

CMPS 312 Mobile Application Development

LAB 5: UI 2

WORKING WITH LISTS

[ANDROID LIST VIEW AND RECYCLER VIEW]

OBJECTIVE

Familiarize yourself how to implement the Android Scrolling List Views using **Listview** and **RecyclerView** to display a large amount of similar items on screen. By the end of the lab you should be able to create

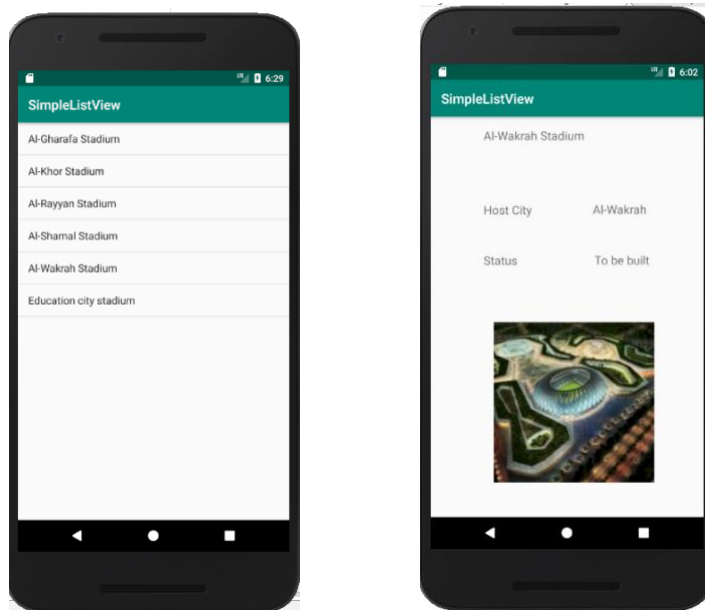
1. Simple listviews with single items in a row
2. Custom listviews with multiple items in a row
3. RecyclerViews with different layouts
 - a. Linear
 - b. Grid
 - c. StaggeredGrid

QATAR2022 STADIUM INFO APPLICATION

Download the Assets folder in github and put it somewhere you can locate in your computer. We will be using this folder to accomplish the application shown below.

PART A – SIMPLE LISTVIEWS

In this exercise, we will create an application that has two activities; the first activity contains a list view. When user selects an item, the application moves to the second activity, which displays more details about the selected stadium.

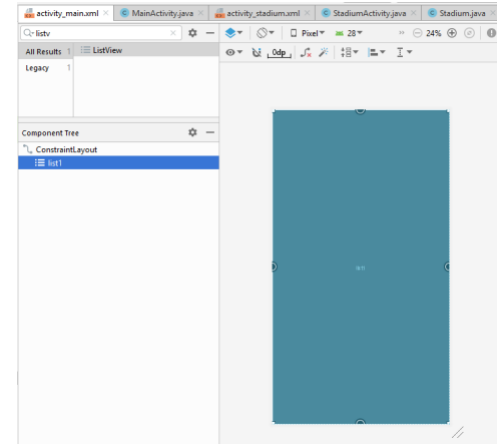


STEPS:

- 1- Create a new project.
- 2- Copy the stadium images to the drawable folder.
- 3- Add ListView to the main layout, call it **list**
- 4- Create a new class call it **"Stadium"** that will have the following attributes

```
private String name;  
private String hostCity;  
private String status;  
private int stadiumImage;
```

- 5- Create a constructor, setter/getter methods.
 - Add the following method that creates a list of objects of Stadium:
 - Link for the stadiums <http://www.qatar.to/stadiums/World-Cup-2022-stadiums.php>

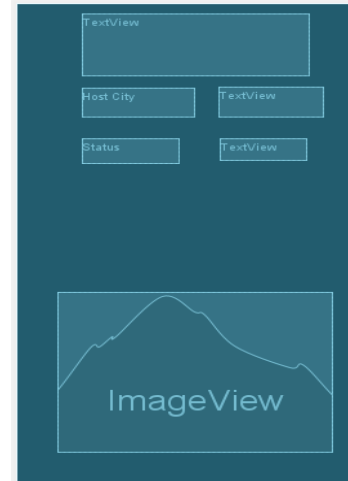


```
public static ArrayList<Stadium> getStadiumsData(){  
    ArrayList<Stadium> stadiums = new ArrayList<>();  
  
    stadiums.add(new Stadium("Al-Gharafa Stadium" , "Al-Rayyan" , "Major renovation" ,  
R.drawable.algharafa));  
    stadiums.add(new Stadium("Al-Khor Stadium" , "Al-Khor " , "To be built" ,  
R.drawable.alkhor));  
    stadiums.add(new Stadium("Al-Rayyan Stadium" , "Al-Rayyan" , "Major renovation" ,  
R.drawable.alrayan));  
    stadiums.add(new Stadium("Al-Shamal Stadium" , "Al-Shamal" , "To be built" ,  
R.drawable.alshamal));  
    stadiums.add(new Stadium("Al-Wakrah Stadium" , "Al-Wakrah" , "To be  
built",R.drawable.alwakara));  
    stadiums.add(new Stadium("Education city stadium" , " Al-Rayyan" , "To be  
buil",R.drawable.educationcity));  
  
    return stadiums;  
}
```

Add the following method that returns a list contains names of the stadiums:

```
public static ArrayList<String> getStadiumNames()  
{  
    ArrayList<String> s=new ArrayList<>();  
    for (Stadium s:getStadiumsData())  
        s.add(s.getName());  
    return s;  
}
```

