

WORKSHOP APLIKASI MOBILE
‘PRAKTIKUM 6 IMPLICIT DAN EXPLICIT INTENT’



DISUSUN OLEH :
YULINAR NUR RAHMAWATI
E41191151
TIF GOLONGAN B

PROGRAM STUDI D4 TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI JEMBER
2021

Intent

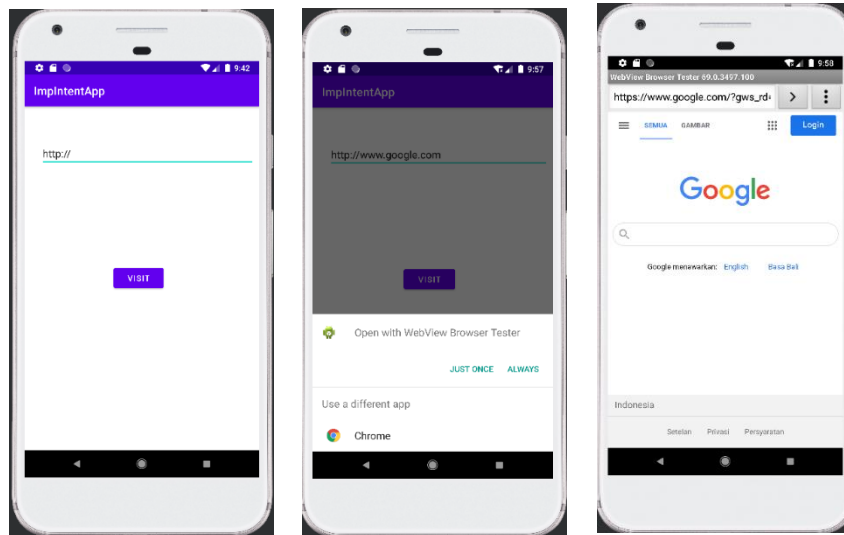
(Implicit dan Explicit Intent)

Intent adalah sebuah kelas dalam programming Android yang berfungsi untuk perpindahan halaman dan juga mekanisme untuk melakukan sebuah action dan komunikasi antar komponen aplikasi. Terdapat tiga penggunaan umum intent dalam aplikasi Android diantaranya yaitu memindahkan satu activity ke activity lain dengan atau tidak membawa data, menjalankan background service dan mengirimkan obyek broadcast ke aplikasi yang membutuhkan.

Intent terbagi menjadi dua yaitu explicit intent dan implicit intent. Explicit intent berfungsi untuk mengaktifkan komponen-komponen dalam satu aplikasi yang sama. Tipe Intent ini digunakan untuk menjalankan komponen dari dalam sebuah aplikasi. Sedangkan implicit intent berfungsi untuk memanggil fungsi activity yang sudah ada di fungsi internal android. Penggunaan tipe intent ini umumnya diperuntukan untuk menjalankan fitur atau fungsi dari komponen aplikasi lain.

Berikut ini hasil *screen capture* apabila aplikasi dijalankan :

1. Implicit Intent

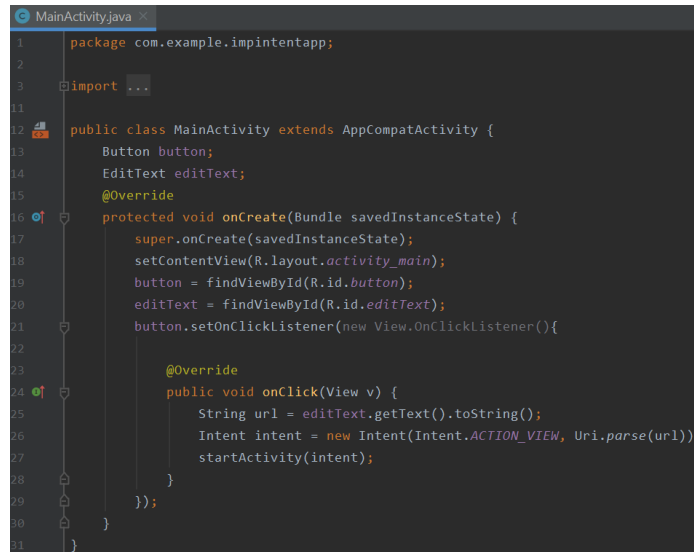


Gambar 1. Implicit Intent

Berikut ini hasil *screen capture* kode program pada aplikasi (ImpIntentApp) dengan *implicit intent* :

a. Kode program pada file *MainActivity.java*

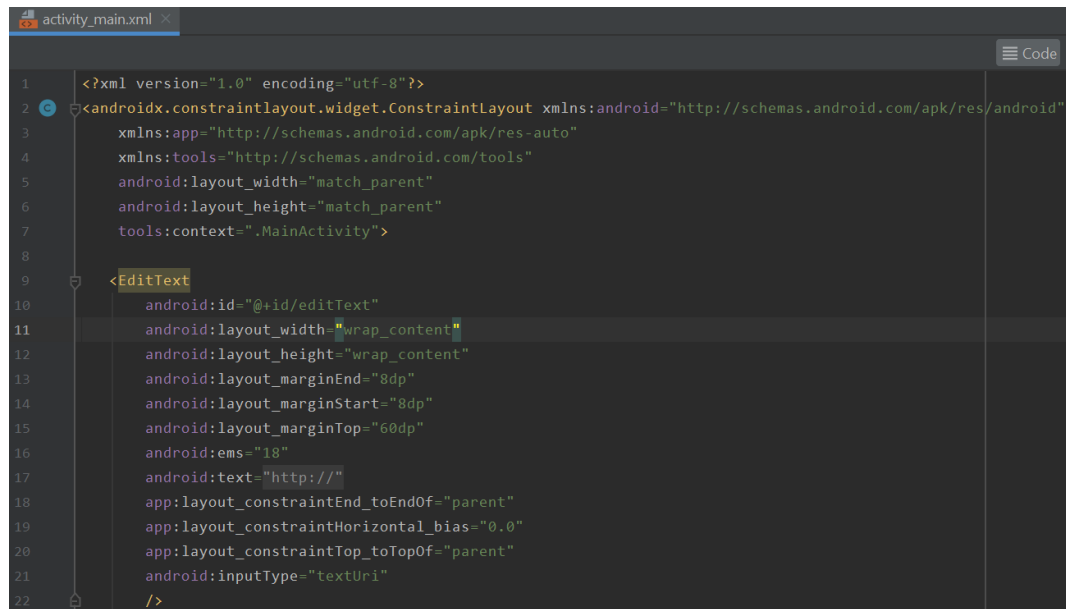
Pada kode program ini memberikan aksi terhadap tombol button saat program dijalankan dengan menggunakan objek intent.



```
1 package com.example.impintentapp;
2
3 import ...
4
5 public class MainActivity extends AppCompatActivity {
6     Button button;
7     EditText editText;
8     @Override
9     protected void onCreate(Bundle savedInstanceState) {
10         super.onCreate(savedInstanceState);
11         setContentView(R.layout.activity_main);
12         button = findViewById(R.id.button);
13         editText = findViewById(R.id.editText);
14         button.setOnClickListener(new View.OnClickListener() {
15
16             @Override
17             public void onClick(View v) {
18                 String url = editText.getText().toString();
19                 Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse(url));
20                 startActivity(intent);
21             }
22         });
23     }
24 }
```

Gambar 1. Kode program MainActivity.java

b. Kode program pada file *activity_main.xml*



```
1 <?xml version="1.0" encoding="utf-8"?>
2 <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     tools:context=".MainActivity">
8
9     <EditText
10         android:id="@+id/editText"
11         android:layout_width="wrap_content"
12         android:layout_height="wrap_content"
13         android:layout_marginEnd="8dp"
14         android:layout_marginStart="8dp"
15         android:layout_marginTop="60dp"
16         android:ems="10"
17         android:text="http://"
18         app:layout_constraintEnd_toEndOf="parent"
19         app:layout_constraintHorizontal_bias="0.0"
20         app:layout_constraintTop_toTopOf="parent"
21         android:inputType="textUri"
22     />
```

```

23     <Button
24         android:id="@+id/button"
25         android:layout_width="wrap_content"
26         android:layout_height="wrap_content"
27         android:layout_marginRight="8dp"
28         android:layout_marginLeft="156dp"
29         android:layout_marginTop="172dp"
30         android:text="Visit"
31         android:textColor="@color/white"
32         app:layout_constraintEnd_toEndOf="parent"
33         app:layout_constraintStart_toStartOf="parent"
34         app:layout_constraintHorizontal_bias="0.0"
35         app:layout_constraintTop_toBottomOf="@+id/editText"
36         android:layout_marginEnd="8dp"
37         android:layout_marginStart="156dp"/>
38 </androidx.constraintlayout.widget.ConstraintLayout>

