

BAB III

Android Studio #1

3.1 Tujuan

1. Praktikan mampu mengetahui lingkungan pengembangan Android
2. Praktikan memahami siklus activity pada aplikasi Android
3. Praktikan memahami penggunaan layout-layout pada Android Studio
4. Praktikan dapat memahami penggunaan intent
5. Praktikan dapat mengetahui fitur-fitur pada Android Studio
6. Praktikan dapat belajar bagaimana membuat project dengan Android Studio

3.2 Alat dan Bahan

1. PC / Notebook (minimal RAM 4GB)



Gambar 3.1 Notebook

Notebook berguna sebagai perangkat yang akan diinstallasi program dan sebagai perangkat yang digunakan untuk melakukan kegiatan praktikum.

2. Android Studio 3.2.1, JDK, Android SDK



Gambar 3.2 Android Studio

Android Studio merupakan sebuah IDE untuk Android Development yang dikenalkan pihak google pada acara Google I/O di tahun 2013. Android Studio merupakan IDE resmi untuk pengembangan aplikasi. Android Studio digunakan sebagai IDE yang akan digunakan dalam praktikum.

3. Device Android dan kabel USB



Gambar 3.3 Perangkat Android dan Kabel USB

Perangkat Android digunakan sebagai objek pengujian dalam praktikum. Kabel USB digunakan untuk menyambungkan laptop/PC dengan perangkat Android yang akan dideploykan projectnya nanti ke dalamnya.

3.3 Dasar Teori

3.3.1 Android

Android adalah sistem operasi untuk telepon seluler yang berbasis Linux. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri sehingga dapat digunakan oleh bermacam peranti bergerak.

Android tersedia secara open source bagi manufaktur perangkat keras untuk memodifikasi sesuai kebutuhan. Meskipun konfigurasi perangkat Android tidak sama antara satu dengan yang lain, namun Android sendiri mendukung beberapa fitur berikut ini:

- a. Penyimpanan (Storage) - menggunakan SQLite yang merupakan basis data relasional yang ringan untuk menyimpan data.
- b. Koneksi(Connectivity) - Mendukung GSM/EDGE, IDEN, CDMA, EVDO, UMTS, Bluetooth, WiFi, LTE, dan WiMAX.
- c. Pesan (Messaging) Mendukung SMS dan MMS
- d. Web Browser - Menggunakan open source WebKit termasuk di dalamnya engine Chrome V8 JavaScript.
- e. Media - media di dukung antara lain : H.263, H.264, MPEG-4 SP, AMR, AMR-WB, AAC, HE-AAC, MP3, MIDI, Ogg Vorbis, JPEG, PNG, GIF, dan BMP.
- f. Perangkat keras - terdapat Accelerometer Sensor, kamera, kompas digital, sensor proximity, dan GPS.
- g. Multi-touch - mendukung layar multi-touch
- h. Multi-tasking - mendukung aplikasi multi-tasking.
- i. Dukungan Flash - Android 2.3 mendukung Flash 10.1

(Sumber : Modul 2 Android Studio RC2.1.1 Praktikum MDP 2018)

3.3.2 Android Studio



Gambar 3.4 Android Studio

Android Studio adalah sebuah **IDE** untuk Android Development yang dikenalkan pihak google pada acara Google I/O di tahun 2013. Android Studio merupakan suatu pengembangan dari Eclipse IDE, dan dibuat berdasarkan IDE Java populer, yaitu **IntelliJ IDEA**. Android Studio merupakan IDE resmi untuk pengembangan aplikasi Android. Berbeda dengan Eclipse yang menggunakan Ant, Android Studio menggunakan Gradle sebagai **build environment**.

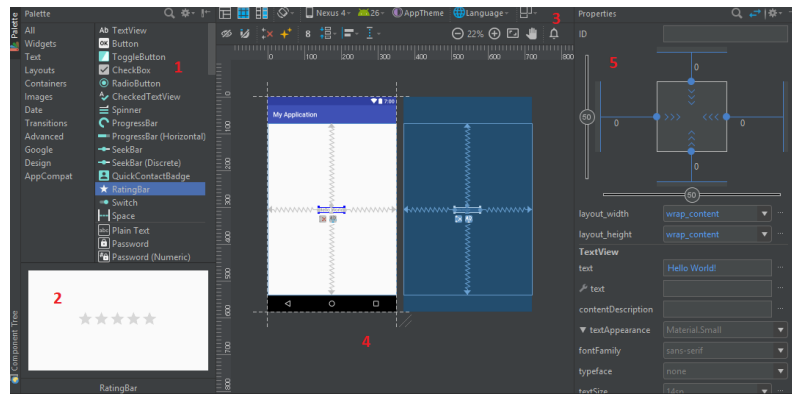
Fitur-fitur lainnya yang sudah disematkan pada Android Studio adalah sebagai berikut :

1. Menggunakan Gradle-based build system yang fleksibel.
2. Bisa mem-build multiple APK .
3. Template support untuk Google Services dan berbagai macam tipe perangkat.
4. Layout editor yang lebih bagus.
5. Built-in support untuk Google Cloud Platform, sehingga mudah untuk integrasi dengan Google Cloud Messaging dan App Engine.
6. Import library langsung dari Maven repository dan sebagainya.

(Sumber : Pengenalan Android Studio,
<http://www.jadibaru.com/android/pengenalan-android-studio-2/>)

3.3.3 Layout

Pada Android Studio layout di definisikan menggunakan bahasa XML, layout di bedakan menjadi dua yaitu design dan text. Pada design, sama halnya Frame GUI di Java, menampilkan layout untuk mengatur komponen UI pada aplikasi yang kita buat. Kita dapat men drag and drop komponen yang tersedia pada tab Palette dan mengatur ukuran komponen di properties.



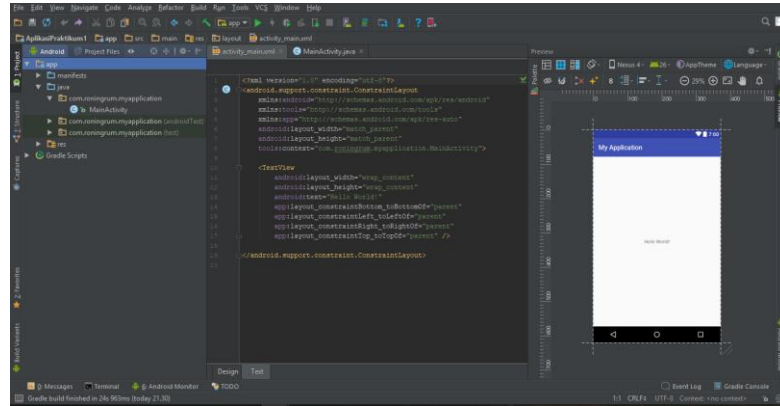
Gambar 3.5 Layout Editor Mode Design pada Android Studio

Region atau daerah pada yang ada pada layout editor design dibagi menjadi 5 area yaitu:

- Palette: Menyediakan daftar widget dan layout yang bisa Anda seret ke dalam layout di editor.
- Component Tree: Menampilkan hierarki tampilan untuk layout Anda. Klik sebuah item di sini untuk melihatnya dipilih dalam editor.
- Toolbar: Menyediakan tombol untuk mengonfigurasi penampilan layout di editor dan untuk mengedit properti layout.
- Design Editor: Menampilkan layout Anda bersama tampilan Design dan Blueprint.
- Properties: Menyediakan kontrol properti untuk tampilan yang dipilih saat ini.

Sementara pada mode text, akan menampilkan barisan kode atau barisan susunan kode dari text XML nya. dalam Android studio sudah memiliki fitur **Live**

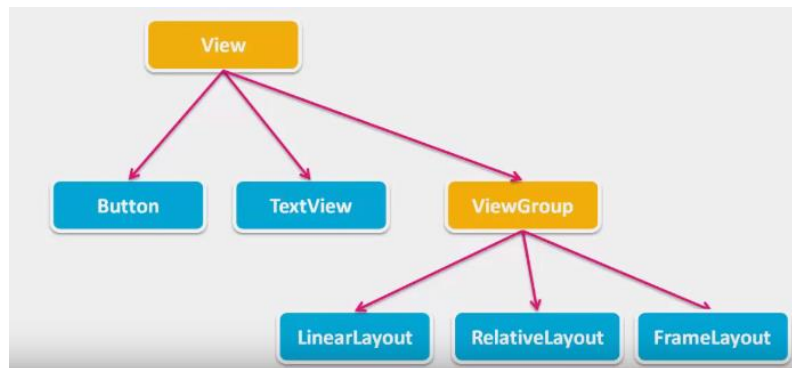
Preview, jadi dengan adanya fitur ini, kalian tidak perlu bolak-balik ke mode design hanya untuk melihat hasil kode XML yang sudah dibuat atau edit.



Gambar 3.6 Layout Editor XML Mode Text pada Android Studio

Pada Layout di android studio setiap komponen GUInya disebut View, View adalah: basic class atau kelas dasar dari setiap komponen User Interface pada Android termasuk tata letak (layout).

Terdapat sebuah subclass dalam view yaitu viewgroup, view group merupakan class yang diperuntukan untuk layout saja.



Gambar 3.7 Struktur View

Layout adalah struktur visual untuk antarmuka pengguna, seperti User Interface sebuah aktivitas atau widget aplikasi. Mendeklarasikan layout dengan dua cara:

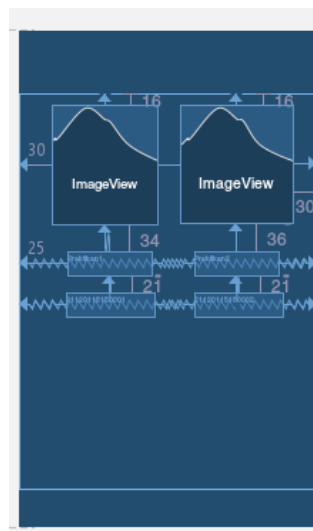
- **Deklarasikan elemen UI dalam XML.** Android menyediakan sebuah kosakata XML sederhana yang sesuai dengan kelas dan subkelas View, seperti halnya untuk widget dan layout.

- **Buat instance elemen layout saat waktu proses.** Aplikasi Anda bisa membuat objek View dan ViewGroup (dan memanipulasi propertinya) lewat program.

Pada Layout terdapat kelas – kelas yang sering digunakan yaitu

1. Constraint Layout

Constraint Layout adalah tampilan grup yang tersedia di pustaka Constraint Layout, yang telah disertakan bersama Android Studio 2.2 dan yang lebih tinggi. Layout ini dibangun dari nol bersama Layout Editor, jadi apa saja dapat diakses dari Design editor dan tidak perlu mengedit XML secara manual. Yang paling penting, sistem layout berbasis batasan ini memungkinkan untuk membuat hampir semua layout tanpa menambahkan tampilan grup apa pun.



Gambar 3.8 Contoh Constraint Layout

2. Linear Layout

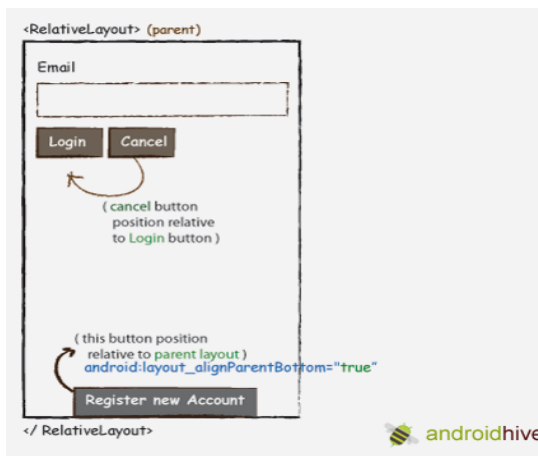
LinearLayout akan menempatkan komponen-komponen didalamnya secara horizontal atau vertical (menyamping atau menurun). LinearLayout memiliki atribut weight untuk masing-masing child view yang terdapat didalam LinearLayout yang berguna untuk mengontrol porsi ukuran view secara Relatif dalam sebuah ruang (space) yang tersedia.



Gambar 3.9 Linear Layout

3. Relative Layout

RelativeLayout adalah layout yang menampilkan elemen-elemen view dalam posisi yang relatif. Posisi dari sebuah view yang dapat diletakkan relatif terhadap posisi elemen view di sekitarnya atau relatif terhadap area layout utama. Sebuah desain tampilan aplikasi bisa dibuat lebih mudah dan sederhana menggunakan RelativeLayout dibandingkan dengan LinearLayout.

