



**BATCH AND ROLL NO:** R08 42435

**EXPERIMENT NO.2**

**TITLE:** Design a mobile application to create home page using grid layout.

**DATE OF PERFORMANCE:**

**DATE OF SUBMISSION:**

**Title:** Designing of mobile application to create home page using grid layout.

**Requirements:**

1. Android studio
2. Java SDK

**Theory:**

**Introduction**

In the realm of mobile application development, the design and layout of user interfaces play a pivotal role in creating a seamless and visually appealing user experience. The choice of layout managers is crucial for efficiently organizing and presenting content on the screen. One popular layout manager for achieving a structured and responsive layout is the Grid Layout.

**Grid Layout:** Grid Layout is a versatile layout manager that arranges UI components in a grid structure. This layout is particularly useful for creating home pages and dashboards in mobile applications, allowing developers to organize content in rows and columns. It provides a flexible and dynamic structure that adapts well to various screen sizes and orientations.

**Objective of the Lab:** The primary objective of this lab is to guide you through the process of designing a home page for a mobile application using the Grid Layout. You will learn how to efficiently organize and display content, such as images, text, and interactive elements, in a grid format. By the end of this lab, you should be adept at using the Grid Layout to create visually appealing and responsive home pages for your mobile applications.

**Lab Prerequisites:**

- Basic understanding of mobile application development concepts.
- Familiarity with the chosen development environment (e.g., Android Studio).
- Prior knowledge of programming languages such as Java (for Android)



**Steps:**

**Designing a Home Page Using Grid Layout:**

**Step 1: Set Up Your Development Environment**

- Ensure you have a suitable development environment installed, such as Android Studio for Android development.
- Create a new project or open an existing one.

**Step 2: Understand Grid Layout Basics**

- Familiarize yourself with the basic concepts of the Grid Layout, including rows, columns, and grid items.
- Explore how the Grid Layout adapts to different screen sizes and orientations.

**Step 3: Create Grid Layout in XML**

- Open the XML layout file (for Android).
- Define a Grid Layout container with a specified number of rows and columns.

**Step 4: Add UI Elements as Grid Items**

- Identify the content you want to display on the home page.
- Add UI elements (e.g., ImageView, TextView, Button) as grid items within the rows and columns of the Grid Layout.

**Step 5: Customize Grid Items**

- Customize the appearance of each grid item by adjusting properties such as size, padding, and margins.
- Consider using features like span to merge multiple rows or columns for specific elements.

**Step 6: Handle Interactions and Navigation**

- If applicable, implement interaction elements such as buttons or clickable components.
- Set up navigation or actions for grid items, allowing users to navigate to other pages or perform specific tasks.

**Step 7: Test Responsiveness**

- Test your home page layout on various devices and screen sizes to ensure responsiveness.
- Adjust layout parameters as needed to optimize the appearance on different devices.



### **Step 8: Implement Dynamic Data**

- If your home page involves displaying dynamic content (e.g., images from a server, user-specific information), implement the necessary logic to fetch and populate the data within the Grid Layout.

### **Step 9: Test and Debug**

- Test your home page thoroughly, including user interactions and data retrieval.
- Use debugging tools to identify and address any issues that may arise during testing.

### **Step 10: Iterate and Enhance**

- Gather user feedback and iterate on the design based on usability and user experience.
- Consider enhancing the home page with animations, transitions, or additional features to make it more engaging.