```

Gambar 2. Kode program activity_main.xml

c. Kode Program AndroidManifest.xml

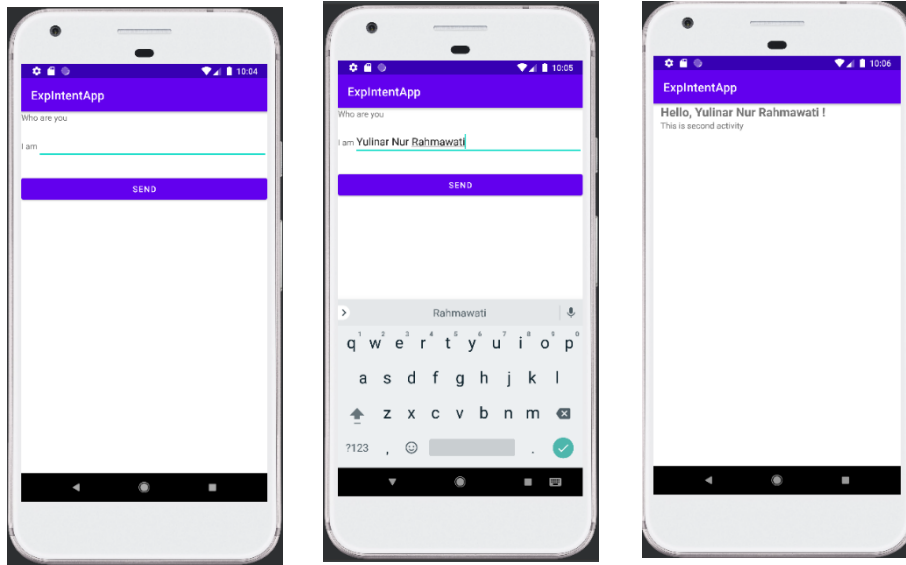
```

AndroidManifest.xml
1  <?xml version="1.0" encoding="utf-8"?>
2  <manifest xmlns:android="http://schemas.android.com/apk/res/android"
3      package="com.example.impintentapp">
4
5      <application
6          android:allowBackup="true"
7          android:icon="@mipmap/ic_launcher"
8          android:label="ImpIntentApp"
9          android:roundIcon="@mipmap/ic_launcher_round"
10         android:supportRtl="true"
11         android:theme="@style/Theme.ImpIntentApp">
12         <activity android:name=".MainActivity">
13             <intent-filter>
14                 <action android:name="android.intent.action.MAIN" />
15
16                 <category android:name="android.intent.category.LAUNCHER" />
17                 <data android:scheme="http" />
18             </intent-filter>
19         </activity>
20     </application>
21
22 </manifest>

```

Gambar 3. Kode program AndroidManifest.xml

2. Explicit Intent



Gambar 4. Explicit Intent

Berikut ini hasil *screen capture* kode program pada aplikasi (ExpIntentApp) dengan *explicit intent* :

a. Kode Program pada file MainActivity.java

```
MainActivity.java x
1 package com.example.expintentapp;
2
3 import ...
11
12 public class MainActivity extends AppCompatActivity {
13     EditText name;
14     Button btnSend;
15     private String KEY_NAME = "NAMA";
16     @Override
17     protected void onCreate(Bundle savedInstanceState) {
18         super.onCreate(savedInstanceState);
19         setContentView(R.layout.activity_main);
20         name = findViewById(R.id.edt_nama);
21         btnSend = findViewById(R.id.btn_send);
```

```

23 btnSend.setOnClickListener(new View.OnClickListener() {
24     @Override
25     public void onClick(View v) {
26         try {
27             String nama = name.getText().toString();
28             if (nama != ""){
29                 Intent i = new Intent( packageContext: MainActivity.this, ActivitySecond.class);
30                 i.putExtra(KEY_NAME, nama);
31                 startActivity(i);
32             } else{
33                 Toast.makeText(getApplicationContext(), text: "YOU NEED TO FILL YOUR NAME", Toast.LENGTH_SHORT);
34             }
35         } catch (Exception e){
36             e.printStackTrace();
37             Toast.makeText(getApplicationContext(), text: "ERROR, TRY AGAIN!", Toast.LENGTH_SHORT);
38         }
39     }
40 });
41 }
42
43 }

```

Gambar 5. MainActivity.java (ExpIntentApp)

b. Kode program activity_main.xml

```

activity_main.xml
1 <?xml version="1.0" encoding="utf-8"?>
2 <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:layout_width="match_parent"
6     android:layout_height="match_parent"
7     tools:context=".MainActivity">
8
9     <TextView
10         android:id="@+id/textView"
11         android:layout_width="wrap_content"
12         android:layout_height="wrap_content"
13         android:text="Who are you"/>
14
15     <TextView
16         android:id="@+id/textView2"
17         android:text="I am"
18         android:layout_width="wrap_content"
19         android:layout_height="wrap_content"
20         android:layout_marginTop="29dp"
21         android:layout_below="@+id/textView"
22         android:layout_alignParentStart="true"/>

```

```

24      <EditText
25          android:layout_width="wrap_content"
26          android:layout_height="wrap_content"
27          android:inputType="textPersonName"
28          android:ems="10"
29          android:id="@+id/edt_nama"
30          android:layout_alignBaseline="@+id/textView2"
31          android:layout_alignBottom="@+id/textView2"
32          android:layout_alignParentEnd="true"
33          android:layout_toEndOf="@+id/textView2"/>
34
35      <Button
36          android:layout_width="wrap_content"
37          android:layout_height="wrap_content"
38          android:text="SEND"
39          android:layout_below="@+id/edt_nama"
40          android:layout_marginTop="26dp"
41          android:id="@+id/btn_send"
42          android:layout_alignParentEnd="true"
43          android:layout_alignParentStart="true"/>
44  </RelativeLayout>

```

Gambar 6. activity_main.xml (ExpIntentApp)

c. Kode Program file ActivitySecond.java

```

ActivitySecond.java
1  package com.example.expintentapp;
2
3  import ...
4
5  public class ActivitySecond extends AppCompatActivity {
6      TextView txtHello;
7      private String nama;
8      private String KEY_NAME = "NAMA";
9
10     @Override
11     protected void onCreate(Bundle savedInstanceState) {
12         super.onCreate(savedInstanceState);
13         setContentView(R.layout.activity_second);
14
15         txtHello = findViewById(R.id.txtHello);
16         Bundle extras = getIntent().getExtras();
17         nama = extras.getString(KEY_NAME);
18         txtHello.setText("Hello, " + nama + " !");
19     }
20 }

```

Gambar 7. ActivitySecond.java (ExpIntentApp)

d. Kode program activity_second.xml

```
1  <?xml version="1.0" encoding="utf-8"?>
2  <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      android:id="@+id/activity_2"
8      tools:context=".ActivitySecond">
9
10     <TextView
11         android:layout_width="wrap_content"
12         android:layout_height="wrap_content"
13         android:layout_alignParentTop="true"
14         android:layout_alignParentStart="true"
15         android:textSize="20sp"
16         android:layout_marginStart="12dp"
17         android:textStyle="bold"
18         android:id="@+id/txtHello" />
19
20     <TextView
21         android:layout_width="wrap_content"
22         android:layout_height="wrap_content"
23         android:text="This is second activity"
24         android:layout_below="@+id/txtHello"
25         android:layout_alignStart="@+id/txtHello"
26         android:id="@+id/textView3" />
27
28 </RelativeLayout>
```

Gambar 8. Activity_second.xml (ExpIntentApp)

Praktikum 3 Layout

Dengan penggunaan intent pada setiap tombol yang ada, sehingga dapat melakukan perpindahan halaman.

TUGAS PRAKTIKUM 3 :

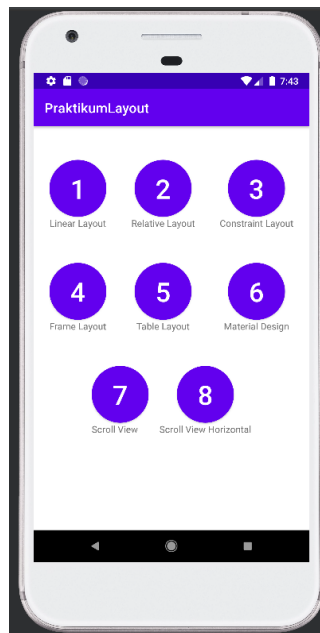
Layout merupakan suatu tampilan tata letak di Android untuk mengatur penempatan bermacam-macam komponen di atasnya sehingga mendukung tampilan pada aplikasi yang dibuat terlihat rapi dan nyaman dilihat oleh pengguna.

Berikut ini penjelasan mengenai tugas praktikum 3 :

- Tugas Layout

Berikut ini *screen capture* tugas layout :

Activity_main.xml



Gambar 1. Tugas Layout

Pada gambar diatas merupakan hasil *screen capture* tugas layout ketika dijalankan menggunakan emulator. Pada gambar diatas menggunakan *table layout* dengan memanfaatkan baris dan kolom yang ada pada layout tersebut dan juga menggunakan *linear*

layout dengan orientasi vertikal. Dalam *layout* tersebut terdapat *Button* dengan mengatur *background*-nya sehingga membentuk oval. Pada setiap tombol tersebut memiliki aksi untuk menuju ke masing-masing tema *layout* berdasarkan text yang ada dibawah tombol tersebut menggunakan *OnClick*.