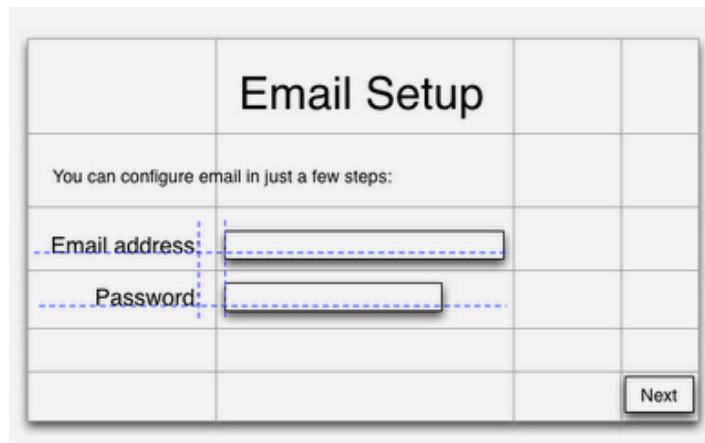


Gambar 3.10 Relative Layout

4. Grid Layout

Diperkenalkan pada API level 14 (Android 4.0 / Ice Cream Sandwich), layout ini akan memberikan kemudahan dengan mengakomodir komponen didalamnya ke

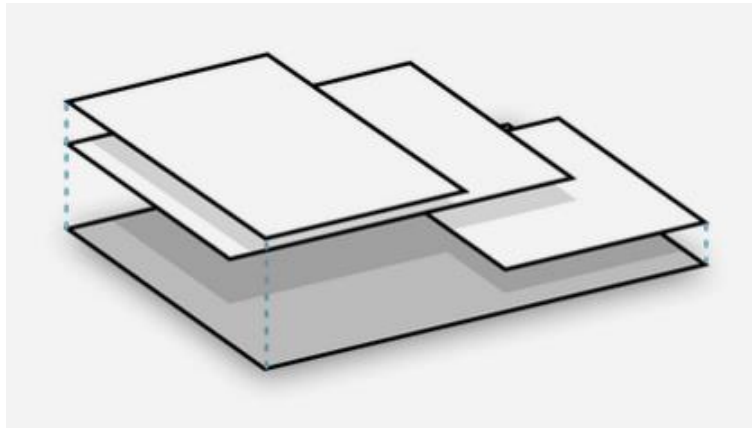
dalam bentuk Grid (Kolom dan Baris). Dalam sebuah referensi, GridLayout merupakan komponen layout yang sangat flexibel dan dapat dimanfaatkan untuk menyederhanakan pembuatan Layout UI yang bersifat kompleks dan bersarang yang terdapat di komponen Layout lainnya.



Gambar 3.11 Grid Layout

5. Frame Layout

Layout ini adalah layout yang paling sederhana. Layout ini akan membuat komponen yang ada didalamnya menjadi menumpuk atau saling menutupi satu dengan yang lainnya (*layering*). Komponen yang paling pertama pada layout ini akan berada dibawah komponen-komponen diatasnya. Pada materi penggunaan fragment di materi sebelumnya, FrameLayout memiliki kemampuan untuk menjadi container untuk fragment-fragment didalam sebuah Activity. Berikut ilustrasi dari penggunaan FrameLayout terhadap child view yang dimiliki didalamnya.



Gambar 3.12 Frame Layout

(Sumber : <https://blog.dicoding.com/android-module-5-android-layout/>)

3.3.4 Shared Preferences

Sebuah penyimpanan data yang ada di dalam Android yang bersifat data primitif pada file internal aplikasi dengan pasangan key-value.

(Sumber : Modul 2 Android Studio RC2.1.1 Praktikum MDP 2018)

3.3.5 SQLite

SQLite merupakan sistem manajemen database relasional yang dibangun dalam sebuah library bahasa pemrograman C. Berbeda dengan kebanyakan sistem manajemen database lainnya, SQLite bukanlah mesin database client-server melainkan berupa file tunggal atau beberapa file saja.

(Sumber : Modul 2 Android Studio RC2.1.1 Praktikum MDP 2018)

3.3.6 Intent

Intent adalah sebuah kelas dalam programming Android yang berfungsi untuk perpindahan halaman. Intent juga merupakan suatu objek yang terdapat dalam suatu activity dimana objek tersebut dapat komunikasi dengan activity yang lain, baik activity pada fungsi internal android misal seperti memanggil activity dalam satu package atau beda package yang masih berada dalam satu project. Intent merupakan objek tipe `android.content.Intent`. Melalui metode `startActivity()` yang digunakan untuk memulai sebuah activity lain.

Intent dibagi menjadi 2, yaitu :

1. Explicit Intent berfungsi untuk mengaktifkan komponen-komponen dalam satu aplikasi yang sama. Misalnya seperti : Berpindah Activity.
2. Implicit Intent berfungsi untuk memanggil fungsi activity yang sudah ada di fungsi internal android seperti Dial Number, Open Browser dan lainnya.

(Sumber : Modul 2 Android Studio RC2.1.1 Praktikum MDP 2018)

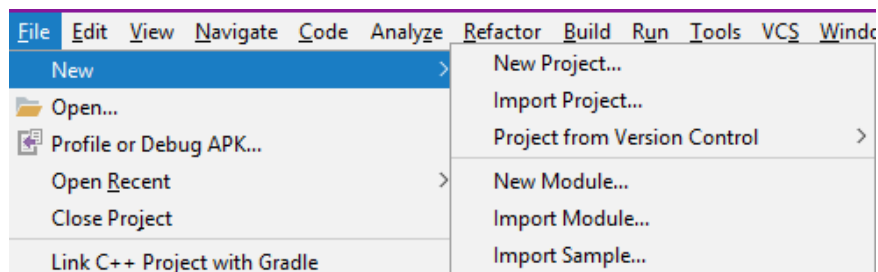
3.4 Langkah Kerja

3.4.1 Mengatur Device Android

1. Pengaturan → Opsi Pengembang → Ubah menjadi “aktif” → aktifkan debugging USB

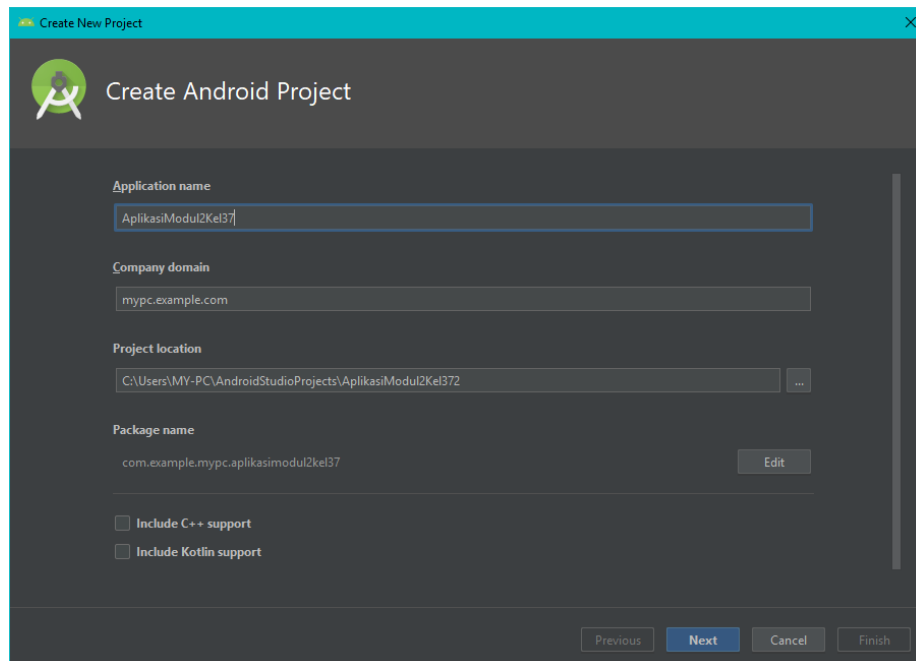
3.4.2 Membuat Project Aplikasi Android

1. Buka Android Studio
2. Klik menu File → New → New Project



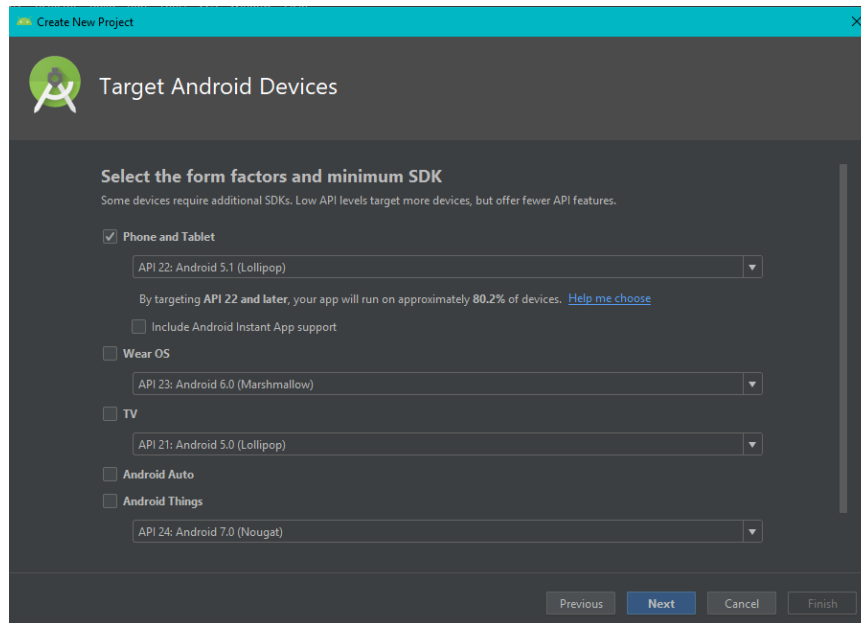
Gambar 3.13 New Project

3. Buat project baru dengan nama AplikasiModul2Kel37



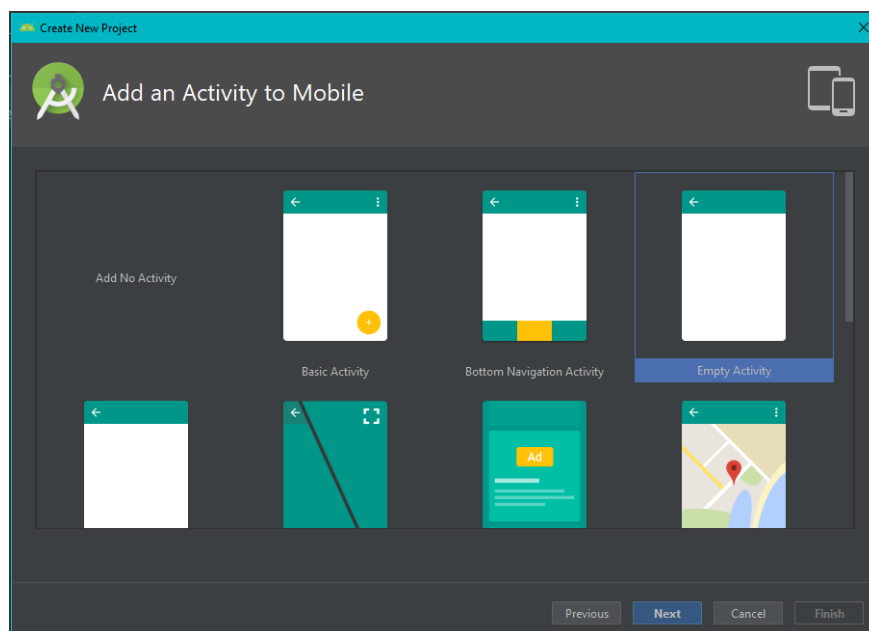
Gambar 3.14 Tab Create New Project

4. Kemudian pilih min.SDK sesuai dengan kebutuhan pengguna. Namun, pada praktikum menggunakan min SDK 22 yaitu Android Lollipop dikarenakan kebanyakan pengguna Android sekarang menggunakan versi Lollipop.