### **XML Code:**

#### **AndroidManifest.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/Theme.MaterialComponents.Light.DarkActionBar">
        <activity
            android:name=".Expt02_42441"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```



**activity\_main.xml:**

```
<?xml version="1.0" encoding="utf-8"?>
<GridLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:columnCount="2"
    android:rowCount="2"
    android:padding="12dp"
    android:background="#ECEFF1"
    android:layout_marginTop="?attr/actionBarSize"
    android:alignmentMode="alignMargins"
    android:useDefaultMargins="true">

    <!-- Home Card -->
    <androidx.cardview.widget.CardView
        android:id="@+id/homeCard"
        android:layout_width="0dp"
        android:layout_height="0dp"
        android:layout_rowWeight="1"
        android:layout_columnWeight="1"
        android:layout_margin="8dp"
        app:cardCornerRadius="15dp"
        app:cardElevation="8dp">

        <GridLayout
            android:layout_width="match_parent"
            android:layout_height="match_parent"
            android:rowCount="3"
            android:columnCount="1"
            android:padding="16dp"
            android:background="@drawable/gradient_blue">

            <ImageView
                android:layout_width="64dp"
                android:layout_height="64dp"
                android:layout_row="0"
                android:layout_column="0"
                android:layout_gravity="center"
                android:src="@drawable/ic_home"
                android:tint="#FFFFFF"/>
```



```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_row="1"
    android:layout_column="0"
    android:layout_gravity="center"
    android:layout_marginTop="8dp"
    android:text="Home"
    android:textSize="20sp"
    android:textStyle="bold"
    android:textColor="#FFFFFF"/>

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_row="2"
    android:layout_column="0"
    android:layout_gravity="center"
    android:text="Welcome Back!"
    android:textColor="#E0FFFFFF"
    android:textSize="14sp"/>
</GridLayout>
</androidx.cardview.widget.CardView>

<!-- Settings Card -->
<androidx.cardview.widget.CardView
    android:id="@+id/settingsCard"
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:layout_rowWeight="1"
    android:layout_columnWeight="1"
    android:layout_margin="8dp"
    app:cardCornerRadius="15dp"
    app:cardElevation="8dp">

<GridLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:rowCount="3"
    android:columnCount="1"
    android:padding="16dp"
    android:background="@drawable/gradient_purple">
```



```
<ImageView
    android:layout_width="64dp"
    android:layout_height="64dp"
    android:layout_row="0"
    android:layout_column="0"
    android:layout_gravity="center"
    android:src="@drawable/ic_settings"
    android:tint="#FFFFFF"/>
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_row="1"
    android:layout_column="0"
    android:layout_gravity="center"
    android:layout_marginTop="8dp"
    android:text="Settings"
    android:textSize="20sp"
    android:textStyle="bold"
    android:textColor="#FFFFFF"/>
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_row="2"
    android:layout_column="0"
    android:layout_gravity="center"
    android:text="Customize App"
    android:textColor="#E0FFFFFF"
    android:textSize="14sp"/>
```

```
</GridLayout>
```

```
</androidx.cardview.widget.CardView>
```

```
<!-- Profile Card -->
```

```
<androidx.cardview.widget.CardView
    android:id="@+id/profileCard"
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:layout_rowWeight="1"
    android:layout_columnWeight="1"
    android:layout_margin="8dp"
    app:cardCornerRadius="15dp"
    app:cardElevation="8dp">
```



```
<GridLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:rowCount="3"
    android:columnCount="1"
    android:padding="16dp"
    android:background="@drawable/gradient_orange">
```

```
<ImageView
    android:layout_width="64dp"
    android:layout_height="64dp"
    android:layout_row="0"
    android:layout_column="0"
    android:layout_gravity="center"
    android:src="@drawable/ic_profile"
    android:tint="#FFFFFF"/>
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_row="1"
    android:layout_column="0"
    android:layout_gravity="center"
    android:layout_marginTop="8dp"
    android:text="Profile"
    android:textSize="20sp"
    android:textStyle="bold"
    android:textColor="#FFFFFF"/>
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_row="2"
    android:layout_column="0"
    android:layout_gravity="center"
    android:text="View Profile"
    android:textColor="#E0FFFFFF"
    android:textSize="14sp"/>
```

```
</GridLayout>
```

```
</androidx.cardview.widget.CardView>
```