Berikut ini *source code* dari *activity_main.xml* :

```
activity_main.xml x
1  <?xml version="1.0" encoding="utf-8"?>
2  <TableLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      tools:context=".MainActivity">
8
9      <TableRow
10         android:layout_marginTop="50dp"
11         android:gravity="center"
12         android:orientation="vertical">
13
14         <LinearLayout
15             android:gravity="center"
16             android:orientation="vertical">
17
18             <Button
19                 android:id="@+id/btn_linear_layout"
20                 android:layout_width="85dp"
21                 android:layout_height="85dp"
22                 android:text="@string/satu_1"
23                 android:textSize="40sp"
24                 android:backgroundTint="@null"
25                 android:onClick="Tombol1"
26                 android:background="@drawable/bg_button"/>
27
28             <TextView
29                 android:layout_width="wrap_content"
30                 android:layout_height="wrap_content"
31                 android:text="Linear Layout"/>
32         </LinearLayout>

```

```

34      <LinearLayout
35          android:gravity="center"
36          android:layout_marginStart="16dp"
37          android:orientation="vertical">
38
39          <Button
40              android:layout_width="85dp"
41              android:layout_height="85dp"
42              android:text="2"
43              android:textSize="40sp"
44              android:backgroundTint="@null"
45              android:onClick="Tombol2"
46              android:background="@drawable/bg_button"/>
47
48          <TextView
49              android:layout_width="wrap_content"
50              android:layout_height="wrap_content"
51              android:text="Relative Layout"/>
52      </LinearLayout>

```

```

54      <LinearLayout
55          android:gravity="center"
56          android:layout_marginStart="16dp"
57          android:orientation="vertical">
58
59          <Button
60              android:layout_width="85dp"
61              android:layout_height="85dp"
62              android:text="3"
63              android:textSize="40sp"
64              android:backgroundTint="@null"
65              android:onClick="Tombol3"
66              android:background="@drawable/bg_button"/>
67
68          <TextView
69              android:layout_width="wrap_content"
70              android:layout_height="wrap_content"
71              android:text="Constraint Layout"/>
72      </LinearLayout>
73  </TableRow>

```

```

75  <TableRow
76      android:layout_marginTop="50dp"
77      android:gravity="center"
78      android:orientation="vertical">
79
80      <LinearLayout
81          android:gravity="center"
82          android:orientation="vertical">
83
84          <Button
85              android:layout_width="85dp"
86              android:layout_height="85dp"
87              android:text="4"
88              android:textSize="40sp"
89              android:backgroundTint="@null"
90              android:onClick="Tombol4"
91              android:background="@drawable/bg_button"/>
92
93          <TextView
94              android:layout_width="wrap_content"
95              android:layout_height="wrap_content"
96              android:text="Frame Layout"/>
97      </LinearLayout>

```

```

99      <LinearLayout
100          android:gravity="center"
101          android:layout_marginStart="16dp"
102          android:orientation="vertical">
103
104          <Button
105              android:layout_width="85dp"
106              android:layout_height="85dp"
107              android:text="5"
108              android:textSize="40sp"
109              android:backgroundTint="@null"
110              android:onClick="Tombol5"
111              android:background="@drawable/bg_button"/>
112
113          <TextView
114              android:layout_width="wrap_content"
115              android:layout_height="wrap_content"
116              android:text="Table Layout"/>
117      </LinearLayout>

```

```

119      <LinearLayout
120          android:gravity="center"
121          android:layout_marginStart="16dp"
122          android:orientation="vertical">
123
124          <Button
125              android:layout_width="85dp"
126              android:layout_height="85dp"
127              android:text="6"
128              android:textSize="40sp"
129              android:backgroundTint="@null"
130              android:onClick="Tombol6"
131              android:background="@drawable/bg_button" />
132
133          <TextView
134              android:layout_width="wrap_content"
135              android:layout_height="wrap_content"
136              android:text="Material Design"/>
137      </LinearLayout>
138  </TableRow>
139

```

Gambar 2. Source Code activity_main.xml

Baris dua (2) pada *source code* tersebut sudah mengidentifikasi bahwa layout yang digunakan yaitu *table layout*. Pada *table layout*, terdapat beberapa tag diantaranya yaitu tag *TableRow* seperti pada baris sembilan (9). *TableRow* digunakan untuk membuat baris.

Pada baris 9 – 73, menggunakan tag *TableRow* yang didalamnya terdapat *linear layout* untuk mengatur tombol (button) dan juga tulisan yang ada dibawah tombol seperti pada *screen capture* hasil *running* tersebut.



Gambar 3. Tombol di Main Activity

Pada gambar diatas tombol 1 hingga tombol 3 terdapat di dalam satu tag *TableRow* dengan tag *LinearLayout* yang berbeda.



Gambar 4. Tombol dengan Linear Layout

Pada gambar diatas, satu tombol dan juga satu textview yang berada dibawahnya menggunakan satu tag *linear layout* dengan orientasi vertikal sehingga komponen *button* dan *textview* membentuk mendatar (vertikal).

Berikut ini *source code* untuk tombol *button* yang disimpan pada *app>res>drawable* :

bg_button.xml

```
bg_button.xml
1  <?xml version="1.0" encoding="utf-8"?>
2  <shape
3      android:shape="oval"
4      xmlns:android="http://schemas.android.com/apk/res/android">
5
6      <stroke
7          android:width="1dp"
8          android:color="@color/purple_700"/>
9
10     <solid
11         android:color="@android:color/holo_purple"/>
12 </shape>
```

Gambar 5. Bg_button.xml

Pada *source code* diatas menggunakan tag *shape* dengan nilai atau bentuk (*oval*), sehingga pada *button* yang menggunakan akan membentuk sebuah bentuk oval dan juga diatur dengan warna di dalamnya. Sedangkan pada gambar dibawah ini ditempatkan pada *script Button* yang ada, untuk memanggil dan menggunakan file *bg_button.xml*

```

<Button
    android:id="@+id/btn_linear_layout"
    android:layout_width="85dp"
    android:layout_height="85dp"
    android:text="1"
    android:textSize="40sp"
    android:backgroundTint="@null"
    android:onClick="Tombol1"
    android:background="@drawable/bg_button"/>

```

Gambar 6. Background Button

Berikut ini *source code* dari *MainActivity.java* :

```

bg_button.xml × activity_main.xml × MainActivity.java ×
1 package com.example.praktikumlayout;
2
3 import ...
4
5
6
7
8
9 public class MainActivity extends AppCompatActivity {
10
11     @Override
12     protected void onCreate(Bundle savedInstanceState) {
13         super.onCreate(savedInstanceState);
14         setContentView(R.layout.activity_main);
15     }
16
17     public void Tombol1(View view) {
18         Intent btnLinear = new Intent( packageContext: MainActivity.this, LinearLayout.class);
19         startActivity(btnLinear);
20     }
21
22     public void Tombol2(View view) {
23         Intent btnRelative = new Intent( packageContext: MainActivity.this, RelativeLayout.class);
24         startActivity(btnRelative);
25     }
26
27     public void Tombol3(View view) {
28         Intent btnConstraint = new Intent( packageContext: MainActivity.this, ConstraintLayout.class);
29         startActivity(btnConstraint);
30     }

```

```

32 public void Tombol4(View view) {
33     Intent btnFrame = new Intent( packageContext: MainActivity.this, FrameLayout.class);
34     startActivity(btnFrame);
35 }
36
37 public void Tombol5(View view) {
38     Intent btnTable = new Intent( packageContext: MainActivity.this, TableLayout.class);
39     startActivity(btnTable);
40 }
41
42 public void Tombol6(View view) {
43     Intent btnMaterial = new Intent( packageContext: MainActivity.this, MaterialDesign.class);
44     startActivity(btnMaterial);
45 }
46
47 public void Tombol7(View view) {
48     Intent btnScroll = new Intent( packageContext: MainActivity.this, ScrollView.class);
49     startActivity(btnScroll);
50 }
51
52 public void Tombol8(View view) {
53     Intent btnScrollHor = new Intent( packageContext: MainActivity.this, ScrollViewHorizontal.class);
54     startActivity(btnScrollHor);
55 }
56 }

```

Gambar 7. MainActivity.java

Pada *source code* diatas digunakan untuk menghubungkan beberapa *activity* ketika *button* atau tombol di klik. Hal pertama untuk menghubungkan *activity* tersebut dengan mendeklarasikan *OnClick* terlebih dahulu pada bagian tombol yang digunakan untuk menuju ke *activity* lainnya, seperti pada gambar dibawah ini :

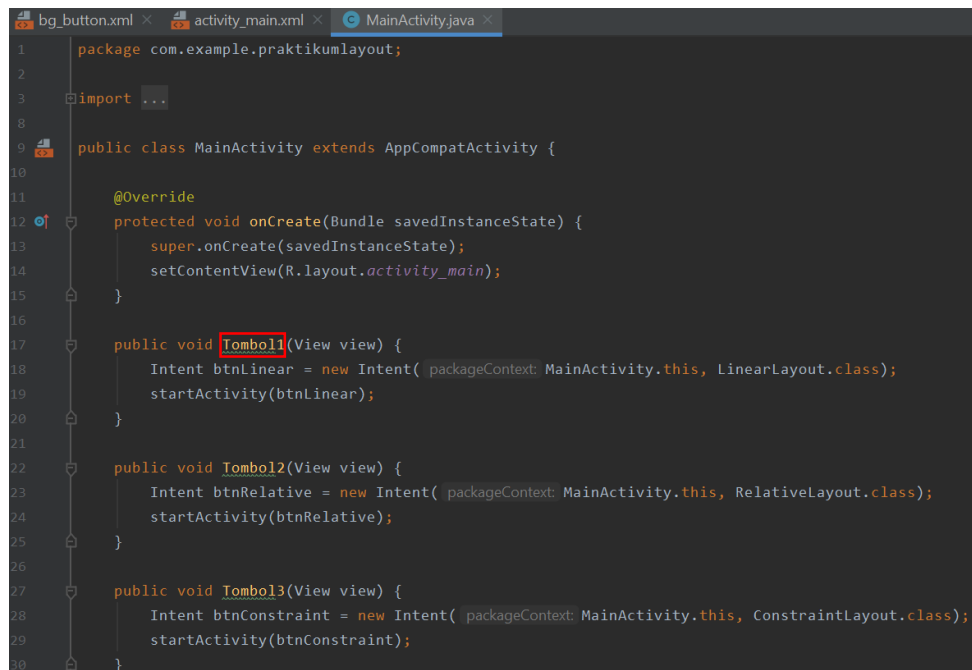
```

<Button
    android:id="@+id/btn_linear_layout"
    android:layout_width="85dp"
    android:layout_height="85dp"
    android:text="1"
    android:textSize="40sp"
    android:backgroundTint="@null"
    android:onClick="Tombol1"
    android:background="@drawable/bg_button"/>