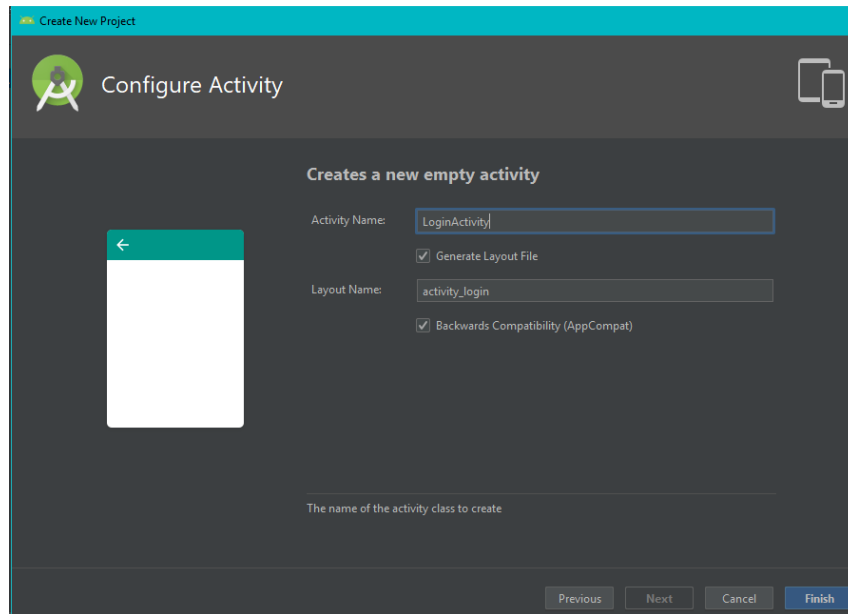
Gambar 3.15 Target Android Devices

5. Kemudian pilih jenis activity yang akan ditampilkan yaitu empty activity.

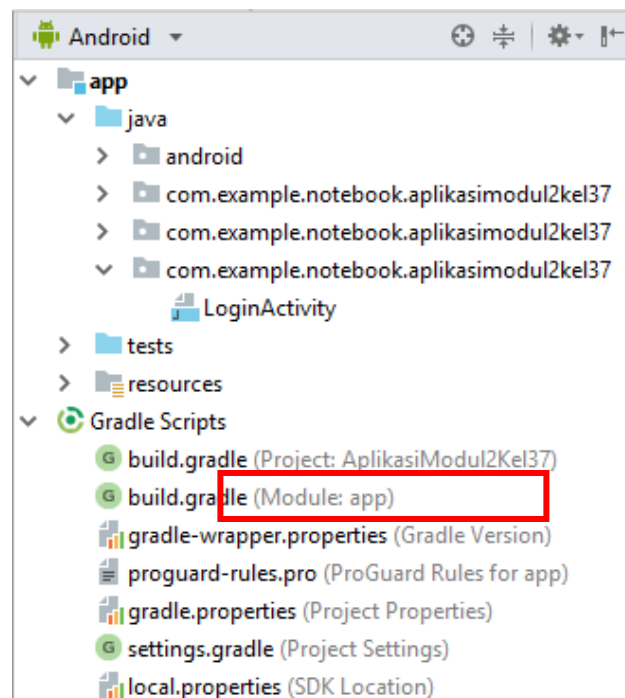


Gambar 3.16 Pilihan Activity

6. Kemudian, beri nama activity dengan nama 'Login Activity' lalu klik finish. Kemudian, tunggu sampai proses gradle selesai. Proses gradle memerlukan beberapa waktu untuk selesai. Saat selesai, akan ada build.gradle(Module: app).



Gambar 3.17 Isi nama activity



Gambar 3.18 Proses gradle selesai

7. Setelah itu kita cari build.gradle(Module:app) untuk men-copy dependency material, berikut *sourcecode* nya:

```
dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation 'com.android.support:appcompat-v7:27.1.1'
    implementation 'com.android.support:design:28.0.0'
    implementation 'com.android.support:cardview-v7:28.0.0'
    implementation 'com.android.support.constraint:constraint-
layout:1.1.3'
    testImplementation 'junit:junit:4.12'
    androidTestImplementation
'com.android.support.test:runner:1.0.2'
    androidTestImplementation
'com.android.support.test.espresso:espresso-core:3.0.2'
}
```

8. Tambahkan beberapa code di build.gradle yang telah kita *click* sebelumnya. Kurang lebih seperti pada di atas.
9. Pertama, kita akan desain tampilan colors, strings, dan styles nya untuk color string dan style xml terletak di folder res – value di panel tree pada android studio. Berikut *sourcecode* dari colors.xml :

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#82aaff</color>
    <color name="colorPrimaryDark">#4d7bcb</color>
    <color name="colorAccent">#bec4ea</color>
    <color name="white">#ffffff</color>
    <color name="button">#b6dbff</color>
</resources>
```

10. Setelah itu kita masuk ke strings.xml. Sesuai Namanya file xml ini berguna untuk menyimpan nilai string, berikut *sourcecode* nya:

```
<resources>
    <string name="app_name">Android Modul</string>
    <string name="tittle">PRAKTIKUM MDP MODUL ANDROID</string>
    <string name="regtittle">REGISTRASI DATA</string>
    <string name="user">username</string>
    <string name="password">password</string>
    <string name="login">masuk</string>
    <string name="register">register</string>
    <string name="logout">Logout</string>
    <string name="Header">Selamat datang Di Praktikum MDP
2018</string>
```



```

<string name="name">Nama Kamu</string>
<string name="description">Selanjutnya akan di jelaskan di
modul android berikutnya Mengenai API dan MVP</string>
</resources>

```

11. Kemudian di dalam `styles.xml`, di dalam `values` ini mengatur mempresentasikan *style* atau ketika dalam bahas pemrograman web, seperti file `css` (*cascading style sheet*) yang mengatur style tampilan aplikasi android studio, berikut *sourcecode* nya :

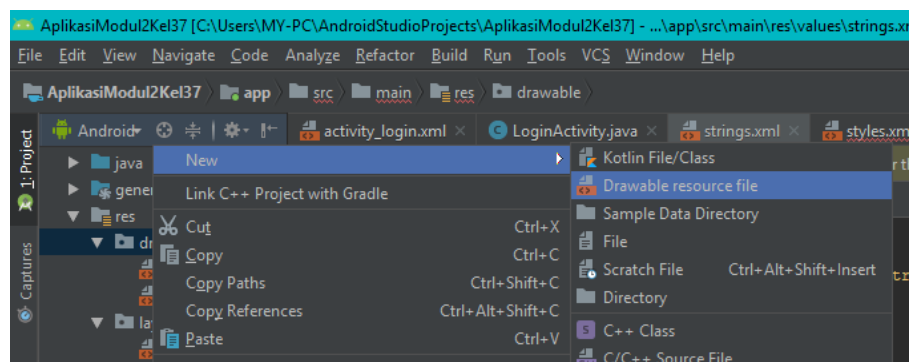
```

<resources>

    <!-- Base application theme. -->
    <style name="AppTheme"
parent="Theme.AppCompat.Light.DarkActionBar">
        <!-- Customize your theme here. -->
        <item name="colorPrimary">@color/colorPrimary</item>
        <item
name="colorPrimaryDark">@color/colorPrimaryDark</item>
        <item name="colorAccent">@color/colorAccent</item>
    </style>
</resources>

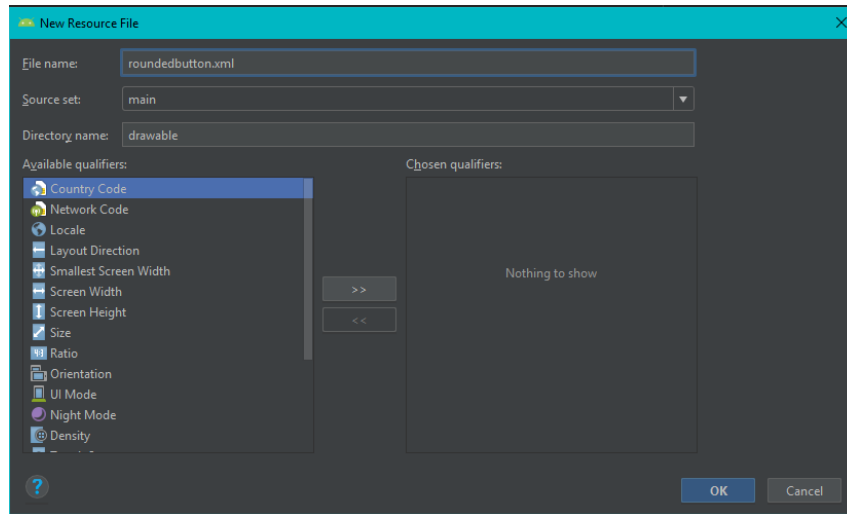
```

12. Kemudian buat drawable resource file untuk membuat desain tombol di aplikasi melengkung, dengan mengklik kanan folder `drawable` dan pilih `new` kemudian klik menu `drawable resource file`.



Gambar 3.19 Tampilan membuat drawable resource

13. Buat file dengan nama `roundedbutton.xml`

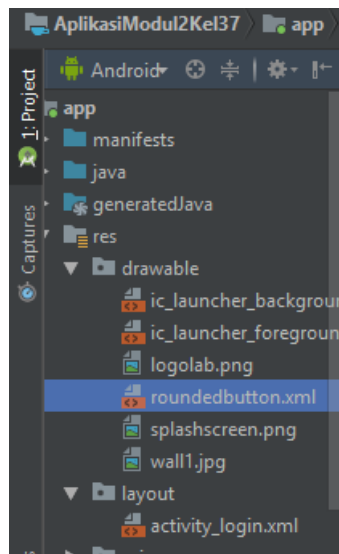


Gambar 3.20 Membuat file roundedbutton.xml

14. Buke file tersebut dan copy kan source codenya

```
<?xml version="1.0" encoding="utf-8"?>
<selector
xmlns:android="http://schemas.android.com/apk/res/android">
  <item android:state_pressed="false">
    <shape android:shape="rectangle">
      <corners android:radius="5dp" />
      <solid android:color="@color/button" />
      <stroke
        android:width="1dp"
        android:color="@color/button" />
      <padding
        android:bottom="4dp"
        android:left="4dp"
        android:right="4dp"
        android:top="4dp" />
    </shape>
  </item>
  <item android:state_pressed="true">
    <shape android:shape="rectangle">
      <corners android:radius="5dp" />
      <solid android:color="@color/button" />
      <stroke
        android:width="1dp"
        android:color="@color/button" />
      <padding
        android:bottom="4dp"
        android:left="4dp"
        android:right="4dp"
        android:top="4dp" />
    </shape>
  </item>
</selector>
```

15. Setelah mengatur di folder values dan drawable, kemudian kita akan memasukkan sebuah gambar gambar yang akan kita gunakan nantinya untuk tampilan aplikasinya. Kita masukkan logonya di folder res→drawable (drag and drop di folder ini).



Gambar 3.21 Gambar-gambar pada folder res→drawable

16. Masuk ke bagian design activity yang akan kita atur melalui folder res ->layout, yaitu `activity_login.xml`. Berikut *sourcecode* nya :

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/white"
    tools:context=".LoginActivity">

    <RelativeLayout
        android:id="@+id/rlLogin"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="60dp">

        <ImageView
            android:id="@+id/ivLogin"
            android:layout_width="100dp"
            android:layout_height="100dp"
```

```

        android:layout_gravity="center_horizontal"
        android:layout_marginLeft="130dp"
        android:src="@drawable/logolab"
        android:contentDescription="LOGO" />

<android.support.v7.widget.AppCompatTextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/tittle"
    android:textAlignment="center"
    android:textSize="20sp"
    android:textStyle="bold"
    android:layout_marginTop="120dp" />

<LinearLayout
    android:id="@+id/llLogin"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_centerInParent="true"
    android:layout_marginTop="320dp"
    android:layout_marginEnd="50dp"
    android:orientation="vertical"
    android:padding="18dp">

    <EditText
        android:id="@+id/etUsernameLogin"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="@string/user"
        android:inputType="textEmailAddress"
        android:maxLines="1"
        android:singleLine="true" />

    <EditText
        android:id="@+id/etPasswdLogin"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="30dp"
        android:hint="@string/password"
        android:imeActionId="6"
        android:imeActionLabel="Sign In"
        android:imeOptions="actionUnspecified"
        android:inputType="textPassword"
        android:maxLines="1"
        android:singleLine="true" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <Button

```

```

        android:id="@+id/btnSignUpLogin"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"

        android:background="@drawable/roundedbutton"
        android:text="@string/register"
        android:textColor="@color/white"/>

        <LinearLayout
            android:layout_width="2dp"
            android:layout_height="2dp"
            android:layout_weight="0"

        android:orientation="vertical"></LinearLayout>

        <Button
            android:id="@+id/btnSignInLogin"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"

        android:background="@drawable/roundedbutton"
            android:text="@string/login"
            android:textColor="@color/white" />
        </LinearLayout>
    </LinearLayout>
</RelativeLayout>
</ScrollView>

