ICP 10

Name: Tej Deep Parvatha Reddy

Email: tpgkd@umsystem.edu

GitHub Link:

https://github.com/TejdeepP/WebProgrammingSpring2022/tree/main/MobilePart/ICP10

Partner Name: Yaswanth Paruchuri

Email: ypfm7@umsystem.edu

GitHub Link: https://github.com/Yaswanthypfm7/WebDevCourse/tree/main/MobilePart/ICP10

In this ICP we created a android application where users will be able to see the list of all the users Log ID and also the Login or User name.

Retrofit is a type-safe REST Client for android. It provides a powerful framework for interacting and authenticating with any APIs and sending network requests.

First we added the permission to access the internet in the Manifest file by giving the highlighted command

```xml
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.icp10">
    <uses-permission android:name="android.permission.INTERNET"/>
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="ICP10"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.ICP10">
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNC
            </intent-filter>
        </activity>
    </application>

</manifest>
```
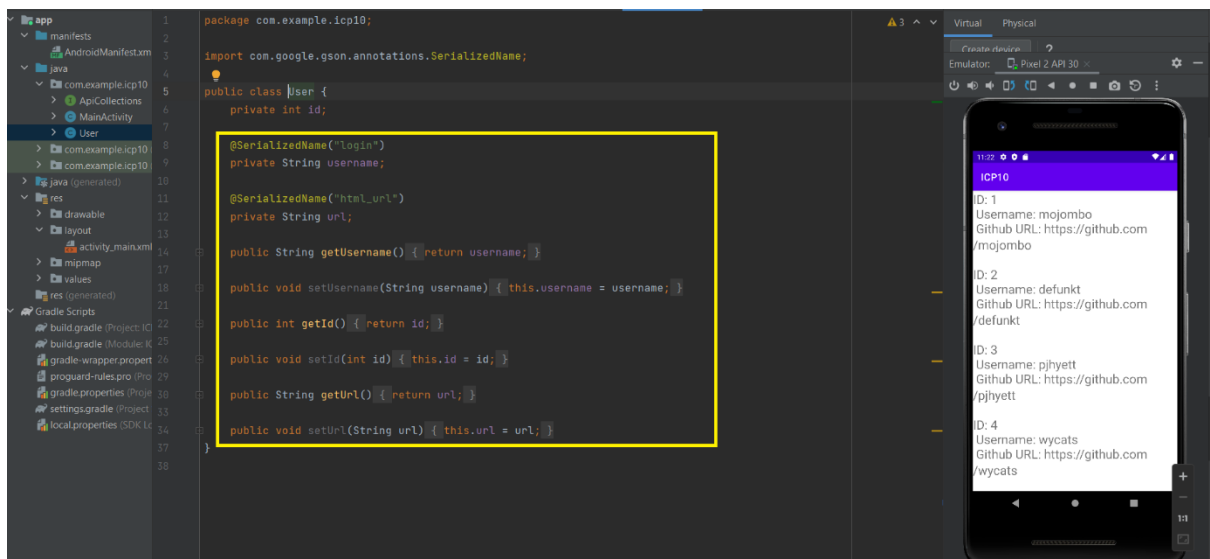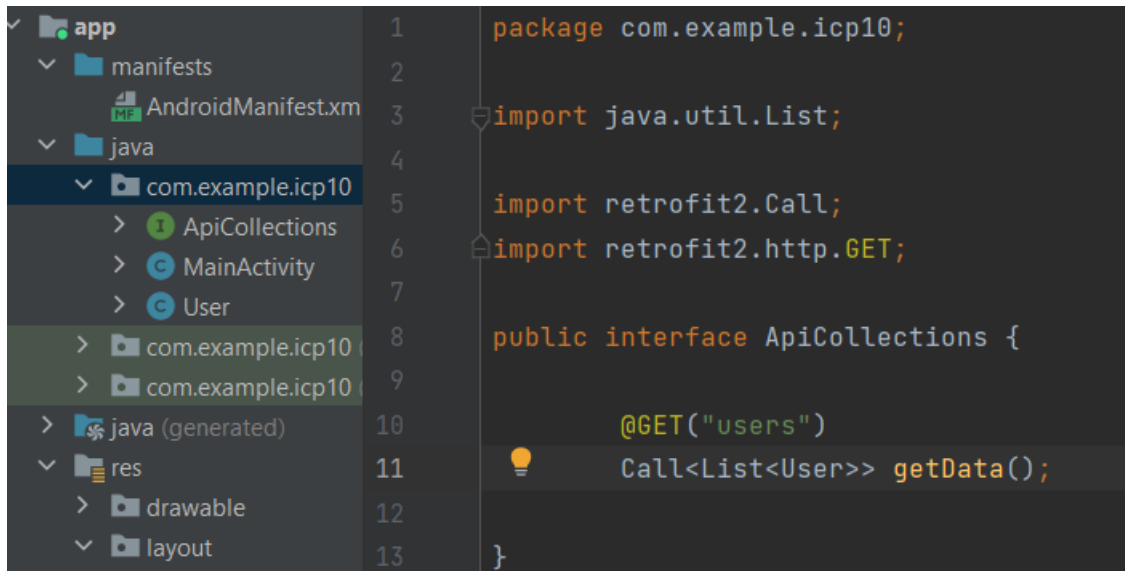
Add the squareup-retrofit and the retrofit converter-gson in the dependencies and then sync them with the project.
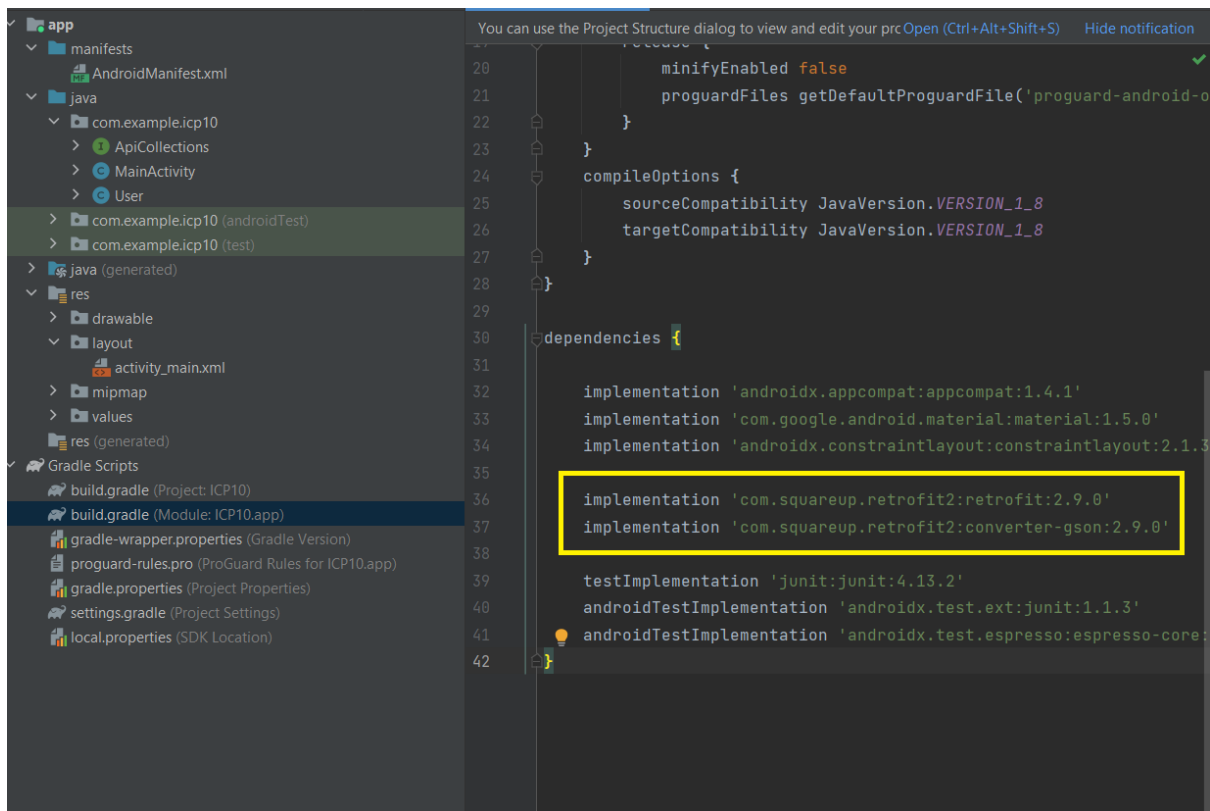

Create a new java class as User
and added the fields that we want to show to the user. In this case we took the ID, User name, GithubURL. We write the getter and setter methods to get the details.