```

Gambar 8. OnClick pada Button

Gambar diatas saling berkaitan dengan gambar yang ada dibawah ini :



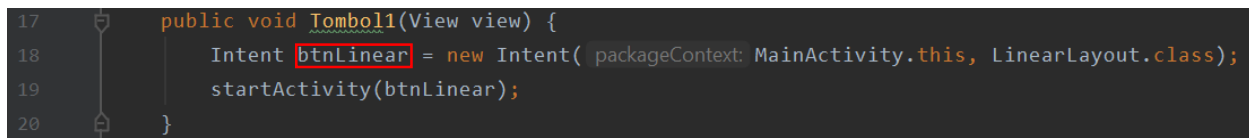
```

1 package com.example.praktikumlayout;
2
3 import ...
4
5
6
7
8
9 public class MainActivity extends AppCompatActivity {
10
11     @Override
12     protected void onCreate(Bundle savedInstanceState) {
13         super.onCreate(savedInstanceState);
14         setContentView(R.layout.activity_main);
15     }
16
17     public void Tombol1(View view) {
18         Intent btnLinear = new Intent( packageContext: MainActivity.this, LinearLayout.class);
19         startActivity(btnLinear);
20     }
21
22     public void Tombol2(View view) {
23         Intent btnRelative = new Intent( packageContext: MainActivity.this, RelativeLayout.class);
24         startActivity(btnRelative);
25     }
26
27     public void Tombol3(View view) {
28         Intent btnConstraint = new Intent( packageContext: MainActivity.this, ConstraintLayout.class);
29         startActivity(btnConstraint);
30     }
31 }

```

Gambar 9. MainActivity-Tombol1.java

Baris 17 – 20 pada gambar diatas digunakan untuk berpindah *activity* begitu juga dengan baris 22 - 24 dan baris-baris seterusnya dengan mendeklarasikan objek Intent yang disertai dengan nama objeknya.



```

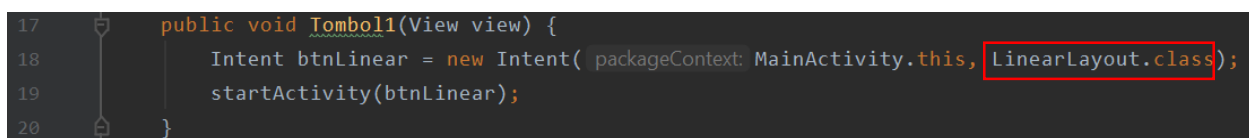
17 public void Tombol1(View view) {
18     Intent btnLinear = new Intent( packageContext: MainActivity.this, LinearLayout.class);
19     startActivity(btnLinear);
20 }

```

Gambar 10. Penggunaan Intent

Pada *source code* diatas terdapat pada *MainActivity.java* untuk itu pada baris 18 terdapat *code* (.this) yang digunakan untuk menunjukkan *MainActivity* ketika tombol tersebut di klik maka akan berpindah pada Linear Layout.

Pada baris 19, *startActivity* digunakan untuk memulai sebuah *activity* lain yang sudah di deklarasikan pada objek *Intent* dengan nama “*btnLinear*” yang berfungsi ketika tombol di klik maka *startActivity* akan memulai sebuah *activity* yang sudah di deklarasikan.



```

17 public void Tombol1(View view) {
18     Intent btnLinear = new Intent( packageContext: MainActivity.this, LinearLayout.class);
19     startActivity(btnLinear);
20 }

```

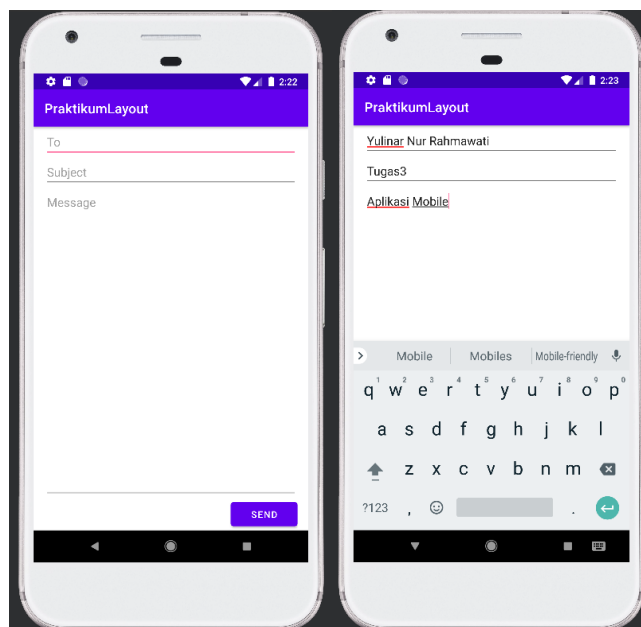

Gambar 11. Penggunaan startActivity

Berikut ini beberapa penjelasan layout, material design dan juga scroll view :

1. Linear Layout

Linier Layout merupakan desain atau tampilan pada aplikasi dengan tata letak secara vertikal dan horizontal dimana tata letak aplikasi hanya bisa memasukan media secara mendatar dan menurun.

Berikut ini *screen capture* pengaplikasian linear layout :



Gambar 12. Linear Layout

Pada gambar diatas merupakan pengaplikasian dengan *linear layout* secara vertikal atau mendatar disertai dengan penggunaan beberapa komponen yaitu *EditText* dan juga *Button*. *EditText* digunakan untuk memasukkan teks pada aplikasi, sedangkan *Button* digunakan sebagai tombol.

Berikut ini *screen capture source code* linear layout :