```

17. Maka tampilannya akan seperti gambar dibawah ini



Gambar 3.22 Layout activity_login.xml

18. Setelah itu kita masukkan *sourcecode* LoginActivity.java yang digunakan untuk masuk ke dalam aplikasi nya. Kali ini kita menggunakan SharedPreferences untuk menyimpan database username dan password yang kita masukkan. Sifat SharedPreferences hamper sama dengan kebanyakan database layaknya SQLite tapi hanya bisa menyimpan bentuk data yang sederhana dan lebih kecil.

```
import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.text.TextUtils;
import android.util.Log;
import android.util.Printer;
import android.view.KeyEvent;
import android.view.View;
import android.view.inputmethod.EditorInfo;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import java.security.PrivateKey;

public class LoginActivity extends AppCompatActivity {
    private EditText etUsernameLogin, etPasswdLogin;
```

```

private Button btnSignUpLogin, btnSignInLogin;
String username, password;
private User user;
private DatabaseHandler presenter;

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

    initView();
    initData();
    login();
    register();
}

private void initView() {
    etUsernameLogin = findViewById(R.id.etUsernameLogin);
    etPasswdLogin = findViewById(R.id.etPasswdLogin);
    btnSignInLogin = findViewById(R.id.btnSignInLogin);
    btnSignUpLogin = findViewById(R.id.btnSignUpLogin);
}

private void register() {
    btnSignUpLogin.setOnClickListener(V -> {
        Intent reg = new Intent(this,
RegisterActivity.class);
        startActivity(reg);
        finish();}
    );
}

private void initData() {
    user = new User();
    presenter = new DatabaseHandler(this);
}

private Boolean validation() {
    username = etUsernameLogin.getText().toString();
    password = etPasswdLogin.getText().toString();

    if(username.isEmpty()){
        Toast.makeText(this, "Isikan username",
Toast.LENGTH_SHORT).show();
        Log.e("Validation", "false");
        return false;
    }

    if(password.isEmpty()){
        Toast.makeText(this, "Isikan Password",
Toast.LENGTH_SHORT).show();

```

```

        Log.e("Validation","false");
        return false;
    }

    Log.e("Validation","true");
    return true;
}

private void login() {
    btnSignInLogin.setOnClickListener(v -> actLogin());
}

private void actLogin(){
    if(validation()){
        if(loginData()){
            Log.e("actLogin","true");
            initPreference();
            Intent main = new Intent(this,
MainActivity.class);
            startActivity(main);
            finish();
        } else {
            Log.e("actLogin","false");
            Toast.makeText(this, "Login gagal",
Toast.LENGTH_LONG).show();
        }
    }
}

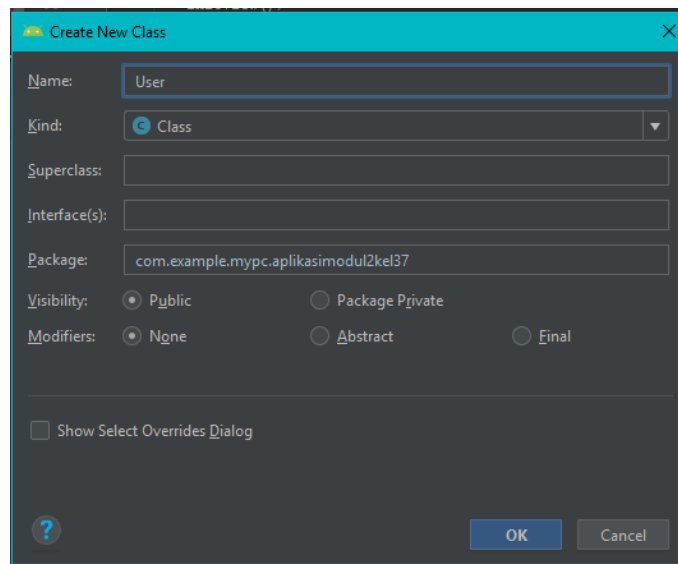
private Boolean loginData(){
    user = presenter.getLogin(username);

    if (password.equals(user.getPassword())){
        Log.e("loginData", "true");
        return true;
    } else {
        Log.e("loginData", "false");
        return false;
    }
}

private void initPreference() {
    SharedPreferences preferences =
getSharedPreferences("LoginPreference", Context.MODE_PRIVATE);
    SharedPreferences.Editor editor = preferences.edit();
    editor.putString("password", password);
    editor.putString("username", username);
    editor.commit();
    editor.apply();
}
}

```


19. Kemudian kita akan membuat sebuah class baru yang berfungsi untuk menampung segala keperluan seorang user yang bersikan username dan password.
20. Kita namani class nya “User” (Di pemograman Bahasa java penulisan Casesensitive).



Gambar 3.23 Membuat class baru User

21. Setelah membuat class nya kita masukkan *sourcecode* nya

```
package com.example.mypc.aplikasimodul2kel37;

public class User {
    int id;
    static String username;
    static String password;

    public User() {
    }

    public User(int id, String username, String password) {
        this.id = id;
        this.username = username;
        this.password = password;
    }

    public int getId() {
        return id;
    }
}
```

```

    public void setId(int id) {
        this.id = id;
    }

    public String getUsername() {
        return username;
    }

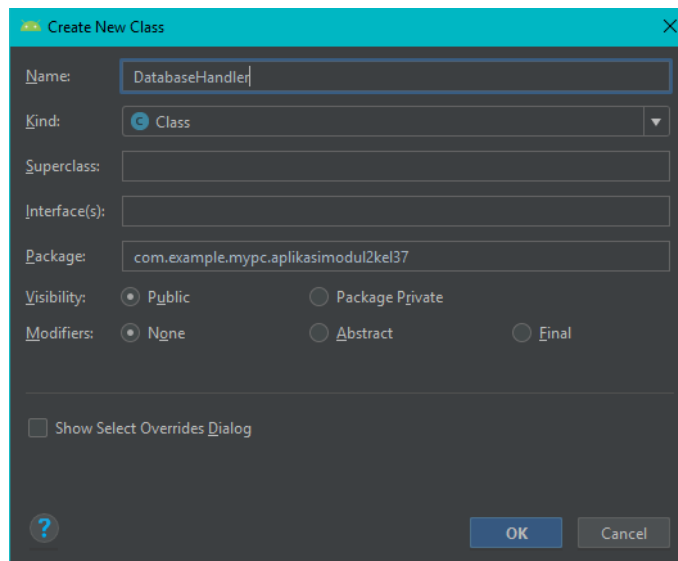
    public void setUsername(String username) {
        this.username = username;
    }

    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
        this.password = password;
    }
}

```

22. Kita membuat class baru lagi yang bernama “DatabaseHandler” yang berguna untuk operasi database create, update, update, dan delete.



Gambar 3.24 Membuat class baru DatabaseHandler

23. Kemudian, masukkan *sourcecode* DatabaseHandler nya

```

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;

```

```

import android.database.sqlite.SQLiteOpenHelper;
import android.util.Log;

import java.util.ArrayList;
import java.util.List;

public class DatabaseHandler extends SQLiteOpenHelper {

    // static variable
    private static final int DATABASE_VERSION = 1;

    // Database name
    private static final String DATABASE_NAME = "modulandroid";

    // table name
    private static final String TABLE_MAHASISWA = "mahasiswa";

    // column tables
    private static final String KEY_ID = "id";
    private static final String KEY_NAME = "username";
    private static final String KEY_TALL = "password";

    public DatabaseHandler(Context context){
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    //Create table
    @Override
    public void onCreate(SQLiteDatabase db) {
        String CREATE_MAHASISWA_TABLE = "CREATE TABLE " +
TABLE_MAHASISWA + " ("
            + KEY_ID + " INTEGER PRIMARY KEY," + KEY_NAME +
" TEXT,"
            + KEY_TALL + " TEXT" + ")";
        db.execSQL(CREATE_MAHASISWA_TABLE);
    }

    // on Upgrade database
    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion,
int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_MAHASISWA);
        onCreate(db);
    }

    public void addUser(User userModel){
        SQLiteDatabase db = getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(KEY_NAME, userModel.getUsername());
        values.put(KEY_TALL, userModel.getPassword());
        db.insert(TABLE_MAHASISWA, null, values);
        db.close();
    }
}

```

```

    }

    public User getMahasiswa(int id) {
        SQLiteDatabase db = this.getReadableDatabase();

        Cursor cursor = db.query(TABLE_MAHASISWA, new String[]
        { KEY_ID,
            KEY_NAME, KEY_TALL }, KEY_ID + "=?",
            new String[] { String.valueOf(id) }, null,
            null, null, null);
        if (cursor != null)
            cursor.moveToFirst();

        User mahasiswa = new
        User(Integer.parseInt(cursor.getString(0)),
            cursor.getString(1), cursor.getString(2));
        // return mahasiswa
        return mahasiswa;
    }

    public User getLogin(String username) {
        User mahasiswa;
        SQLiteDatabase db = this.getReadableDatabase();

        Cursor cursor = db.query(TABLE_MAHASISWA, new String[]
        { KEY_ID,
            KEY_NAME, KEY_TALL }, KEY_NAME + "=?",
            new String[] { username }, null, null, null,
            null);

        if (cursor != null) {
            cursor.moveToFirst();
        }

        int nomor = cursor.getCount();

        if(nomor>0){
            mahasiswa = new
            User(Integer.parseInt(cursor.getString(0)),
                cursor.getString(1), cursor.getString(2));
        } else {
            mahasiswa = new User(0,
                "failed", "failed");
        }

        // return mahasiswa
        return mahasiswa;
    }

    // get All Record
    public List<User> getAllRecord() {
        List<User> mahasiswaList = new ArrayList<User>();
    }

```

```

        // Select All Query
        String selectQuery = "SELECT  * FROM " +
TABLE_MAHASISWA;

        SQLiteDatabase db = this.getWritableDatabase();
        Cursor cursor = db.rawQuery(selectQuery, null);

        if (cursor.moveToFirst()) {
            do {
                User userModel = new User();
userModels.setId(Integer.parseInt(cursor.getString(0)));
                userModel.setUsername(cursor.getString(1));
                userModel.setPassword(cursor.getString(2));

                mahasiswaList.add(userModel);
            } while (cursor.moveToNext());
        }

        // return mahasiswa list
        return mahasiswaList;
    }

    public int updateMahasiswa(User mahasiswa) {
        SQLiteDatabase db = this.getWritableDatabase();

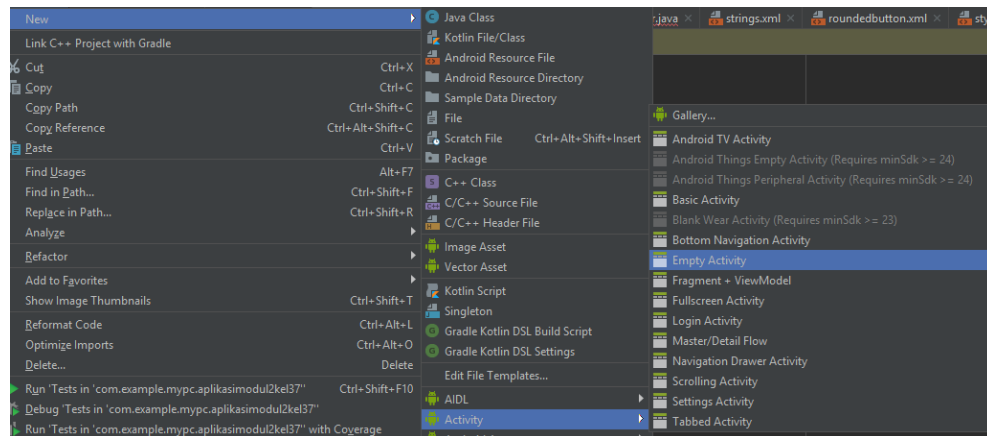
        ContentValues values = new ContentValues();
        values.put(KEY_NAME, mahasiswa.getUsername());
        values.put(KEY_TALL, mahasiswa.getPassword());

        // updating row
        return db.update(TABLE_MAHASISWA, values, KEY_ID + " =
?",
            new String[] {
String.valueOf(mahasiswa.getId()) });
    }

    public void deleteModel(User mahasiswa) {
        SQLiteDatabase db = this.getWritableDatabase();
        db.delete(TABLE_MAHASISWA, KEY_ID + " = ?",
            new String[] {
String.valueOf(mahasiswa.getId()) });
        db.close();
    }
}

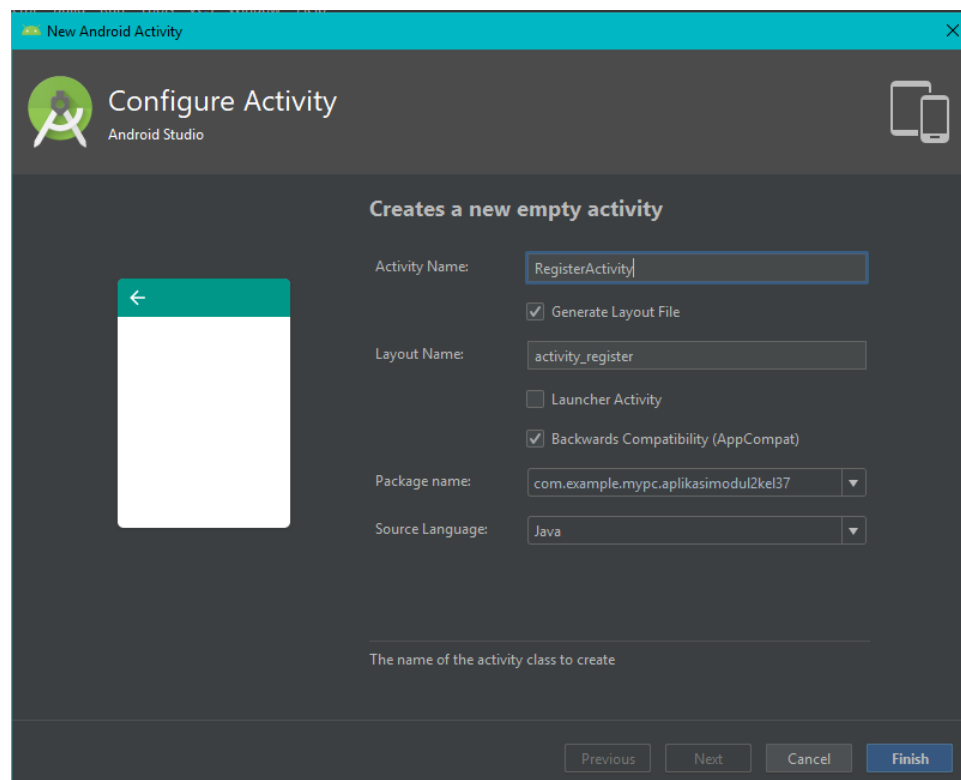
```

