your code.

```
Retrofit retrofit = new Retrofit.Builder()
        .baseUrl("https://api.github.com/")
        .addConverterFactory(GsonConverterFactory.create())
        .build();
```

- This retrofit class generates the implementation of the interface and each call created from this interface can make HTTP requests to the web server.

```
ApiCollections apiCollections = retrofit.create(ApiCollections.class);

Call<List<User>> usersCall = apiCollections.getData();
```

- We have to add a dependency in build.gradle(implementation 'com.squareup.retrofit2:converter-gson:2.9.0') for using Retrofit.

```
implementation 'com.squareup.retrofit2:retrofit:2.9.0'
implementation 'com.squareup.retrofit2:converter-gson:2.9.0'
```

- As we are fetching the user details from the web, which makes our application to use the internet we have to add an extra permission in our AndroidManifest.xml file. If we don't add this permission in the xml file, we will not be able to use the internet.

```
<uses-permission android:name="android.permission.INTERNET"/>
```

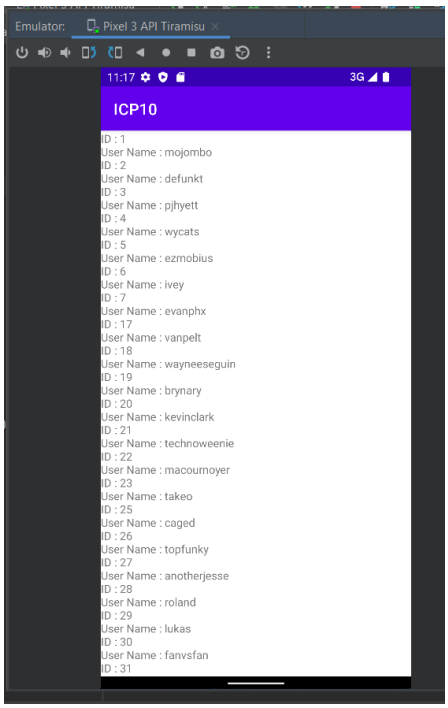
- We have used an annotation named @SerializedName for fetching the username of the users. This annotation is used to serialize a field with an alternate name rather than its field name.

```
@SerializedName("login")
private String userName;
```

- We have used a scrollview which will help us to scroll the screen to see the user details.

Output:

The results of fetching the user details are shown below:



Contribution:

We both have contributed equally.