activity_linear_layout.xml

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      android:paddingStart="16dp"
8      android:paddingEnd="16dp"
9      android:orientation="vertical"
10     tools:context=".LinearLayout">
11
12     <EditText
13         android:id="@+id/edit_to"
14         android:hint="To"
15         android:layout_width="match_parent"
16         android:layout_height="wrap_content" />
17
18     <EditText
19         android:id="@+id/edt_subject"
20         android:hint="Subject"
21         android:layout_width="match_parent"
22         android:layout_height="wrap_content" />
23
24     <EditText
25         android:id="@+id/edt_msg"
26         android:layout_width="match_parent"
27         android:layout_height="0dp"
28         android:gravity="top"
29         android:layout_weight="1"
30         android:hint="Message" />
31
32     <Button
33         android:id="@+id/submit"
34         android:layout_width="100dp"
35         android:layout_height="wrap_content"
36         android:layout_gravity="right"
37         android:text="Send" />
38 </LinearLayout>

```

Gambar 13. Source Code Linear Layout

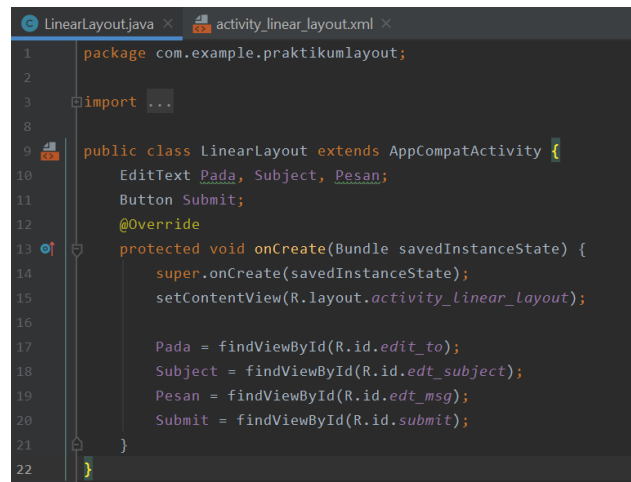
Pada baris kode program tersebut menunjukkan penggunaan *linear layout* pada baris ke dua dengan lebar dan tinggi *layout* menggunakan “*match_parent*” serta menggunakan orientasi vertikal atau mendatar dalam penggunaan *linear layout* tersebut. Dalam mengatur sebuah lebar dan tinggi *layout* atau komponen terdapat dua macam yaitu :

- Match_parent*, digunakan untuk mengatur lebar atau tinggi dengan mengikuti layar atau *screen* pada perangkat android.
- Wrap_content*, digunakan untuk mengatur lebar atau tinggi dengan mengikuti objek di dalam komponen.

Pada baris kode program tersebut juga terdapat dua komponen yaitu *EditText* dan juga *Button*, masing-masing komponen tersebut diatur melalui *source code* seperti pada gambar diatas dengan beberapa penjelasan seperti berikut :

- a. *Android:hint*, digunakan untuk membuat sebuah text bertulis pada komponen *EditText* semacam *placeholder*.
- b. *Android:gravity*, digunakan untuk menentukan atau menempatkan objek di dalam objek itu sendiri. Pada baris 27, *gravity* dengan nilai *top* atau diatas sehingga penempatan *EditText* tersebut berada pada bagian atas.
- c. *Android:text*, digunakan untuk memberikan sebuah keterangan berupa teks di dalamnya. Pada baris 34, text tersebut berupa tulisan “*Send*” pada komponen *Button* yang digunakan sebagai tombol.

Berikut ini *source code LinearLayout.java* :



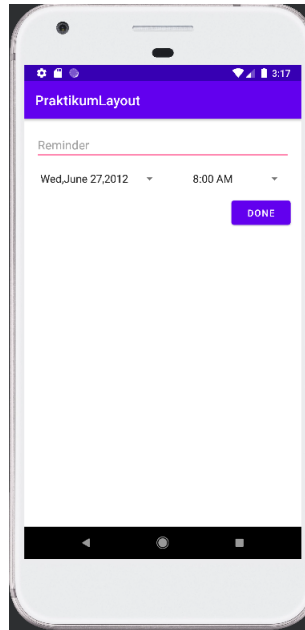
```
1 package com.example.praktikumlayout;
2
3 import ...
4
5
6
7
8
9 public class LinearLayout extends AppCompatActivity {
10     EditText Pada, Subject, Pesan;
11     Button Submit;
12     @Override
13     protected void onCreate(Bundle savedInstanceState) {
14         super.onCreate(savedInstanceState);
15         setContentView(R.layout.activity_linear_layout);
16
17         Pada = findViewById(R.id.edit_to);
18         Subject = findViewById(R.id.edt_subject);
19         Pesan = findViewById(R.id.edt_msg);
20         Submit = findViewById(R.id.submit);
21     }
22 }
```

Gambar 14. LinearLayout.java

2. Relative Layout

Relative Layout merupakan desain tampilan pada aplikasi dengan penataan yang menempatkan *widget-widget* di dalamnya seperti *layer*, sehingga sebuah *widget* dapat berada di atas atau di bawah *widget* lainnya dengan kata lain *relative layout* memiliki penempatan secara bebas tanpa adanya aturan layaknya *Linear Layout*.

Berikut ini *screen capture* pengaplikasian *relative layout* :



Gambar 15. Relative Layout

Pada gambar diatas merupakan pengaplikasian *relative layout* dengan beberapa komponen di dalamnya, seperti :

- a. *EditText*, digunakan untuk memasukkan atau menginputkan sebuah teks.
- b. *Spinner*, digunakan untuk menampilkan *list* atau *dropdown* menu yang terdiri dari item-item untuk membantu pengguna dalam memilih sebuah item pada aplikasi. Dalam menggunakan *spinner* untuk menyimpan item atau *value*-nya dapat menggunakan sebuah *array*.
- c. *Button*, digunakan sebagai tombol.

Berikut ini *screen capture source code* relative layout :

activity_relative_layout.xml

```

activity_relative_layout.xml ×
1  <?xml version="1.0" encoding="utf-8"?>
2  <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      tools:context=".RelativeLayout">
8
9      <EditText
10         android:id="@+id/edt_name"
11         android:layout_width="match_parent"
12         android:layout_height="wrap_content"
13         android:hint="Reminder"
14         android:layout_marginTop="16dp"
15         android:layout_marginStart="16dp"
16         android:layout_marginEnd="16dp" />

```

```

18     <Spinner
19         android:id="@+id/dates"
20         android:layout_width="wrap_content"
21         android:layout_height="wrap_content"
22         android:layout_marginStart="16dp"
23         android:paddingTop="16dp"
24         android:paddingBottom="16dp"
25         android:layout_below="@+id/edt_name"
26         android:spinnerMode="dropdown"
27         android:entries="@array/reminder_tgl" />
28
29     <Spinner
30         android:id="@+id/spinner_jam"
31         android:layout_width="wrap_content"
32         android:layout_height="wrap_content"
33         android:layout_below="@id/edt_name"
34         android:layout_alignParentEnd="true"
35         android:layout_marginStart="16dp"
36         android:layout_marginEnd="16dp"
37         android:layout_toEndOf="@id/dates"
38         android:entries="@array/jam"
39         android:padding="16dp"
40         android:spinnerMode="dropdown" />

```

```

42     <Button
43         android:id="@+id/btn_done"
44         android:text="Done"
45         android:layout_width="wrap_content"
46         android:layout_height="wrap_content"
47         android:layout_below="@id/spinner_jam"
48         android:layout_alignParentEnd="true"
49         android:layout_marginEnd="16dp" />
50 </RelativeLayout>

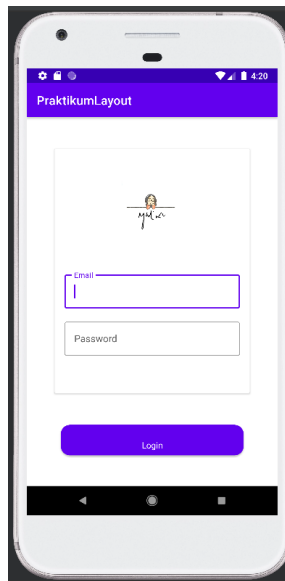
```

Gambar 16. Source Code Relative Layout

3. Constraint Layout

Constraint layout disebut juga pengembangan dari *relative layout* karena letak *view* bergantung pada *view* lain dalam satu *layout* ataupun dengan *parent layout*-nya. Dalam *relative layout* dapat meletakkan sebuah *view* di atas, bawah, atau samping *view* lain.

Berikut ini *screen capture* pengaplikasian *constraint layout* :



Gambar 17. Constraint Layout

Pada gambar diatas merupakan pengaplikasian *constraint layout* dengan menggunakan *Card View* sebagai pembungkusan data yang ditandai dengan tampilan view nya yang memiliki *drop shadow (elevation)* dan *corner radius* dan juga terdapat beberapa komponen di dalamnya, seperti :

- a. *TextView*, digunakan untuk menampilkan *output* berupa teks seperti pada gambar diatas dengan *output* “Login”.
- b. *TextInputLayout* atau *TextField*, digunakan untuk sebuah text control yang memungkinkan pengguna menginputkan atau memasukkan teks singkat.
- c. *ImageView*, komponen dasar android yang tugasnya menampilkan gambar.

Berikut ini *screen capture source code* constraint layout :

activity_constraint_layout.xml

```
activity_constraint_layout.xml <
1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout
3      xmlns:android="http://schemas.android.com/apk/res/android"
4      xmlns:app="http://schemas.android.com/apk/res-auto"
5      xmlns:tools="http://schemas.android.com/tools"
6      android:layout_width="match_parent"
7      android:layout_height="match_parent"
8      tools:context=".ConstraintLayout">
9
10     <androidx.cardview.widget.CardView
11         android:id="@+id/cardView"
12         android:layout_width="300dp"
13         android:layout_height="50dp"
14         android:elevation="20dp"
15         app:cardCornerRadius="15dp"
16         app:cardBackgroundColor="@color/purple_500"
17         android:layout_marginBottom="50dp"
18         app:layout_constraintBottom_toBottomOf="parent"
19         app:layout_constraintEnd_toEndOf="parent"
20         app:layout_constraintStart_toStartOf="parent">
21
22     <androidx.constraintlayout.widget.ConstraintLayout
23         android:layout_width="match_parent"
24         android:layout_height="match_parent">
25
26         <TextView
27             android:id="@+id/txt_login"
28             android:text="@string/login"
29             android:textColor="@color/white"
30             android:layout_marginTop="16dp"
31             app:layout_constraintTop_toTopOf="parent"
32             app:layout_constraintBottom_toBottomOf="parent"
33             app:layout_constraintEnd_toEndOf="parent"
34             app:layout_constraintStart_toStartOf="parent"
35             android:layout_width="wrap_content"
36             android:layout_height="wrap_content" />
37     </androidx.constraintlayout.widget.ConstraintLayout>
38
39 </androidx.cardview.widget.CardView>
```

```

41 <androidx.cardview.widget.CardView
42     android:layout_width="320dp"
43     android:layout_height="400dp"
44     app:layout_constraintStart_toStartOf="parent"
45     app:layout_constraintEnd_toEndOf="parent"
46     app:layout_constraintTop_toTopOf="parent"
47     app:layout_constraintBottom_toTopOf="@+id/cardView">
48
49     <androidx.constraintlayout.widget.ConstraintLayout
50         android:layout_width="match_parent"
51         android:layout_height="match_parent">
52
53         <ImageView
54             android:id="@+id/img_logo"
55             android:layout_width="100dp"
56             android:layout_height="100dp"
57             android:src="@drawable/yulinar"
58             android:layout_marginTop="50dp"
59             app:layout_constraintTop_toTopOf="parent"
60             app:layout_constraintEnd_toEndOf="parent"
61             app:layout_constraintStart_toStartOf="parent" />

```