24. Untuk menambahkan seorang user dalam aplikasi, maka dibutuhkan suatu Register, kita akan membuat activity baru yang bernama Register Activity



Gambar 3.25 Membuat Empty Activity

25. Activity Name nya kita beri nama “RegisterActivity” dan Layout Name nya kita beri nama “activity_register” . Kemudian kalau sudah benar konfigurasi nya maka klik tombol finish untuk menyelesaikannya



Gambar 3.26 Membuat Register Activity

26. Setelah Activity nya terbuat, kita bikin design nya terlebih dahulu, melalui folder res → layout, berikut source code nya :

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="@color/white"
tools:context=".LoginActivity">

    <RelativeLayout
        android:id="@+id/rlLogin"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="60dp">

        <ImageView
            android:id="@+id/ivLogin"
            android:layout_width="100dp"
            android:layout_height="100dp"
            android:layout_gravity="center_horizontal"
            android:layout_marginLeft="130dp"
            android:src="@drawable/logolab"
            android:contentDescription="LOGO" />

        <android.support.v7.widget.AppCompatTextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="@string/regittitle"
            android:textAlignment="center"
            android:textSize="20sp"
            android:textStyle="bold"
            android:layout_marginTop="120dp" />

        <LinearLayout
            android:id="@+id/llLogin"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_centerInParent="true"
            android:layout_marginTop="330dp"
            android:layout_marginEnd="50dp"
            android:orientation="vertical"
            android:padding="18dp">

            <EditText
                android:id="@+id/etUsernameRegister"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
```

```

        android:hint="@string/user"
        android:inputType="textEmailAddress"
        android:maxLines="1"
        android:singleLine="true" />

        <EditText
            android:id="@+id/etPasswordRegister"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginBottom="30dp"
            android:hint="@string/password"
            android:imeActionId="6"
            android:imeActionLabel="Sign In"
            android:imeOptions="actionUnspecified"
            android:inputType="textPassword"
            android:maxLines="1"
            android:singleLine="true" />

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content">

            <Button
                android:id="@+id/btRegister"
                android:layout_width="0dp"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:background="@drawable/roundedbutton"
                android:text="@string/register"
                android:textColor="@color/white" />

            </LinearLayout>
        </LinearLayout>
    </RelativeLayout>
</ScrollView>

```

28. Selanjutnya kita masukkan *sourcecode* Regsiter Activity nya

```

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.widget.Button;
import android.widget.EditText;

public class RegisterActivity extends AppCompatActivity {

    private DatabaseHandler databaseHandler;
    private User userModel;
    private EditText etUsernameRegister;
    private EditText etPasswordRegister;

```



```

private Button btRegister;
private String username,password;

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

    initView();
    register();
}

private void register() {
    btRegister.setOnClickListener(V ->
        initDataHandler()
    );
}

private void initView() {
    etUsernameRegister =
findViewById(R.id.etUsernameRegister);
    etPasswordRegister =
findViewById(R.id.etPasswordRegister);
    btRegister = findViewById(R.id.btRegister);
}

private void initUser() {
    username = etUsernameRegister.getText().toString();
    password = etPasswordRegister.getText().toString();

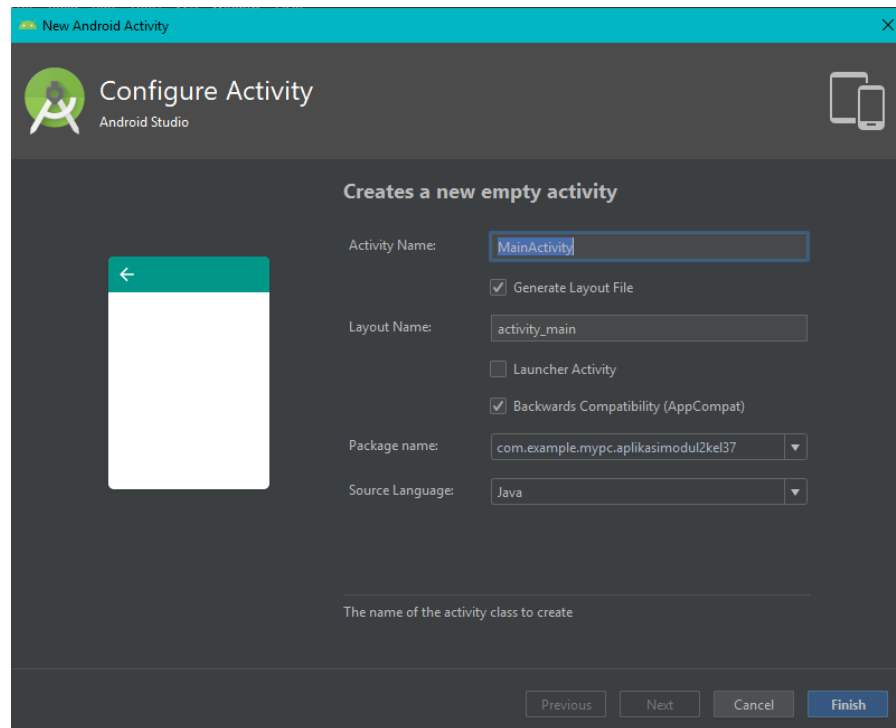
    userModel = new User();
    userModel.setUsername(username);
    userModel.setPassword(password);
}

private void initDataHandler() {
    initUser();

    databaseHandler = new DatabaseHandler(this);
    databaseHandler.addUser(usermodel);
    User model = databaseHandler.getMahasiswa(1);
    Log.e("record", model.getUsername().toString());
    Intent admin = new Intent(RegisterActivity.this,
LoginActivity.class);
    startActivity(admin);
    finish();
}
}

```

29. Activity login dan register sudah kita buat, maka kita akan membuat Activity baru lagi, tahap membuat activity baru seperti langkah sebelumnya. Activity Name nya kita beri nama “MainActivity” dan Layout Name nya kita beri nama “activity_main” harap diperhatikan besar kecilnya penulisan nama.



Gambar 3.27 Membuat MainActivity

30. Masuk ke layout activity_main.xml, kemudian masukkan source code nya :

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <android.support.v7.widget.CardView
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:id="@+id/cardView"
        style="@style/CardView.Light"
        android:layout_width="320dp"
        android:layout_height="400dp"
        android:layout_marginBottom="8dp"
        app:layout_constraintBottom_toBottomOf="parent"
```

```

        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.100000024">

        <android.support.constraint.ConstraintLayout
xmlns:app="http://schemas.android.com/apk/res-auto"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

            <ImageView
                android:id="@+id/media_image"
                android:layout_width="0dp"
                android:layout_height="0dp"
                android:scaleType="fitCenter"
                android:src="@drawable/wall1"

app:layout_constraintBottom_toTopOf="@+id/primary_text"
                app:layout_constraintDimensionRatio="H,16:9"
                app:layout_constraintEnd_toEndOf="parent"
                app:layout_constraintStart_toStartOf="parent"
                app:layout_constraintTop_toTopOf="parent"

app:layout_constraintVertical_chainStyle="packed" />

            <TextView
                android:id="@+id/primary_text"
                android:layout_width="0dp"
                android:layout_height="61dp"
                android:layout_marginStart="16dp"
                android:layout_marginTop="24dp"
                android:layout_marginEnd="16dp"
                android:text="@string/Header"

android:textAppearance="@style/TextAppearance.AppCompat.Headline"
                "
                android:textColor="#DE000000"

app:layout_constraintBottom_toTopOf="@+id/tvNameMain"
                app:layout_constraintEnd_toEndOf="parent"
                app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toBottomOf="@+id/media_image" />

            <TextView
                android:id="@+id/tvNameMain"
                android:layout_width="0dp"
                android:layout_height="wrap_content"
                android:layout_marginStart="16dp"
                android:layout_marginTop="0dp"
                android:layout_marginEnd="16dp"
                android:text="@string/name"

```

```

        android:textAppearance="@style/TextAppearance.AppCompat.Body1"
        android:textColor="#8A000000"
        android:textSize="15sp"
        app:layout_constraintBottom_toTopOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"

    app:layout_constraintTop_toBottomOf="@+id/primary_text" />

    <TextView
        android:id="@+id/tvdescription"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_marginTop="32dp"
        android:layout_marginBottom="8dp"
        android:text="@string/description"

        android:textAppearance="@style/TextAppearance.AppCompat.Body1"
        android:textColor="#8A000000"
        android:textSize="20sp"
        app:layout_constraintBottom_toBottomOf="parent"

    app:layout_constraintEnd_toEndOf="@+id/tvNameMain"
        app:layout_constraintHorizontal_bias="0.0"

    app:layout_constraintStart_toStartOf="@+id/tvNameMain"

    app:layout_constraintTop_toTopOf="@+id/tvNameMain"
        app:layout_constraintVertical_bias="1.0" />

    </android.support.constraint.ConstraintLayout>

</android.support.v7.widget.CardView>

<Button
    android:id="@+id/btnLogoutMain"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="8dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="6dp"
    android:layout_marginBottom="8dp"
    android:background="@drawable/roundedbutton"
    android:minWidth="0dp"
    android:paddingStart="8dp"
    android:paddingEnd="8dp"
    android:text="@string/logout"
    android:textColor="#de101010"

```

```

        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="@+id/cardView"
        app:layout_constraintTop_toBottomOf="@+id/cardView"
        app:layout_constraintVertical_bias="0.753" />
    </android.support.constraint.ConstraintLayout>

```

31. Setelah layout sudah kita buat, maka kita masuk ke class MainActivity.java, dan masukkan source code nya :

```

import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.SharedPreferences;
import android.support.v7.app.AlertDialog;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    private DatabaseHandler databaseHandler;
    private User userModel;
    private TextView tvNameMain;
    private Button btnLogoutMain;
    private Button exit;

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

        initView();
        initPreference();
        logout();
    }

    @Override
    public void onBackPressed() {
        showAlertDialog();
    }

    private void initView() {
        tvNameMain = findViewById(R.id.tvNameMain);
        btnLogoutMain = findViewById(R.id.btnLogoutMain);
    }
}

