****

**Green University of Bangladesh**

**Department of Computer Science and Engineering (CSE)**

**Faculty of Sciences and Engineering**

**Semester: (Fall, Year:2024, B.Sc. in CSE (Day)**

**Lab Report NO #05**

**Course Title: Mobile Application Development**

**Course Code: CSE-402 Section: 213 D-3**

**Lab Experiment Name: List where we can add both image and name of that image in the same row.**

**Student Details**

|  |  |  |
| --- | --- | --- |
| **Name** | | **ID** |
| **1.** | Nadib Rana | 213002247 |

**Lab Date : 05.10.2024**

**Submission Date : 12.10.2024**

**Course Teacher’s Name** **: Md. Jahid Tanvir**

|  |
| --- |
| **Lab Report Status**  **Marks: ………………………………… Signature:.....................**  **Comments:.............................................. Date:..............................** |

**1.TITLE OF THE LAB REPORT EXPERIMENT**

Implement a list where you can add both image and name of that image in the same row.

**2. OBJECTIVES**

-To design and implement a ListView for displaying items in a vertically scrollable format.

-To customize ListView rows to integrate both an image and its associated text.

**3. PROCEDURE**

The Android ListView is a user interface element that displays a series of items in a scrollable vertical layout.Customizing the rows allows each entry to present an image along with its related text. The steps are:

1. Designing a unique XML layout for the rows.
2. Utilizing an adapter to link the data with the ListView.
3. Filling the ListView with data retrieved from an array or other source.

**4. IMPLEMENTATION:**

**Login page: list\_item.xml**

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:orientation="horizontal"

android:padding="8dp">

<ImageView

android:id="@+id/itemImage"

android:layout\_width="50dp"

android:layout\_height="50dp"

android:layout\_marginRight="8dp"

android:src="@drawable/ic\_launcher\_background" />

<TextView

android:id="@+id/itemName"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Sample Item"

android:textSize="18sp" />

</LinearLayout>

**activity\_main.xml**

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical">

<ListView

android:id="@+id/listView"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent" />

</LinearLayout>

**CustomAdapter.java:**

package com.example.listview;

import android.content.Context;

import android.view.LayoutInflater;

import android.view.View;

import android.view.ViewGroup;

import android.widget.BaseAdapter;

import android.widget.ImageView;

import android.widget.TextView;

public class CustomAdapter extends BaseAdapter {

private Context context;

private String[] names;

private int[] images;

public CustomAdapter(Context context, String[] names, int[] images) {

this.context = context;

this.names = names;

this.images = images;

}

@Override

public int getCount() {

return names.length;

}

@Override

public Object getItem(int position) {

return names[position];

}

@Override

public long getItemId(int position) {

return position;

}

@Override

public View getView(int position, View convertView, ViewGroup parent) {

if (convertView == null) {

convertView = LayoutInflater.from(context).inflate(R.layout.list\_item, parent, false);

}

ImageView imageView = convertView.findViewById(R.id.itemImage);

TextView textView = convertView.findViewById(R.id.itemName);

imageView.setImageResource(images[position]);

textView.setText(names[position]);

return convertView;

}

}

**MainActivity.java:**

package com.example.listview;

import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

import android.widget.ListView;

public class MainActivity extends AppCompatActivity {

ListView listView;

String[] names = {"Item 1", "Item 2", "Item 3"};

int[] images = {R.drawable.image1, R.drawable.image2, R.drawable.image3};

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

listView = findViewById(R.id.listView);

CustomAdapter adapter = new CustomAdapter(this, names, images);

listView.setAdapter(adapter);

}

}

**OUTPUT:**

****

**5. ANALYSIS AND DISCUSSION**

What Went Well

Successfully implemented the ListView with custom-designed rows that displayed both images and their corresponding names.

Achieved smooth integration between the adapter and the ListView for data binding.

**Challenges**

Encountered difficulties in scaling images proportionally to fit within the custom row layout.

Faced performance issues when handling a large dataset, requiring optimization to maintain smooth scrolling.

**Learning Outcomes**

Acquired a deeper understanding of creating and using custom adapters in Android.

Enhanced proficiency in managing dynamic UI components and optimizing performance for ListView.