```

63 <com.google.android.material.textfield.TextInputLayout
64     android:id="@+id/edt_txt_email"
65     android:layout_width="match_parent"
66     android:layout_height="wrap_content"
67     android:layout_marginTop="50dp"
68     android:layout_marginStart="16dp"
69     android:layout_marginEnd="16dp"
70     style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
71     app:layout_constraintStart_toStartOf="parent"
72     app:layout_constraintTop_toBottomOf="@id/img_logo">
73
74     <com.google.android.material.textfield.TextInputEditText
75         android:layout_width="match_parent"
76         android:layout_height="wrap_content"
77         android:hint="Email"
78         android:inputType="textEmailAddress" />
79
80 </com.google.android.material.textfield.TextInputLayout>

```

```

82 <com.google.android.material.textfield.TextInputLayout
83     android:layout_width="match_parent"
84     android:layout_height="wrap_content"
85     android:layout_marginTop="16dp"
86     android:layout_marginStart="16dp"
87     android:layout_marginEnd="16dp"
88     style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
89     app:layout_constraintTop_toBottomOf="@id/edt_txt_email">
90
91     <com.google.android.material.textfield.TextInputEditText
92         android:hint="Password"
93         android:inputType="textPassword"
94         android:layout_width="match_parent"
95         android:layout_height="wrap_content" />
96
97 </com.google.android.material.textfield.TextInputLayout>
98
99 </androidx.constraintlayout.widget.ConstraintLayout>
100 </androidx.cardview.widget.CardView>
101
102 </androidx.constraintlayout.widget.ConstraintLayout>

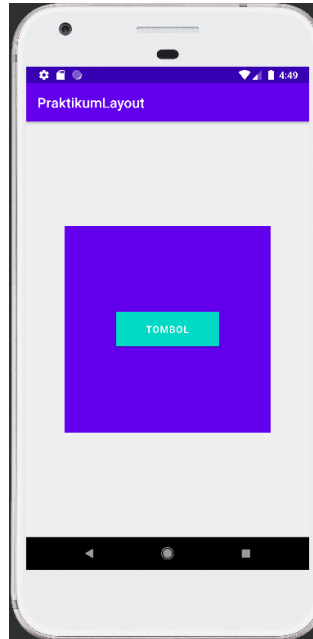
```

Gambar 18. Source Code Constraint Layout

4. Frame Layout

Frame Layout merupakan *subclass ViewGroup* yang digunakan untuk menentukan posisi beberapa tampilan yang ditempatkan di atas satu sama lain untuk mewakili layar tampilan tunggal atau dengan kata lain membuat objek yang saling bertindihan.

Berikut ini *screen capture* pengaplikasian *frame layout* :



Gambar 19. Frame Layout

Pada gambar diatas merupakan pengaplikasian *frame layout* yang terlihat penggunaan dua komponen yang saling bertindih dengan komponen atau objek *button* yang berada diatas objek atau komponen lain yakni *ImageView*.

Berikut ini *screen capture source code* frame layout :

activity_frane_layout.xml

```

1  <?xml version="1.0" encoding="utf-8"?>
2  <FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      android:background="#1113"
8      tools:context=".FrameLayout">
9
10     <ImageView
11         android:layout_width="300dp"
12         android:layout_height="300dp"
13         android:layout_gravity="center"
14         android:background="@color/purple_500"/>
15
16     <Button
17         android:layout_width="150dp"
18         android:layout_height="50dp"
19         android:text="Tombol"
20         android:background="@color/teal_200"
21         app:backgroundTint="@null"
22         android:layout_gravity="center"/>
23
24 </FrameLayout>

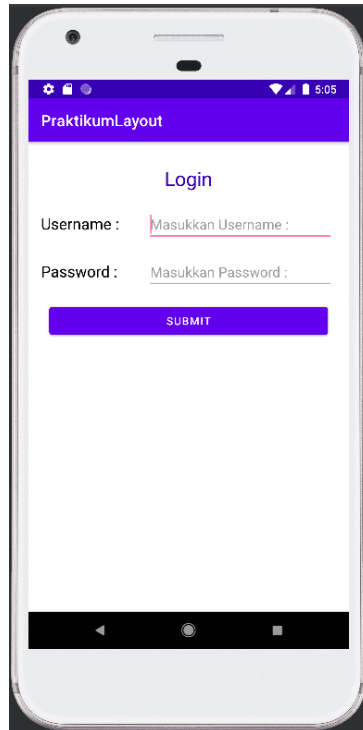
```

Gambar 20. Source Code Frame Layout

5. Table Layout

Table Layout adalah sebuah layout yang terdiri dari kolom dan baris, setiap objek atau *view* yang berada di dalam table layout akan tersusun seperti tabel, yaitu kolom dan juga baris. *Row* atau baris pada dasarnya digunakan untuk menyimpan satu jenis *record*, hanya satu informasi yang dapat disimpan. Sedangkan kolom adalah sub bagian terbagi dari setiap baris dan satu baris dapat menampung beberapa jenis kolom. Setiap kolom terdiri dari jenis informasi yang berbeda mengenai baris tersebut. Table Layout juga digunakan untuk mempermudah dalam merancang *user interface* pada *activity*.

Berikut ini *screen capture* pengaplikasian *table layout* :



Gambar 21. Table Layout

Pada gambar diatas merupakan pengaplikasian dari *table layout* yang dibuat dengan berdasarkan konsep sebuah table yakni baris dan kolom. Pada *table layout* terdapat beberapa tag yang dapat digunakan. Pada gambar diatas *table layout* terdiri dari beberapa komponen seperti :

- TextView*, digunakan untuk menampilkan *output* berupa teks.
- EditText*, digunakan untuk memasukkan atau menginputkan sebuah teks.
- Button*, biasa digunakan sebagai tombol.

Berikut ini *screen capture source code* table layout :

activity_table_layout.xml

```

activity_table_layout.xml
1  <?xml version="1.0" encoding="utf-8"?>
2  <TableLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      tools:context=".TableLayout">
8
9      <TableRow
10         android:paddingTop="10dp"
11         android:gravity="center"
12         android:layout_marginTop="20dp">
13
14         <TextView
15             android:id="@+id/status"
16             android:layout_width="wrap_content"
17             android:layout_height="wrap_content"
18             android:layout_gravity="center"
19             android:layout_span="2"
20             android:text="Login"
21             android:textSize="25sp"
22             android:textColor="@color/purple_700" />
23     </TableRow>

```

```

25     <TableRow
26         android:layout_marginTop="20dp">
27
28         <TextView
29             android:layout_width="wrap_content"
30             android:textSize="20sp"
31             android:textColor="@color/black"
32             android:layout_marginEnd="16dp"
33             android:layout_marginStart="16dp"
34             android:text="Username :"/>
35
36         <EditText
37             android:hint="Masukkan Username :"
38             android:layout_height="wrap_content"
39             android:layout_marginEnd="20dp"
40             android:layout_marginStart="20dp"
41             android:layout_weight="1"/>
42
43     </TableRow>

```

```

45      <TableRow
46          android:layout_marginTop="16dp">
47
48          <TextView
49              android:layout_width="wrap_content"
50              android:textSize="20sp"
51              android:textColor="@color/black"
52              android:layout_marginEnd="16dp"
53              android:layout_marginStart="16dp"
54              android:text="Password :"/>
55
56          <EditText
57              android:hint="Masukkan Password :"
58              android:layout_height="wrap_content"
59              android:layout_marginEnd="20dp"
60              android:layout_marginStart="20dp"
61              android:layout_weight="1"/>
62      </TableRow>

```