```

```

        exit = findViewById(R.id.btnLogoutMain);
    }

    private void initPreference() {
        SharedPreferences preferences =
        getSharedPreferences("LoginPreference", MODE_PRIVATE);
        String username = preferences.getString("username", "");

        tvNameMain.setText(username);
    }

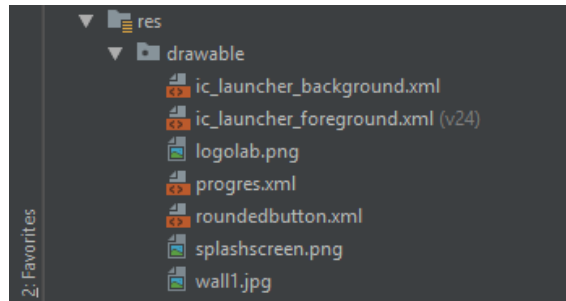
    private void deletePreference() {
        SharedPreferences preferences =
        getSharedPreferences("LoginPreference", MODE_PRIVATE);
        preferences.edit().remove("username").commit();
        preferences.edit().remove("password").commit();
    }

    private void logout() {
        exit.setOnClickListener(view -> showAlertDialog());
    }

    public void showAlertDialog() {
        new AlertDialog.Builder(this)
            .setMessage("Apa kalian ingin Logout?")
            .setCancelable(false)
            .setPositiveButton("Yes", new
        DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog,
        int id) {
                deletePreference();
                Intent login = new
        Intent(MainActivity.this, LoginActivity.class);
                startActivity(login);
                finish();
            }
        })
            .setNegativeButton("No", null)
            .show();
    }
}

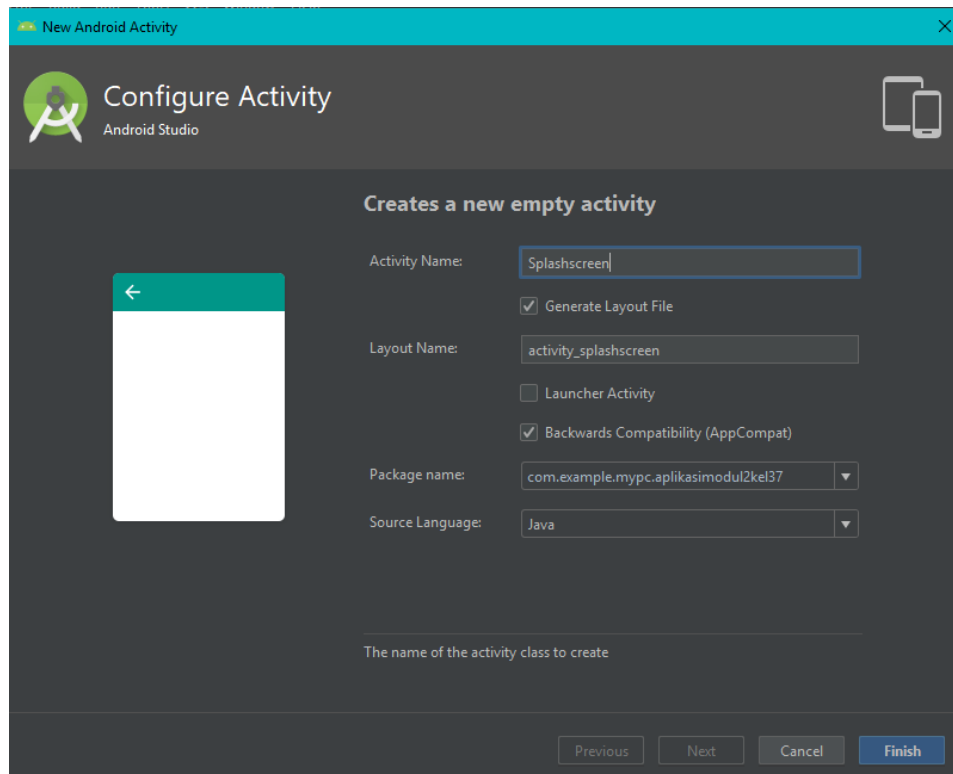
```

32. Kita akan membuat Splashscreen pada tampilan awal aplikasi saat dijalankan, masukkan gambar / icon yang akan digunakan dan menggunakan type .png agar dapat dibaca oleh system. Lakukan drag and drop pada folder berikut res→ drawable.



Gambar 3.28 File gambar splashscreen di res-drawable

33. Langkah selanjutnya, membuat activity baru seperti langkah sebelumnya yang sudah kita buat, dengan nama Activity nya “Splashscreen” dan nama Layout nya “activity_splashscreen”



Gambar 3.29 Membuat Activity Splashscreen

34. Masuk ke bagian layout activiy_splashscreen.xml, dan kemudian masukkan source code nya :

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

```

        android:layout_width="match_parent"
        android:layout_height="match_parent" >

        <ImageView
            android:src="@drawable/splashscreen"
            android:layout_width="fill_parent"
            android:layout_height="fill_parent"
            android:scaleType="fitXY"/>

        <ProgressBar
            android:id="@+id/progressBar1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_alignParentBottom="true"
            android:layout_centerHorizontal="true"
            android:layout_marginBottom="58dp" />

    </RelativeLayout>

```

35. Maka akan muncul tampilannya kurang lebih seperti ini



Gambar 3.30 Tampilan Splashscreen

36. Kemudian masukkan source code Splashscreen.java nya

```
import android.content.SharedPreferences;
import android.support.v7.app.AppCompatActivity;
import android.content.Intent;
import android.os.Bundle;
import android.os.Handler;
import android.view.Window;
import android.view.WindowManager;

public class Splashscreen extends AppCompatActivity {

    private static int splashInterval = 2000;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        this.requestWindowFeature(Window.FEATURE_NO_TITLE);

        getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,
            WindowManager.LayoutParams.FLAG_FULLSCREEN);

        setContentView(R.layout.activity_splashscreen);

        handler();
    }

    private void handler(){
        new Handler().postDelayed(() -> {
            if (initPreference()){
                Intent main = new Intent(Splashscreen.this,
MainActivity.class);
                startActivity(main);
                finish();
            } else {
                Intent i = new Intent(Splashscreen.this,
LoginActivity.class);
                startActivity(i);
                finish();
            }
        }, splashInterval);
    }

    private Boolean initPreference() {
        SharedPreferences preferences =
getSharedPreferences("LoginPreference", MODE_PRIVATE);
        String username = preferences.getString("username",
        "");

        if (username.isEmpty()){
            return false;
        }
    }
}
```

```

        return true;
    }
}

```

37. Setelah itu, kita masuk ke dalam folder Manifest nya, melalui app→Manifest→AndroidManifest.xml. Kemudian masukkan *sourcecode* nya :

```

<?xml version="1.0" encoding="utf-8"?>
<manifest
xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.mypc.aplikasimodul2kel37">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">

        <!-- Splash screen -->

        <activity
            android:name="
com.example.mypc.aplikasimodul2kel37.Splashscreen"
            android:label="@string/app_name"
            android:theme="@style/Splashscreen"
            android:screenOrientation="portrait"
            android:noHistory="true">
            <intent-filter>
                <action
android:name="android.intent.action.MAIN" />

                <category
android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".RegisterActivity" />
        <activity android:name=".MainActivity" />
        <activity android:name=".LoginActivity"></activity>
    </application>

</manifest>

```

38. Tambahkan beberapa *sourcecode* yang ada di styles.xml untuk mengatur tampilan Splash Screen nya. Berikut di bawah *sourcecode*-nya:

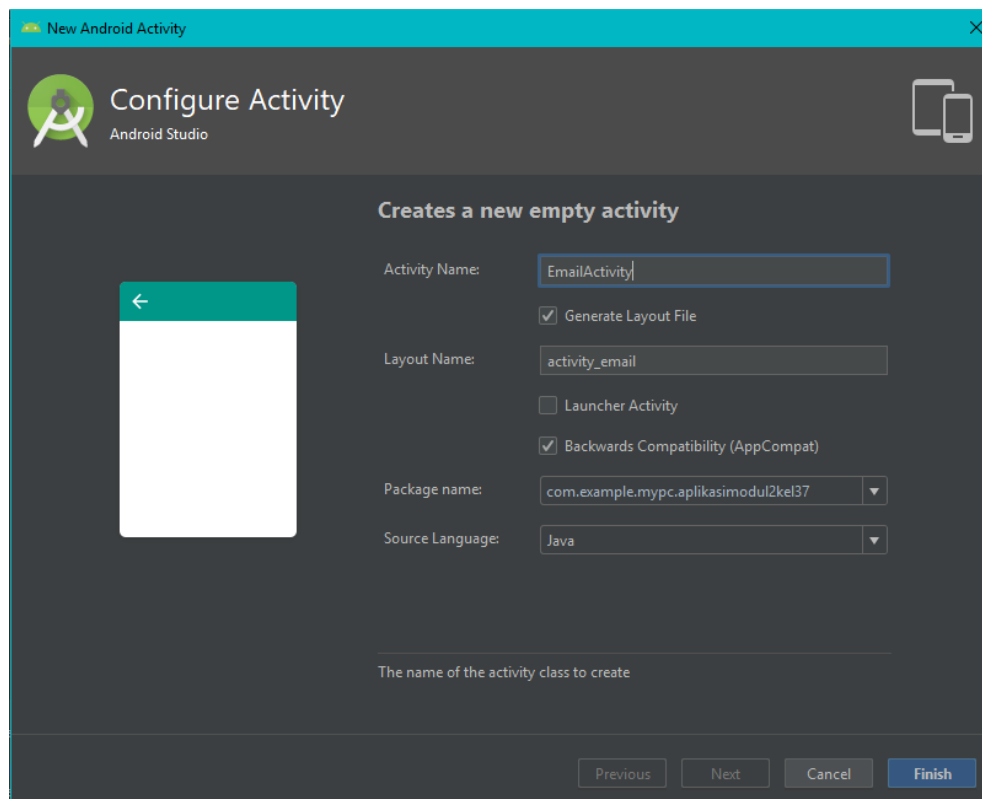
```

<style name="Splashscreen"

```

```
parent="Theme.AppCompat.Light.DarkActionBar">
    <item name="windowActionBar">false</item>
    <item name="windowNoTitle">true</item>
</style>
```

39. Berikutnya kita membuat sebuah *activity* bernama EmailActivity dengan nama layoutnya activity_email



Gambar 3.31 Membuat EmailActivity

40. Kemudian, masukkan *sourcecode* pada EmailActivity.java seperti di bawah:

```
import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class EmailActivity extends AppCompatActivity {

    Button buttonSend;
    EditText textTo;
```

```

EditText textSbj;
EditText textMsg;

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

    buttonSend = (Button) findViewById(R.id.buttonSend);
    textTo = (EditText) findViewById (R.id.editTextTo);
    textSbj = (EditText) findViewById (R.id.editTextSbj);
    textMsg = (EditText) findViewById (R.id.editTextMsg);

    buttonSend.setOnClickListener(new View.OnClickListener()
    {
        @Override
        public void onClick(View view) {

            String to = textTo.getText().toString();
            String subject = textSbj.getText().toString();
            String message = textMsg.getText().toString();

            Intent email = new Intent(Intent.ACTION_SEND);
            email.putExtra(Intent.EXTRA_EMAIL, new String[]{
to});

            email.putExtra(Intent.EXTRA_SUBJECT, subject);
            email.putExtra(Intent.EXTRA_TEXT, message);

            email.setType("message/rfc822");

            startActivity(Intent.createChooser(email,
"Choose an Email client :"));
        }
    });

    @Override
    public void onBackPressed() {
        back();
    }

    private void back() {
        Intent reg = new Intent(this, MainActivity.class);
        startActivity(reg);
        finish();
    }
}

```

41. Berikutnya masukkan *sourcecode* pada *activity_mail.xml*

```
<?xml version="1.0" encoding="utf-8"?>
```

```

<android.support.constraint.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".EmailActivity">

    <EditText
        android:id="@+id/editTextSbj"
        android:layout_width="0dp"
        android:layout_height="44dp"
        android:layout_marginStart="30dp"
        android:ems="10"
        android:inputType="textEmailSubject"
        android:hint="Subject "
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        tools:layout_constraintLeft_creator="1"
        tools:layout_constraintTop_creator="1"
        tools:layout_constraintRight_creator="1"
        android:layout_marginEnd="30dp"
        android:layout_marginTop="25dp"
        app:layout_constraintTop_toBottomOf="@+id/editTextTo"
    />

    <EditText
        android:id="@+id/editTextTo"
        android:layout_width="0dp"
        android:layout_height="44dp"
        android:hint="To "
        android:fontFamily="@string/app_name"
        android:inputType="textEmailAddress"
        tools:layout_constraintTop_creator="1"
        android:layout_marginTop="25dp"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        tools:layout_constraintRight_creator="1"
        android:layout_marginStart="30dp"
        android:layout_marginEnd="30dp"
        tools:layout_constraintLeft_creator="1"
        app:layout_constraintTop_toTopOf="parent" />

    <EditText
        android:id="@+id/editTextMsg"
        android:layout_width="291dp"
        android:layout_height="176dp"
        android:layout_marginStart="30dp"
        android:ems="10"
        android:inputType="textMultiLine"