We created a java interface for the collection of the data we are fetching using the API as a List and store the data in to the users by using the getData method.


```java
package com.example.icp10;

import java.util.List;

import retrofit2.Call;
import retrofit2.http.GET;


public interface ApiCollections {

    @GET("users")
    Call<List<User>> getData();

}
```

Coming to the MainActivity, we created the logic for the function to work using the retrofit call back first we initialized a retrofit builder and then assigned it to a variable. Later the retrofit is created using the apiCollections interface that we created earlier, and we get the data and store in the userCall. And from the userCall enqueue we create the call back of the Use list and take the response we created two methods here one is for a successful response and the other is in case of any error. In the successful response we add the data that we get from the list using the for loop and add it to the text view in the Layout form to display to the user.
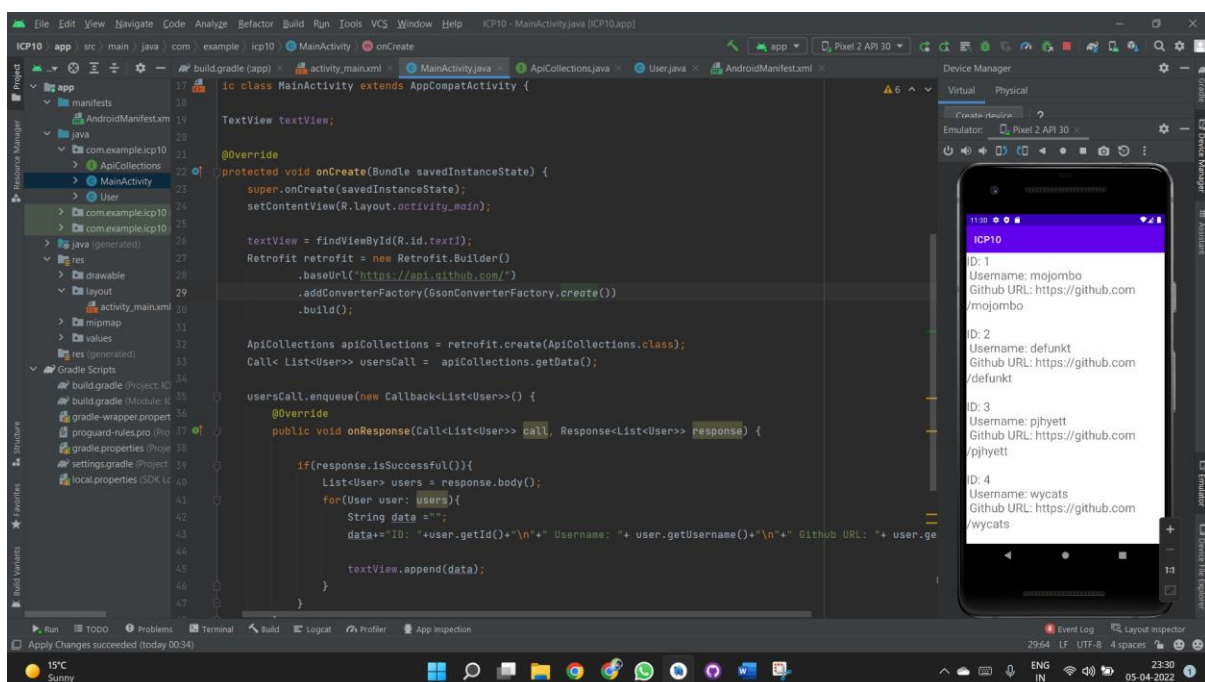
Layout of the application is simple as we just added a textview to show that data that is passed from the Mainactivity using the id text1. In order to get the app to scroll up and down when we have the huge data we added the nestedscrollView widget as shown below.
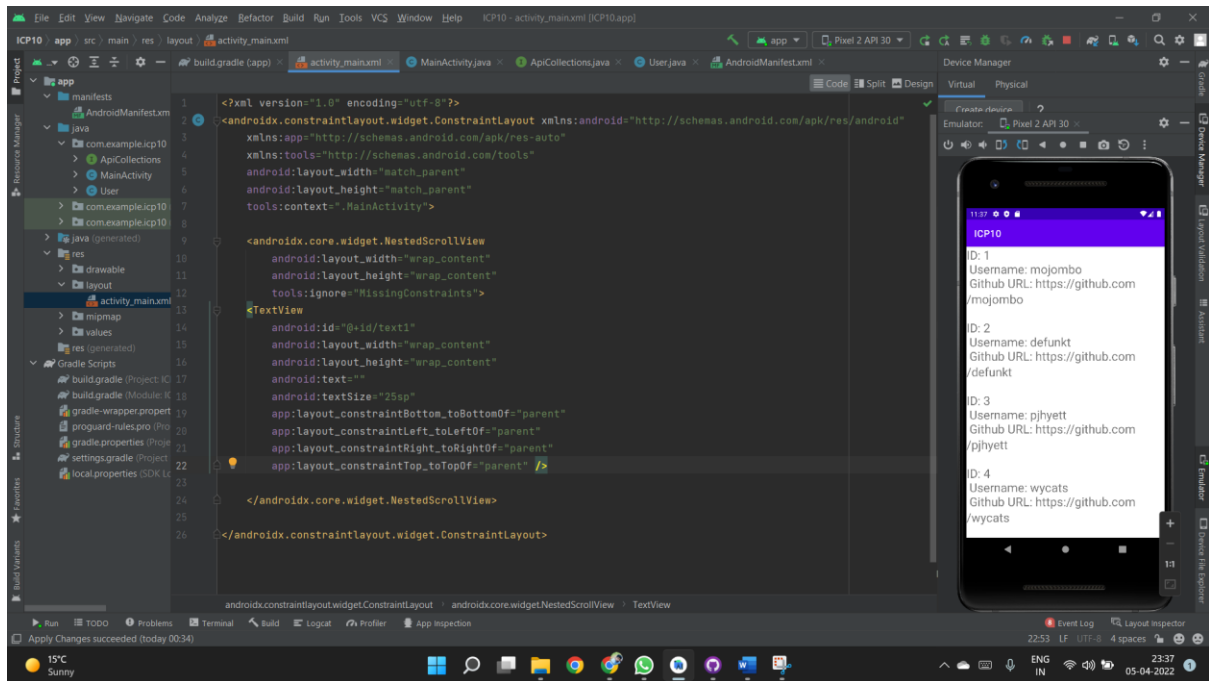
In the final output when we open the app we will be able to see the User ID, UserName, GithubURL of all the Github users using the API.
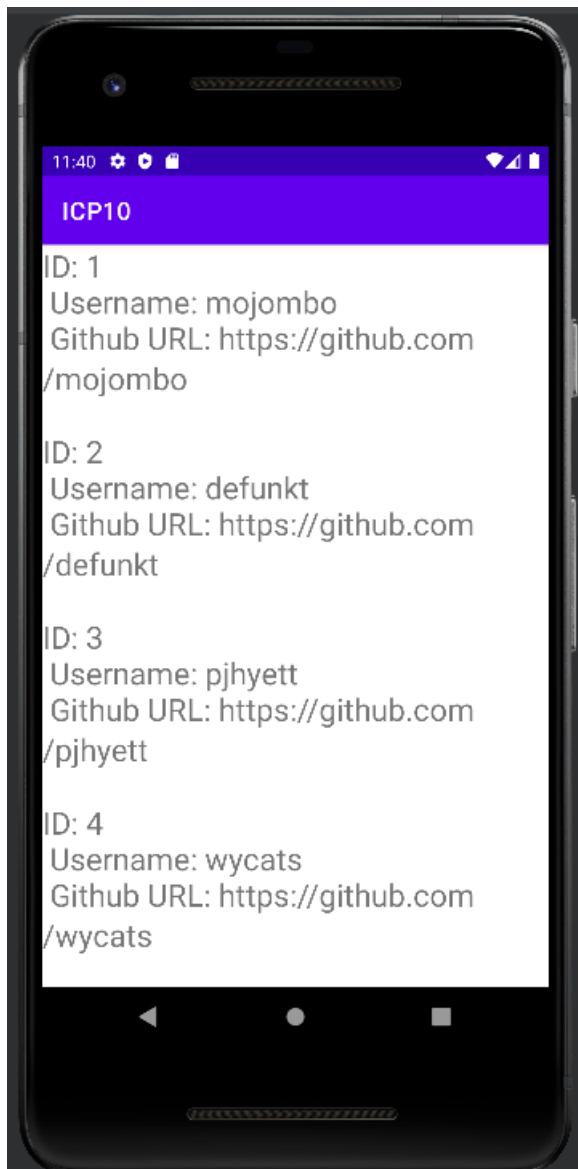
Contribution:

We have contributed equally.