```

64      <TableRow
65          android:gravity="center"
66          android:layout_marginTop="16dp">
67
68          <Button
69              android:id="@+id/btn_save"
70              android:text="Submit"
71              android:clickable="true"
72              android:layout_width="wrap_content"
73              android:layout_height="wrap_content"
74              android:layout_span="2"
75              android:focusable="true"/>
76      </TableRow>
77  </TableLayout>

```

Gambar 22. Source Code Table Layout

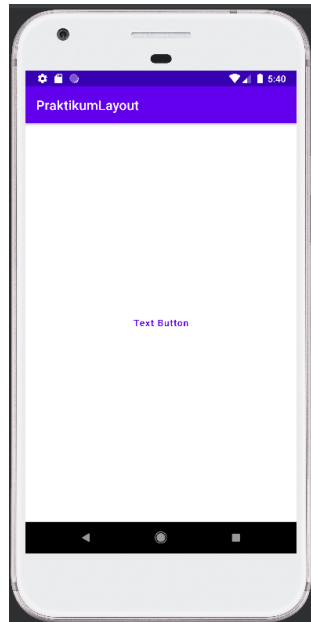
6. Material Design

Material Design merupakan *library* desain untuk OS Android yang dikembangkan oleh Google. Dirilis pertama kali untuk OS Android Versi 5 yaitu Lollipop. Dengan menggunakan *Material Design* dapat membuat tampilan *UI / UX (User Interface / User Experince)* aplikasi android terlihat lebih nyata, bertekstur, modern, dan profesional.

Ada beberapa komponen Material Design yang bisa kita integrasikan pada aplikasi android yang sedang kita buat, di antaranya :

- a. Tema Material
- b. List dan Card
- c. Shadow
- d. Animasi
- e. dan lainnya

Berikut ini *screen capture* pengaplikasian *material design* :

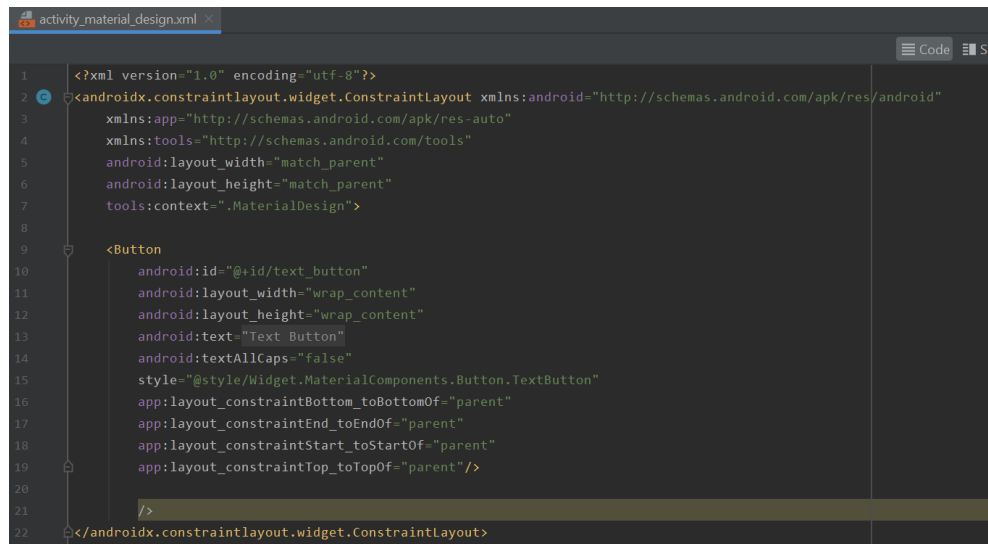


Gambar 23. Material Design

Pada gambar di atas merupakan pengaplikasian *material design* dengan menggunakan satu komponen di dalamnya, yakni komponen *Button*. Pada *material design* ini kemudian diatur dalam menambahkan beberapa Required Dependency, merubah warna dan banyak lainnya.

Berikut ini *screen capture source code material design* :

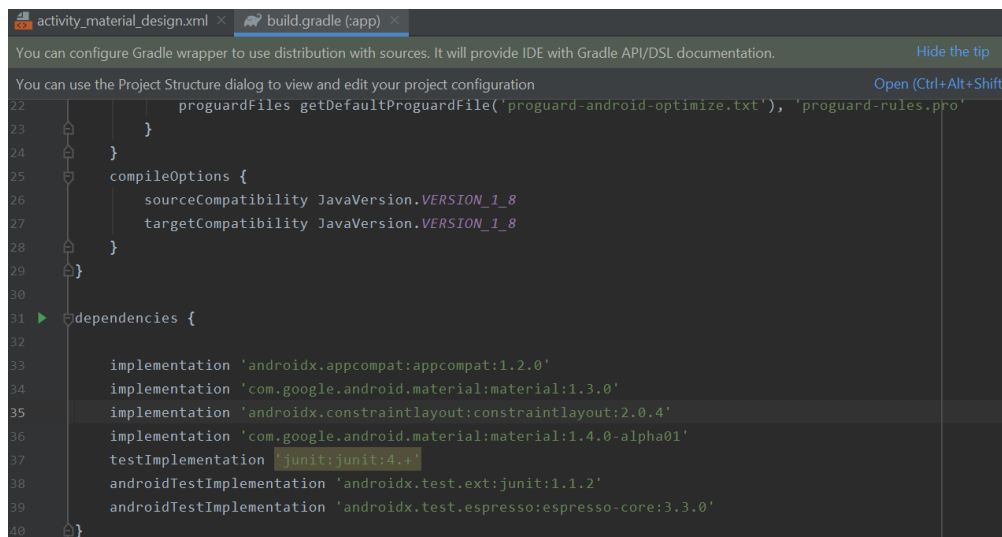
activity_material_design.xml



```

1  <?xml version="1.0" encoding="utf-8"?>
2  <androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      tools:context=".MaterialDesign">
8
9      <Button
10         android:id="@+id/text_button"
11         android:layout_width="wrap_content"
12         android:layout_height="wrap_content"
13         android:text="Text Button"
14         android:textAllCaps="false"
15         style="@style/Widget.MaterialComponents.Button.TextButton"
16         app:layout_constraintBottom_toBottomOf="parent"
17         app:layout_constraintEnd_toEndOf="parent"
18         app:layout_constraintStart_toStartOf="parent"
19         app:layout_constraintTop_toTopOf="parent"/>
20
21     />
22 </androidx.constraintlayout.widget.ConstraintLayout>

```



```

22     proguardFiles getDefaultProguardFile('proguard-android-optimize.txt'), 'proguard-rules.pro'
23 }
24 }
25 compileOptions {
26     sourceCompatibility JavaVersion.VERSION_1_8
27     targetCompatibility JavaVersion.VERSION_1_8
28 }
29 }
30 }
31 dependencies {
32     implementation 'androidx.appcompat:appcompat:1.2.0'
33     implementation 'com.google.android.material:material:1.3.0'
34     implementation 'androidx.constraintlayout:constraintlayout:2.0.4'
35     implementation 'com.google.android.material:material:1.4.0-alpha01'
36     testImplementation 'junit:junit:4.+'
37     androidTestImplementation 'androidx.test.ext:junit:1.1.2'
38     androidTestImplementation 'androidx.test.espresso:espresso-core:3.3.0'
39 }
40 }

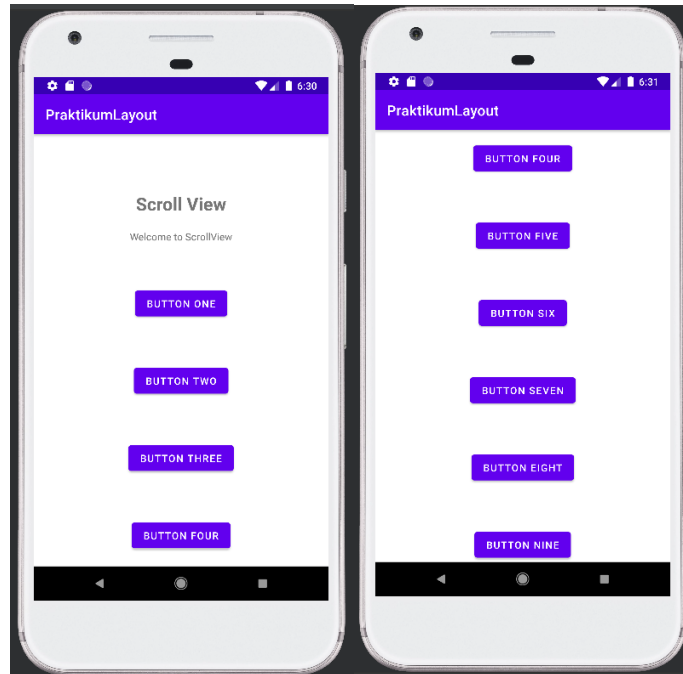
```

Gambar 24. Source Code Material Design

7. Scroll View

Scroll view digunakan untuk menampilkan *layout* yang memanjang baik ke samping ataupun ke bawah. Dengan scroll view, *layout* seolah-olah layar yang lebih luas. Pengguna dapat melakukan scroll secara horizontal maupun vertikal. Di dalam Android Studio terdapat beberapa jenis *ScrollView*, yaitu *ScrollView Vertikal*, *HorizontalScrollView* dan *Nested ScrollView* (*ScrollView* bersarang).

Berikut ini *screen capture* pengaplikasian *scroll view* :



Gambar 25. Scroll View

Pada gambar diatas merupakan pengaplikasian *scroll view* dengan melakukan *scroll* secara vertikal dengan menggunakan dua komponen yakni *TextView* dan juga *Button*. Komponen *TextView* digunakan untuk menampilkan *output* berupa teks yaitu (“*Scroll View*” dan juga “*Welcome to ScrollView*”).

Berikut ini *screen capture source code scroll view* :

activity_scroll_view.xml


```

activity_scroll_view.xml
1  <?xml version="1.0" encoding="utf-8"?>
2  <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      android:fillViewport="false"
8      tools:context=".ScrollView">
9
10     <LinearLayout
11         android:orientation="vertical"
12         android:layout_width="match_parent"
13         android:layout_height="wrap_content">
14
15         <TextView
16             android:id="@+id/loginscrn"
17             android:layout_width="wrap_content"
18             android:layout_height="wrap_content"
19             android:layout_marginTop="80dp"
20             android:text="Scroll View"
21             android:textSize="25dp"
22             android:textStyle="bold"
23             android:layout_gravity="center"/>

```

```

25     <TextView
26         android:id="@+id/fstText"
27         android:layout_width="wrap_content"
28         android:layout_height="wrap_content"
29         android:layout_marginTop="20dp"
30         android:text="Welcome to ScrollView"
31         android:layout_gravity="center"/>
32
33     <Button
34         android:layout_width="wrap_content"
35         android:layout_height="wrap_content"
36         android:layout_gravity="center"
37         android:layout_marginTop="60dp"
38         android:text="Button One" />
39
40     <Button
41         android:layout_width="wrap_content"
42         android:layout_height="wrap_content"
43         android:layout_gravity="center"
44         android:layout_marginTop="60dp"
45         android:text="Button Two" />

```