- 6- Create the second activity “**StadiumDetailsActivity**” that will display the details of the selected stadium.
The layout of this activity will look like the following



- 7- The second activit “**StadiumDetailsActivity**” will receive an integer from the first activity indicates the selected stadium, it will use that to retrieve the data of that stadium using method “**getStadiumData**” in the

```
public class StadiumActivity extends AppCompatActivity {
    TextView name,hostcity, status;
    ImageView img;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_stadium);
        name=findViewById(R.id.textView);
        hostcity=findViewById(R.id.textView3);
        status=findViewById(R.id.textView5);
        img=findViewById(R.id.imageView);

        int i=getIntent().getIntExtra( name: "selected", defaultValue: 0 );
        ArrayList<Stadium> staduims=Stadium.getStadiumsData();
        name.setText(staduims.get(i).getName());
        hostcity.setText(staduims.get(i).getHostCity());
        status.setText(staduims.get(i).getStatus());
        img.setImageResource(staduims.get(i).getStadiumImage());
    }
}
```

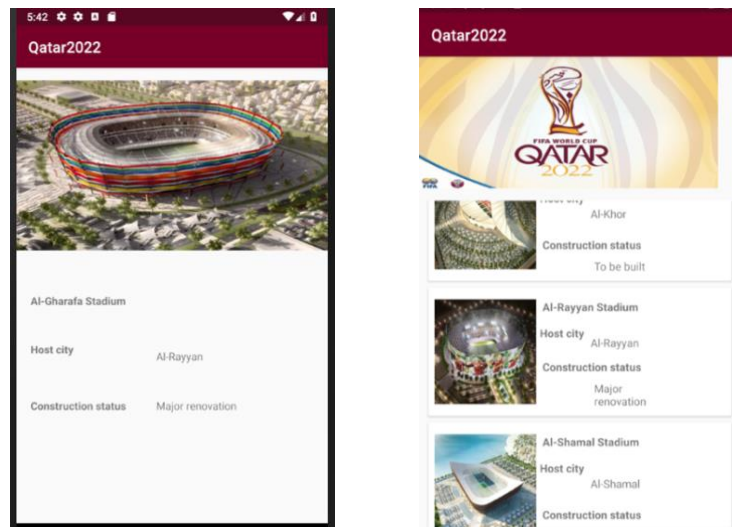
class “Stadium”.

- 8- The main activity will display the **ListView** and handle the event of selecting an item in the list. As follows:
- a- You need to create a reference for the ListView
`ListView list=findViewById(R.id.list);`
 - b- Create an adapter to link the listview with the data
`ArrayAdapter<String> adapter=new ArrayAdapter<>(this,
android.R.layout.simple_list_item_1,Stadium.getStadiumNames());
list.setAdapter(adapter);`
 - c- To handle the event

```
list.setOnItemClickListener(new AdapterView.OnItemClickListener() {
    @Override
    public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
        Intent intent=new Intent(getApplicationContext(), StadiumActivity.class);
        intent.putExtra("selected",position);
        startActivity(intent);
    }
});
}
```

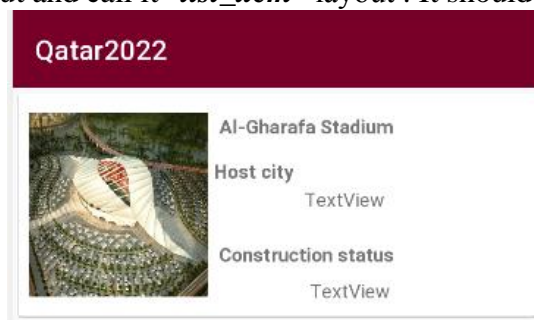
PART B-CUSTOM LISTVIEW

Issue: In this first part we created a listview with a single textview. But what if we want to display an image and a text in the list. In this case we will not be able to use the arrayAdapter. So, we will be required to create a custom adapter.



STEPS:

- 1- Create an application with a name “Qatar2022CustomListViewImplementation”
- 2- Repeat steps 2 to 7
- 3- Make the stadium class implement the **Parcelable** Interface
- 4- Create a layout and call it “**list_item**” layout . It should look something like this



- 5- Create a **custom Adapter** and name it **MyCustomAdapter**.
- 6- The adapter should allow you to render the stadium **image**, **host city** , **status** and **stadium** as shown below.



- 7- Also Make sure your displaying all the information in a cardview.
- 8- Add a button that allows you to add new ListItem.
- 9- Once you add the list item make sure you notify the changes.
- 10- Improve your listview by implementing the ViewHolder pattern.

PART-C RECYCLERVIEW

1. Create an application with a name “**Qatar2022RecyclerViewImplementation**”
2. Open **build.gradle** and add recycler view dependency and cardview dependency

```
implementation 'com.android.support:cardview-v7:28.0.0'
implementation 'com.android.support:recyclerview-v7:28.0.0'
```

3. In the **activity_main.xml** add a recyclerView widget.
4. Copy the **list_item.xml** file you created in PART B to your project
5. Also copy the Stadium class from PART B into your project
6. Create the Adapter Class for the recyclerView. Call the adapter **StadiumAdapter**
In the MainActivity reference your recyclerview
7. Add the adapter
8. Add the RecyclerView Layout. Use Linear / Grid Layout
9. Add a click listener to your List Items . Once the item is clicked open a second activity that displays the following information