<!-- Notifications Card -->

```
<androidx.cardview.widget.CardView
    android:id="@+id/notificationsCard"
    android:layout_width="0dp"
    android:layout_height="0dp"
    android:layout_rowWeight="1"
    android:layout_columnWeight="1"
    android:layout_margin="8dp"
    app:cardCornerRadius="15dp"
    app:cardElevation="8dp">
```

<GridLayout

```
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:rowCount="3"
    android:columnCount="1"
    android:padding="16dp"
    android:background="@drawable/gradient_green">
```

<ImageView

```
    android:layout_width="64dp"
    android:layout_height="64dp"
    android:layout_row="0"
    android:layout_column="0"
    android:layout_gravity="center"
    android:src="@drawable/ic_notifications"
    android:tint="#FFFFFF"/>
```

<TextView

```
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_row="1"
    android:layout_column="0"
    android:layout_gravity="center"
    android:layout_marginTop="8dp"
    android:text="Notifications"
    android:textSize="20sp"
    android:textStyle="bold"
    android:textColor="#FFFFFF"/>
```





```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_row="2"
    android:layout_column="0"
    android:layout_gravity="center"
    android:text="Check Updates"
    android:textColor="#E0FFFFFF"
    android:textSize="14sp"/>
</GridLayout>
</androidx.cardview.widget.CardView>

</GridLayout>
```

### **Java Code:**

#### **MainActivity.java:**

```
package com.example.myapplication;

import androidx.appcompat.app.AppCompatActivity;
import androidx.cardview.widget.CardView;
import android.os.Bundle;
import android.view.View;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```



```
// Find all CardViews
```

```
CardView homeCard = findViewById(R.id.homeCard);
```

```
CardView settingsCard = findViewById(R.id.settingsCard);
```

```
CardView profileCard = findViewById(R.id.profileCard);
```

```
CardView notificationsCard = findViewById(R.id.notificationsCard);
```

```
// Set click listeners for each card
```

```
View.OnClickListener cardClickListener = new View.OnClickListener() {
```

```
    @Override
```

```
    public void onClick(View v) {
```

```
        String message = "";
```

```
        if (v == homeCard) {
```

```
            message = "Home clicked";
```

```
        } else if (v == settingsCard) {
```

```
            message = "Settings clicked";
```

```
        } else if (v == profileCard) {
```

```
            message = "Profile clicked";
```

```
        } else if (v == notificationsCard) {
```

```
            message = "Notifications clicked";
```

```
        }
```

```
        Toast.makeText(Expt02_42441.this, message, Toast.LENGTH_SHORT).show();
```