```

47      <Button
48          android:layout_width="wrap_content"
49          android:layout_height="wrap_content"
50          android:layout_gravity="center"
51          android:layout_marginTop="60dp"
52          android:text="Button Three"/>
53
54      <Button
55          android:layout_width="wrap_content"
56          android:layout_height="wrap_content"
57          android:layout_gravity="center"
58          android:layout_marginTop="60dp"
59          android:text="Button Four"/>
60
61      <Button
62          android:layout_width="wrap_content"
63          android:layout_height="wrap_content"
64          android:layout_gravity="center"
65          android:layout_marginTop="60dp"
66          android:text="Button Five"/>

```

```

67      <Button
68          android:layout_width="wrap_content"
69          android:layout_height="wrap_content"
70          android:layout_gravity="center"
71          android:layout_marginTop="60dp"
72          android:text="Button Six"/>
73
74      <Button
75          android:layout_width="wrap_content"
76          android:layout_height="wrap_content"
77          android:layout_gravity="center"
78          android:layout_marginTop="60dp"
79          android:text="Button Seven"/>
80
81      <Button
82          android:layout_width="wrap_content"
83          android:layout_height="wrap_content"
84          android:layout_gravity="center"
85          android:layout_marginTop="60dp"
86          android:text="Button Eight"/>

```

```

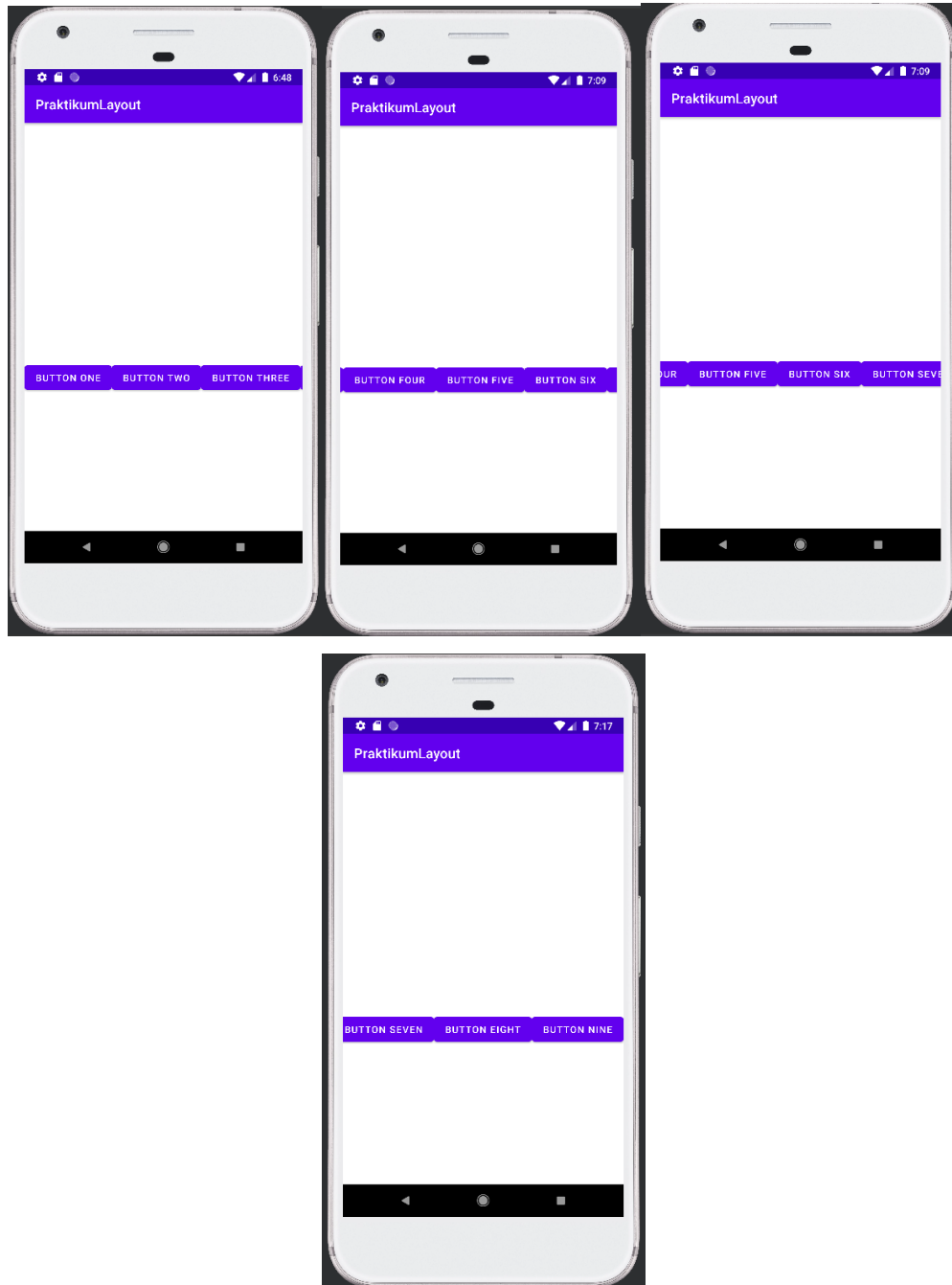
87      <Button
88          android:layout_width="wrap_content"
89          android:layout_height="wrap_content"
90          android:layout_gravity="center"
91          android:layout_marginTop="60dp"
92          android:text="Button Nine"/>
93
94
95
96
97  </LinearLayout>
98
99  </ScrollView>

```

Gambar 26. Source Code Scroll View

8. Scroll View Horizontal

Scroll View Horizontal merupakan sebuah scroll view yang dapat di scroll secara horizontal. Berikut ini *screen capture* pengaplikasian *scroll view horizontal* :



Gambar 27. Scroll View Horizontal

Pada gambar diatas merupakan pengaplikasian dari *scroll view horizontal* dengan ada beberapa komponen diantaranya yaitu *TextView* dan juga *Button*.

Berikut ini *screen capture source code scroll view horizontal* :

activity_scroll_view_horizontal.xml

```
activity_scroll_view.xml <
1  <?xml version="1.0" encoding="utf-8"?>
2  <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
3      xmlns:app="http://schemas.android.com/apk/res-auto"
4      xmlns:tools="http://schemas.android.com/tools"
5      android:layout_width="match_parent"
6      android:layout_height="match_parent"
7      android:fillViewport="false"
8      tools:context=".ScrollView">
9
10     <LinearLayout
11         android:orientation="vertical"
12         android:layout_width="match_parent"
13         android:layout_height="wrap_content">
14
15         <TextView
16             android:id="@+id/loginscrn"
17             android:layout_width="wrap_content"
18             android:layout_height="wrap_content"
19             android:layout_marginTop="80dp"
20             android:text="Scroll View"
21             android:textSize="25dp"
22             android:textStyle="bold"
23             android:layout_gravity="center"/>
24
25         <TextView
26             android:id="@+id/fstText"
27             android:layout_width="wrap_content"
28             android:layout_height="wrap_content"
29             android:layout_marginTop="20dp"
30             android:text="Welcome to ScrollView"
31             android:layout_gravity="center"/>
32
33         <Button
34             android:layout_width="wrap_content"
35             android:layout_height="wrap_content"
36             android:layout_gravity="center"
37             android:layout_marginTop="60dp"
38             android:text="Button One" />
39
40         <Button
41             android:layout_width="wrap_content"
42             android:layout_height="wrap_content"
43             android:layout_gravity="center"
44             android:layout_marginTop="60dp"
45             android:text="Button Two" />
```

```
47 <Button
48     android:layout_width="wrap_content"
49     android:layout_height="wrap_content"
50     android:layout_gravity="center"
51     android:layout_marginTop="60dp"
52     android:text="Button Three"/>
53 <Button
54     android:layout_width="wrap_content"
55     android:layout_height="wrap_content"
56     android:layout_gravity="center"
57     android:layout_marginTop="60dp"
58     android:text="Button Four"/>
59
60 <Button
61     android:layout_width="wrap_content"
62     android:layout_height="wrap_content"
63     android:layout_gravity="center"
64     android:layout_marginTop="60dp"
65     android:text="Button Five"/>
```

```
67 <Button
68     android:layout_width="wrap_content"
69     android:layout_height="wrap_content"
70     android:layout_gravity="center"
71     android:layout_marginTop="60dp"
72     android:text="Button Six"/>
73
74 <Button
75     android:layout_width="wrap_content"
76     android:layout_height="wrap_content"
77     android:layout_gravity="center"
78     android:layout_marginTop="60dp"
79     android:text="Button Seven"/>
80
81 <Button
82     android:layout_width="wrap_content"
83     android:layout_height="wrap_content"
84     android:layout_gravity="center"
85     android:layout_marginTop="60dp"
86     android:text="Button Eight"/>
```

```
88      <Button
89          android:layout_width="wrap_content"
90          android:layout_height="wrap_content"
91          android:layout_gravity="center"
92          android:layout_marginTop="60dp"
93          android:text="Button Nine"/>
94
95      </LinearLayout>
96
97  </ScrollView>
```

Gambar 28. Source Code Scroll View Horizontal