```

```

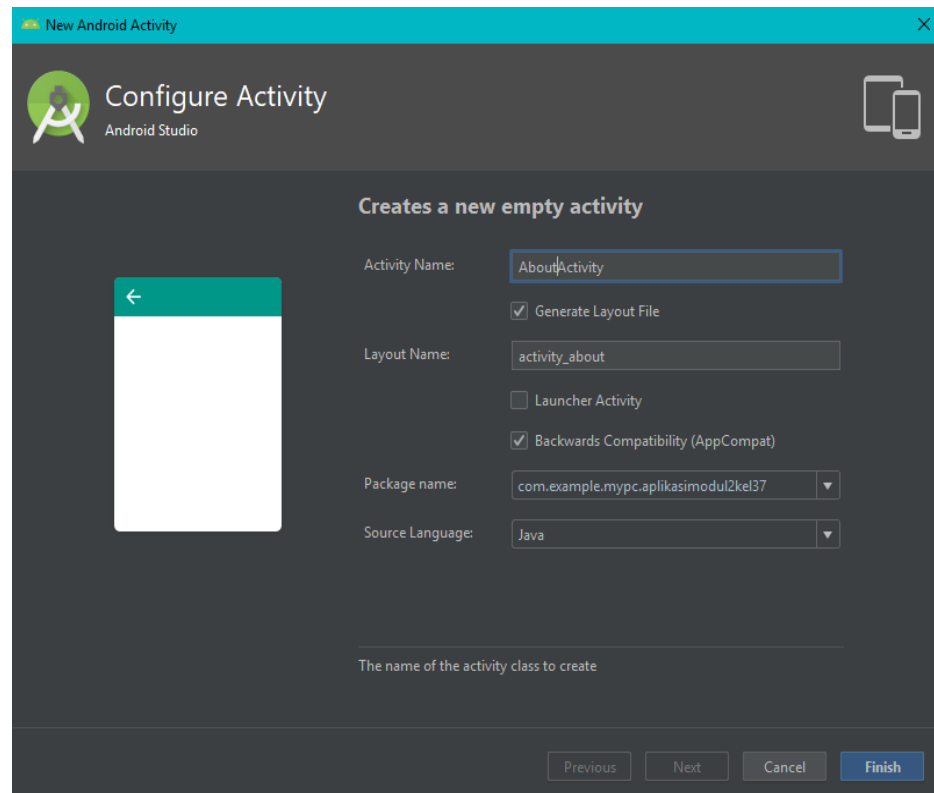
        android:hint="Message "
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        tools:layout_constraintLeft_creator="1"
        tools:layout_constraintTop_creator="1"
        tools:layout_constraintRight_creator="1"
        android:layout_marginEnd="30dp"
        android:layout_marginTop="25dp"
        app:layout_constraintTop_toBottomOf="@+id/editTextSbj"
        app:layout_constraintHorizontal_bias="0.516"
        android:layout_marginLeft="16dp"
        android:layout_marginRight="16dp"
        tools:layout_editor_absoluteX="62dp" />

<Button
    android:id="@+id/buttonSend"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginEnd="30dp"
    android:layout_marginLeft="30dp"
    android:layout_marginRight="30dp"
    android:layout_marginStart="30dp"
    android:background="#26c6da"
    android:text="Send "
    android:textColor="@android:color/background_light"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    tools:layout_constraintBottom_creator="1"
    tools:layout_constraintLeft_creator="1"
    tools:layout_constraintRight_creator="1"
    tools:layout_constraintTop_creator="1"
    android:layout_marginTop="25dp"
    app:layout_constraintTop_toBottomOf="@+id/editTextMsg"
    app:layout_constraintVertical_bias="0.251" />

</android.support.constraint.ConstraintLayout>

```

42. Selanjutnya kita membuat *activity* baru lagi bernama AboutActivity dengan nama layoutnya activity_about



Gambar 3.32 Membuat AboutActivity

43. *Sourcecode* untuk AboutActivity.java

```
import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;

public class AboutActivity extends AppCompatActivity {

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

    @Override
    public void onBackPressed() {
        back();
    }

    private void back() {
        Intent reg = new Intent(this, MainActivity.class);
        startActivity(reg);
        finish();
    }

}
```

44. Selanjutnya, masukkan *sourcecode* pada *activity_about.xml* seperti di bawah:

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="
com.example.mypc.aplikasimodul2kel37.AboutActivity">

    <ImageView
        android:id="@+id/ivpraktikan1"
        android:layout_width="126dp"
        android:layout_height="145dp"
        android:layout_marginStart="8dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="8dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:srcCompat="@drawable/logolab" />

    <android.support.v7.widget.CardView
xmlns:android="http://schemas.android.com/apk/res/android"
        android:id="@+id/cardView"
        style="@style/CardView.Light"
        android:layout_width="320dp"
        android:layout_height="80dp"
        android:layout_marginTop="8dp"
        android:layout_marginBottom="8dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toBottomOf="@+id/ivpraktikan1"
        app:layout_constraintVertical_bias="0.0">

        <android.support.constraint.ConstraintLayout
xmlns:app="http://schemas.android.com/apk/res-auto"
            android:layout_width="match_parent"
            android:layout_height="wrap_content">

            <TextView
                android:id="@+id/twnama1"
                android:layout_width="97dp"
                android:layout_height="24dp"
                android:layout_marginLeft="24dp"
                android:layout_marginTop="10dp"
                android:layout_marginEnd="196dp"
                android:text="Nama kamu"
```



```

        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

        <TextView
            android:id="@+id/twnim1"
            android:layout_width="109dp"
            android:layout_height="26dp"
            android:layout_marginTop="12dp"
            android:text="Nim kamu"

app:layout_constraintStart_toStartOf="@+id/twnama1"
app:layout_constraintTop_toBottomOf="@+id/twnama1" />

        </android.support.constraint.ConstraintLayout>

    </android.support.v7.widget.CardView>

    <ImageView
        android:id="@+id/ivpraktikan2"
        android:layout_width="126dp"
        android:layout_height="145dp"
        android:layout_marginTop="16dp"
        app:layout_constraintEnd_toEndOf="@+id/ivpraktikan1"
        app:layout_constraintHorizontal_bias="0.0"

app:layout_constraintStart_toStartOf="@+id/ivpraktikan1"
        app:layout_constraintTop_toBottomOf="@+id/cardView"
        app:srcCompat="@drawable/logolab" />

    <android.support.v7.widget.CardView
        android:id="@+id/cardView2"
        style="@style/CardView.Light"
        android:layout_width="320dp"
        android:layout_height="80dp"
        android:layout_marginTop="8dp"
        android:layout_marginBottom="8dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="@+id/cardView"
        app:layout_constraintStart_toStartOf="@+id/cardView"

app:layout_constraintTop_toBottomOf="@+id/ivpraktikan2">

        <android.support.constraint.ConstraintLayout
            xmlns:app="http://schemas.android.com/apk/res-auto"
            android:layout_width="match_parent"
            android:layout_height="wrap_content">

            <TextView
                android:id="@+id/twnama2"

```

```

        android:layout_width="97dp"
        android:layout_height="24dp"
        android:layout_marginLeft="24dp"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="196dp"
        android:text="Nama kamu"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

        <TextView
            android:id="@+id/twnim2"
            android:layout_width="115dp"
            android:layout_height="32dp"
            android:layout_marginTop="8dp"
            android:text="Nim kamu"

            app:layout_constraintStart_toStartOf="@+id/twnama2"

            app:layout_constraintTop_toBottomOf="@+id/twnama2" />

        </android.support.constraint.ConstraintLayout>

    </android.support.v7.widget.CardView>

</android.support.constraint.ConstraintLayout>

```

45. Selanjutnya kita diharuskan menambah method pada MainActivity.java, berikut di bawah ini adaah *sourcecode*-nya:

```

private void email() {
    btnEmail.setOnClickListener(v -> {
        Intent reg = new Intent(this, EmailActivity.class);
        startActivity(reg);
        finish();
    });
}

private void about() {
    btnAboutMain.setOnClickListener(v -> {
        Intent reg = new Intent(this, AboutActivity.class);
        startActivity(reg);
        finish();
    });
}

```

3.5 Analisa Hasil Percobaan

3.5.1 Membuat LoginActivity



Gambar 3.33 Tampilan LoginActivity

Pada percobaan pertama halaman login akan terlihat seperti pada gambar di atas dengan *Source code* activity_login.xml di bawah ini:

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/white"
    tools:context=".LoginActivity">

    <RelativeLayout
        android:id="@+id/rlLogin"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="60dp">

        <ImageView
            android:id="@+id/ivLogin"
            android:layout_width="100dp"
            android:layout_height="100dp"
            android:layout_gravity="center_horizontal"
            android:layout_marginLeft="130dp"
            android:src="@drawable/logolab"
            android:contentDescription="LOGO" />
```

```

<android.support.v7.widget.AppCompatTextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="@string/tittle"
    android:textAlignment="center"
    android:textSize="20sp"
    android:textStyle="bold"
    android:layout_marginTop="120dp" />

<LinearLayout
    android:id="@+id/llLogin"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_centerInParent="true"
    android:layout_marginTop="320dp"
    android:layout_marginEnd="50dp"
    android:orientation="vertical"
    android:padding="18dp">

    <EditText
        android:id="@+id/etUsernameLogin"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="@string/user"
        android:inputType="textEmailAddress"
        android:maxLines="1"
        android:singleLine="true" />

    <EditText
        android:id="@+id/etPasswdLogin"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="30dp"
        android:hint="@string/password"
        android:imeActionId="6"
        android:imeActionLabel="Sign In"
        android:imeOptions="actionUnspecified"
        android:inputType="textPassword"
        android:maxLines="1"
        android:singleLine="true" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <Button
            android:id="@+id/btnSignUpLogin"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"

```

```

        android:background="@drawable/roundedbutton"
        android:text="@string/register"
        android:textColor="@color/white"/>

        <LinearLayout
            android:layout_width="2dp"
            android:layout_height="2dp"
            android:layout_weight="0"

android:orientation="vertical"></LinearLayout>

        <Button
            android:id="@+id/btnSignInLogin"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:background="@drawable/roundedbutton"
            android:text="@string/login"
            android:textColor="@color/white" />
    </LinearLayout>
</LinearLayout>
</RelativeLayout>
</ScrollView>

```

Source code diatas digunakan untuk membuat tampilan pada login dengan Bahasa XML. Tampilannya terdiri dari *image*, *button*, *textfield*. Ada 2 cara yang dapat dilakukan dalam proses merancang tampilan dengan Android Studio yaitu, dengan melakukan *coding* keseluruhan rancangan yang diinginkan atau dengan metode *drag and drop*. Pada *activity* ini, dengan menggunakan kodingan yang berisi kodingan awalnya `<?xml version="1.0" encoding="utf-8"?>` merupakan inisialisai awal XML dan versi yang digunakan yaitu 1.0 dan baris ini optional bisa ditulis atau tidak, jika iya maka harus diletakkan diawal dokumen. Kemudian *tag* `ScrollView` dipakai supaya user bisa melakukan scroll secara vertikal pada program. *Tag* ini merupakan *tag root* dari XML. Kemudian *tag* selanjutnya `RelativeLayout` yang berguna untuk pengaturan layout pada android dimana setiap komponen dapat memposisikan dirinya dengan bergantung pada komponen lain. Selanjutnya *Tag* `ImageView` ini berfungsi untuk menampilkan gambar, dimana lebar gambar diatur di `android:layout_width="100dp"`, sedangkan untuk tinggi gambar diatur pada `android:layout_height="100dp"`, posisi dari gambar yang bergantung

pada sumbu X dan Y diatur di `android:layout_gravity="center_horizontal"` dan gambar diakses dari `android:src="@drawable/logolab"`. *Tag* selanjutnya `LinearLayout` yang berfungsi untuk pengaturan layout dimana pengaturannya komponennya dapat rapi secara vertical dan horizontal. *Tag* `EditText` yang pertama ini digunakan untuk memasukan username kemudian yang kedua untuk mengisi password. Kemudian *tag* `button` yang pertama bertulisan `resigister` dengan baris koding `android:text="@string/register"`. `Button` yang kedua bertuliskan `login` dengan baris koding untuk mengatur nya `android:text="@string/login"`.