```
    }
```

```
};
```

```
// Assign click listener to each card
```

```
homeCard.setOnClickListener(cardClickListener);
```

```
settingsCard.setOnClickListener(cardClickListener);
```

```
profileCard.setOnClickListener(cardClickListener);
```

```
notificationsCard.setOnClickListener(cardClickListener);
```

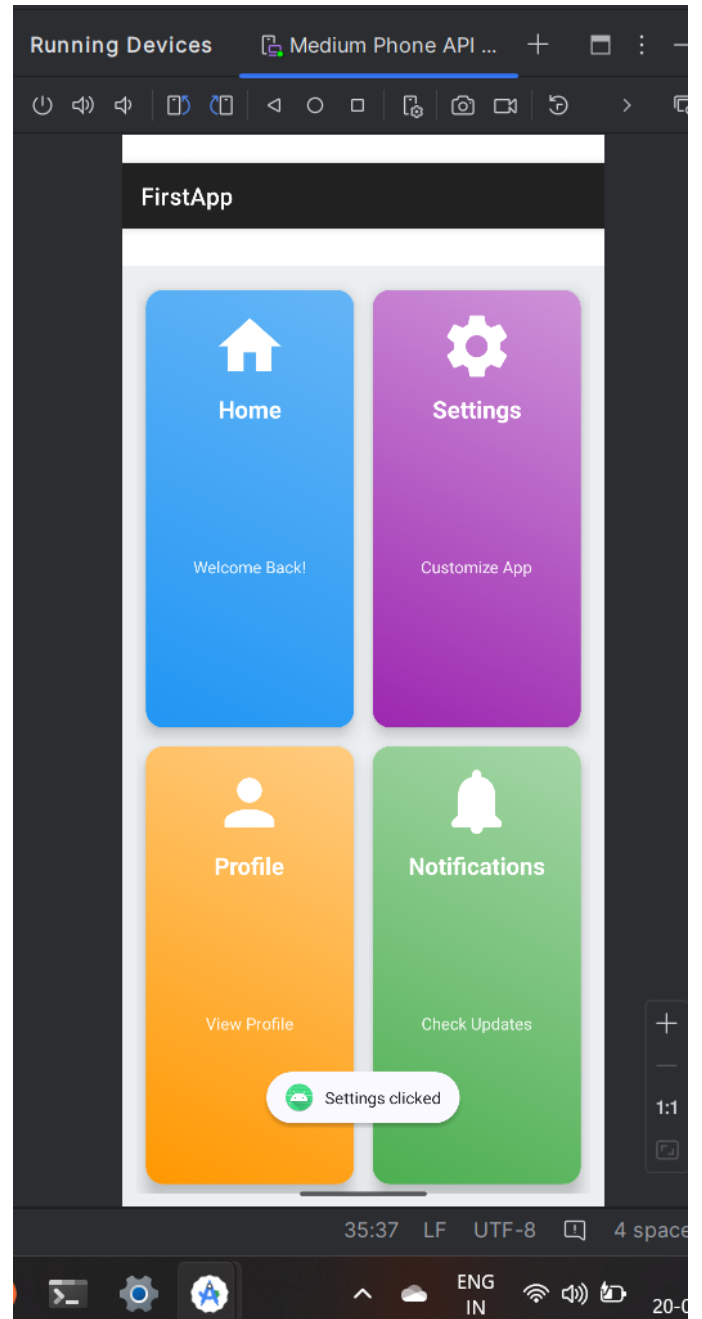
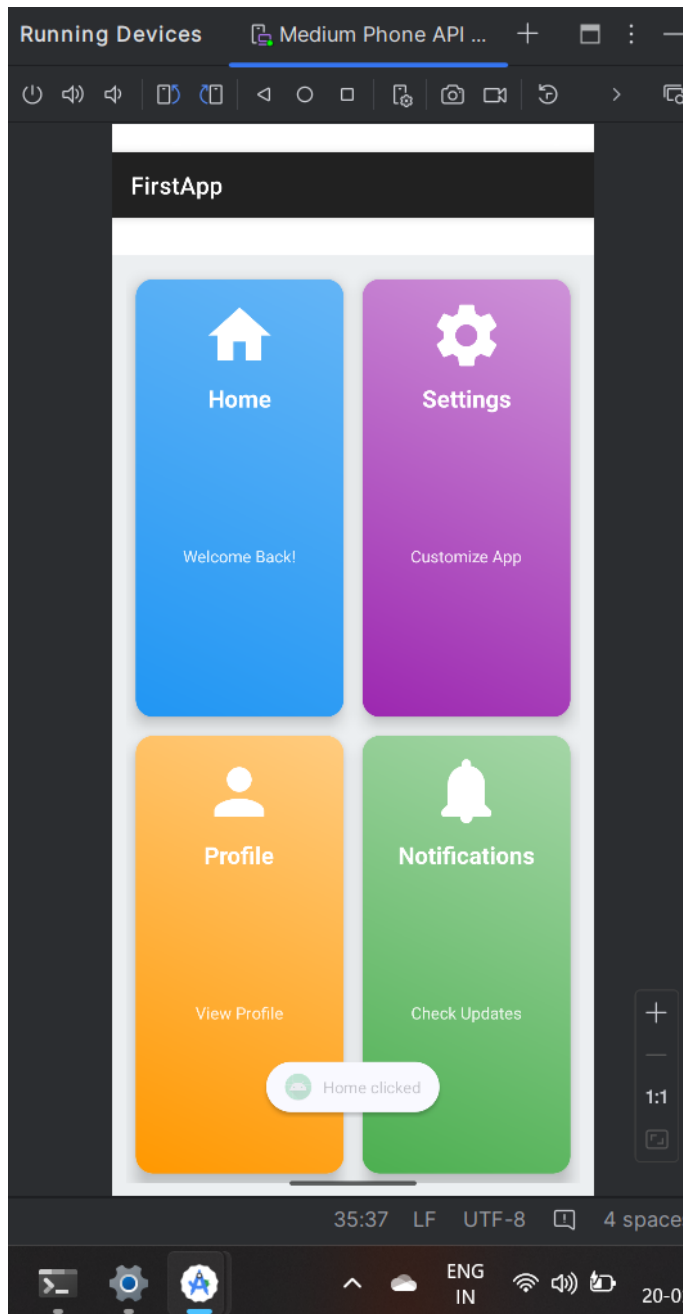
```
}
```

```
}
```



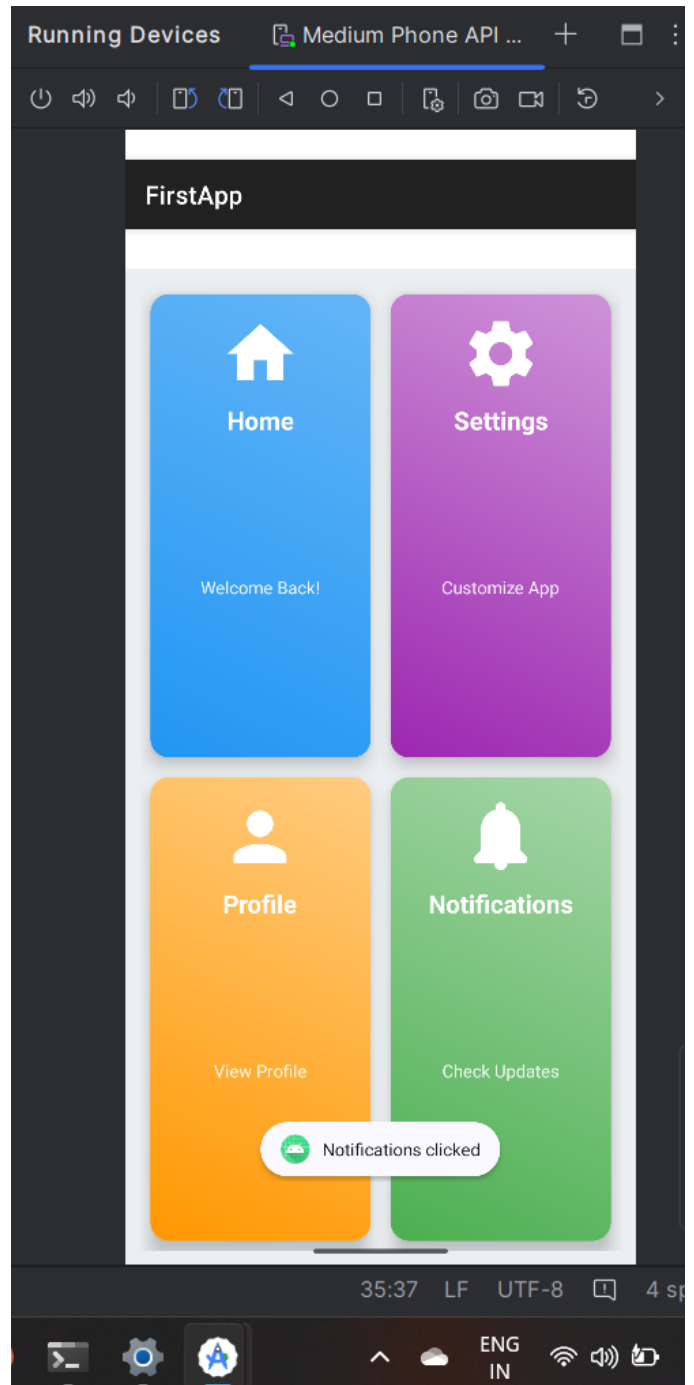
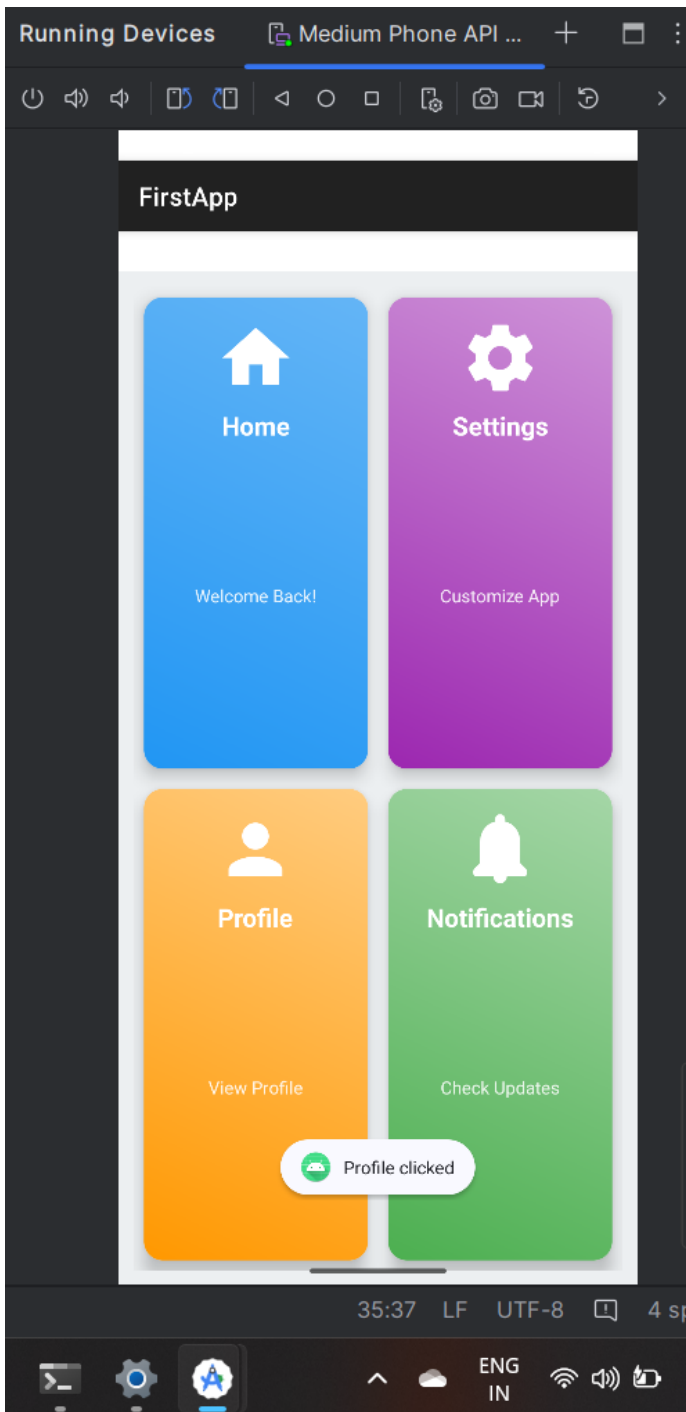
**PUNE INSTITUTE OF COMPUTER TECHNOLOGY, PUNE – 411043**  
**Department of Electronics & Telecommunication Engineering**

**Output:**





**PUNE INSTITUTE OF COMPUTER TECHNOLOGY, PUNE – 411043**  
**Department of Electronics & Telecommunication Engineering**



**Conclusion:**

.....

.....

.....