Selanjutnya agar halaman ini dapat bekerja, kita perlu mengatur *back-end language* pada halaman ini yaitu Java pada Android Studio. *Source code* diinput pada file `LoginActivity.java` seperti di bawah:

```
import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.text.TextUtils;
import android.util.Log;
import android.util.Printer;
import android.view.KeyEvent;
import android.view.View;
import android.view.inputmethod.EditorInfo;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import java.security.PrivateKey;

public class LoginActivity extends AppCompatActivity {

    private EditText etUsernameLogin, etPasswdLogin;
    private Button btnSignUpLogin, btnSignInLogin;
    String username, password;
    private User user;
    private DatabaseHandler presenter;

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

```

        initView();
        initData();
        login();
        register();
    }

    private void initView() {
        etUsernameLogin = findViewById(R.id.etUsernameLogin);
        etPasswdLogin = findViewById(R.id.etPasswdLogin);
        btnSignInLogin = findViewById(R.id.btnSignInLogin);
        btnSignUpLogin = findViewById(R.id.btnSignUpLogin);
    }

    private void register() {
        btnSignUpLogin.setOnClickListener(V -> {
            Intent reg = new Intent(this, RegisterActivity.class);
            startActivity(reg);
            finish();
        });
    }

    private void initData() {
        user = new User();
        presenter = new DatabaseHandler(this);
    }

    private Boolean validation() {
        username = etUsernameLogin.getText().toString();
        password = etPasswdLogin.getText().toString();

        if(username.isEmpty()){
            Toast.makeText(this, "Isikan username",
                Toast.LENGTH_SHORT).show();
            Log.e("Validation","false");
            return false;
        }

        if(password.isEmpty()){
            Toast.makeText(this, "Isikan Password",
                Toast.LENGTH_SHORT).show();
            Log.e("Validation","false");
            return false;
        }

        Log.e("Validation","true");
        return true;
    }

    private void login() {
        btnSignInLogin.setOnClickListener(v -> actLogin());
    }

```

```

private void actLogin(){
    if(validation()){
        if(loginData()){
            Log.e("actLogin","true");
            initPreference();
            Intent main = new Intent(this,
MainActivity.class);
            startActivity(main);
            finish();
        } else {
            Log.e("actLogin","false");
            Toast.makeText(this, "Login gagal",
Toast.LENGTH_LONG).show();
        }
    }
}

private Boolean loginData(){
    user = presenter.getLogin(username);

    if (password.equals(user.getPassword())){
        Log.e("loginData", "true");
        return true;
    } else {
        Log.e("loginData", "false");
        return false;
    }
}

private void initPreference() {
    SharedPreferences preferences =
getSharedPreferences("LoginPreference", Context.MODE_PRIVATE);
    SharedPreferences.Editor editor = preferences.edit();
    editor.putString("password", password);
    editor.putString("username", username);
    editor.commit();
    editor.apply();
}
}

```

Bagian awal pada *source code* diatas berisi library-library untuk mendukung fitur-fitur yang akan digunakan pada halaman LoginActivity. Kemudian pendeklarasian variabel yaitu username dan password dengan tipe data String, kemudian etUsernameLogin dan etPasswordLogin yang diambil dari *tag* XML EditText, btnSignUpLogin dan btnSignInLogin yang diambil dari *tag* XML Button, *user* dengan *modifier private* diambil dari Class User, presenter yang dibuat dari class DatabaseHandler. *Source code* terdiri dari banyak method, method yang

pertama adalah Oncreate dengan modifier protected dan metode akses void (non-return). Pada method ini terletak semua method yang akan diinisialisasi yakni initView(), initData(), login(), dan register(). Method initView merupakan method yang berfungsi untuk menginisialisasi tampilan pada halaman LoginActivity. Method register dengan modifier private dan metode akses void(non-return berisikan btnSignUpLogin.setOnClickListener(V -> { Intent reg = new Intent(this, RegisterActivity.class); startActivity(reg);finish();}); yang berguna untuk memproses button register. Ketika button register (btnSignUpLogin) di klik maka activity akan berpindah ke RegisterActivity. Method initData bertugas untuk melakukan inisialisasi data yang diambil dari user dan class DatabaseHandler. Kemudian method validation yang betipe Boolean sehingga hanya mempunyai 2 hasil keluaran yaitu true or false. Method ini yang akan mengecek apakah username dan password sudah sesuai apa belum. Nantinya jika username tidak diisikan maka akan muncul notifikasi “Isikan username”, jika password belum diisikan maka akan muncul notifikasi “Isikan password”, dan jika sudah terisi semua maka akan validation akan bernilai true. Method login berisi btnSignInLogin.setOnClickListener(v -> actLogin()); ketika button login di click maka activity baru akan berpindah ke actLogin. Dan diproses pada method actLogin yang berisi validasi dari username dan password jika semua proses sudah sesuai maka akan berpindah ke MainActivity. Kemudian method loginData yang berisi pengecekan password dimana baris kodingannya user = presenter.getLogin(username); if(password.equals(user.getPassword())) Log.e("loginData" "true"); return true; } else { Log.e("loginData", "false"); return false; }. Method terakhir initPreference yang mengambil hasil username dan password yang tersimpan pada database.

```
package com.example.mypc.aplikasimodul2kel37;
```

```

public class User {
    int id;
    static String username;
    static String password;

    public User() {
    }

    public User(int id, String username, String password) {
        this.id = id;
        this.username = username;
        this.password = password;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getUsername() {
        return username;
    }

    public void setUsername(String username) {
        this.username = username;
    }

    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
        this.password = password;
    }
}

```

Source code diatas diisi pada user.java yang berisi package com.example.mypc.aplikasimodul2kel37; package ini sesuai dengan nama package saat awal proyek dibuat. Kodingan ini berisi variable id bertipe integer, username bertipe data string, dan password bertipe data string. Kemudian constructor user dengan parameter int id, string username, string password. Method ini berhubungan dengan segala kebutuhan user dari id, username dan juga password.

```

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;

```

```

import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.util.Log;

import java.util.ArrayList;
import java.util.List;

public class DatabaseHandler extends SQLiteOpenHelper {

    // static variable
    private static final int DATABASE_VERSION = 1;

    // Database name
    private static final String DATABASE_NAME = "modulandroid";

    // table name
    private static final String TABLE_MAHASISWA = "mahasiswa";

    // column tables
    private static final String KEY_ID = "id";
    private static final String KEY_NAME = "username";
    private static final String KEY_TALL = "password";

    public DatabaseHandler(Context context){
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    //Create table
    @Override
    public void onCreate(SQLiteDatabase db) {
        String CREATE_MAHASISWA_TABLE = "CREATE TABLE " +
TABLE_MAHASISWA + "("
            + KEY_ID + " INTEGER PRIMARY KEY," + KEY_NAME + "
TEXT,"
            + KEY_TALL + " TEXT" + ")";
        db.execSQL(CREATE_MAHASISWA_TABLE);
    }

    // on Upgrade database
    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int
newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_MAHASISWA);
        onCreate(db);
    }

    public void addUser(User userModel){
        SQLiteDatabase db = getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(KEY_NAME, userModel.getUsername());
        values.put(KEY_TALL, userModel.getPassword());
        db.insert(TABLE_MAHASISWA, null, values);
    }
}

```

```

        db.close();
    }

    public User getMahasiswa(int id) {
        SQLiteDatabase db = this.getReadableDatabase();

        Cursor cursor = db.query(TABLE_MAHASISWA, new String[] {
KEY_ID,
                                KEY_NAME, KEY_TALL }, KEY_ID + "=?",
        new String[] { String.valueOf(id) }, null, null,
        null, null);
        if (cursor != null)
            cursor.moveToFirst();

        User mahasiswa = new
User(Integer.parseInt(cursor.getString(0)),
        cursor.getString(1), cursor.getString(2));
        // return mahasiswa
        return mahasiswa;
    }

    public User getLogin(String username) {
        User mahasiswa;
        SQLiteDatabase db = this.getReadableDatabase();

        Cursor cursor = db.query(TABLE_MAHASISWA, new String[] {
KEY_ID,
                                KEY_NAME, KEY_TALL }, KEY_NAME + "=?",
        new String[] { username }, null, null, null,
        null);

        if (cursor != null) {
            cursor.moveToFirst();
        }

        int nomor = cursor.getCount();

        if(nomor>0){
            mahasiswa = new
User(Integer.parseInt(cursor.getString(0)),
            cursor.getString(1), cursor.getString(2));
        } else {
            mahasiswa = new User(0,
            "failed", "failed");
        }

        // return mahasiswa
        return mahasiswa;
    }

    // get All Record
    public List<User> getAllRecord() {

```

```

        List<User> mahasiswaList = new ArrayList<User>();
        // Select All Query
        String selectQuery = "SELECT  * FROM " + TABLE_MAHASISWA;

        SQLiteDatabase db = this.getWritableDatabase();
        Cursor cursor = db.rawQuery(selectQuery, null);

        if (cursor.moveToFirst()) {
            do {
                User userModel = new User();

                userModel.setId(Integer.parseInt(cursor.getString(0)));
                userModel.setUsername(cursor.getString(1));
                userModel.setPassword(cursor.getString(2));

                mahasiswaList.add(userModel);
            } while (cursor.moveToNext());
        }

        // return mahasiswa list
        return mahasiswaList;
    }

    public int updateMahasiswa(User mahasiswa) {
        SQLiteDatabase db = this.getWritableDatabase();

        ContentValues values = new ContentValues();
        values.put(KEY_NAME, mahasiswa.getUsername());
        values.put(KEY_TALL, mahasiswa.getPassword());

        // updating row
        return db.update(TABLE_MAHASISWA, values, KEY_ID + " = ?",
            new String[] { String.valueOf(mahasiswa.getId())
        });
    }

    public void deleteModel(User mahasiswa) {
        SQLiteDatabase db = this.getWritableDatabase();
        db.delete(TABLE_MAHASISWA, KEY_ID + " = ?",
            new String[] { String.valueOf(mahasiswa.getId())
        });

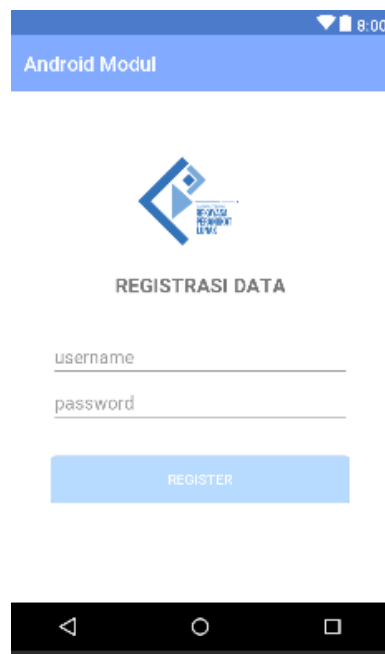
        db.close();
    }
}

```

Pada DatabaseHandler.java yang berisi source code seperti diatas digunakan untuk mengatur segala yang berhubungan dengan database. Database yang dipakai dengan nama modulandroid baris kodingannya `private static final String DATABASE_NAME = "modulandroid";`. Dan nama tabelnya mahasiswa baris

kodingannya `private static final String TABLE_MAHASISWA = "mahasiswa";` . pada *source code* ini berisi tambah user, update mahasiswa, Delete mahasiswa. Pada method `addUser` akan menambahkan data ke database dengan kodingan `SQLiteDatabase db = getWritableDatabase(); ContentValues values = new ContentValues(); values.put(KEY_NAME, userModel.getUsername()); values.put(KEY_TALL, userModel.getPassword()); db.insert(TABLE_MAHASISWA, null, values); db.close();` . Pada method `updateMahasiswa` akan mengupdate data dari tabel mahasiswa yang sudah dimasukan sebelumnya. Kemudian method `deleteModel` untuk menghapus nilai dalam tabel sesuai dengan id yang terpilih.

3.5.2 Membuat RegisterActivity



Gambar 3.34 Tampilan RegisterActivity

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/white"
    tools:context=".LoginActivity">

    <RelativeLayout
```

```

        android:id="@+id/rlLogin"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="60dp">

        <ImageView
            android:id="@+id/ivLogin"
            android:layout_width="100dp"
            android:layout_height="100dp"
            android:layout_gravity="center_horizontal"
            android:layout_marginLeft="130dp"
            android:src="@drawable/logolab"
            android:contentDescription="LOGO" />

        <android.support.v7.widget.AppCompatTextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="@string/regtittle"
            android:textAlignment="center"
            android:textSize="20sp"
            android:textStyle="bold"
            android:layout_marginTop="120dp" />

        <LinearLayout
            android:id="@+id/llLogin"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_centerInParent="true"
            android:layout_marginTop="330dp"
            android:layout_marginEnd="50dp"
            android:orientation="vertical"
            android:padding="18dp">

            <EditText
                android:id="@+id/etUsernameRegister"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:hint="@string/user"
                android:inputType="textEmailAddress"
                android:maxLines="1"
                android:singleLine="true" />

            <EditText
                android:id="@+id/etPasswordRegister"
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginBottom="30dp"
                android:hint="@string/password"
                android:imeActionId="6"
                android:imeActionLabel="Sign In"
                android:imeOptions="actionUnspecified"

```

```

        android:inputType="textPassword"
        android:maxLines="1"
        android:singleLine="true" />

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content">

            <Button
                android:id="@+id/btRegister"
                android:layout_width="0dp"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:background="@drawable/roundedbutton"
                android:text="@string/register"
                android:textColor="@color/white" />

            </LinearLayout>
        </LinearLayout>
    </RelativeLayout>
</ScrollView>

```

Source code diatas untuk membuat tampilan pada RegisterActivity. Yang terdiri dari fitur *image*, *textfield*, *button*. Pengaturan gambar dikelola di *Tag* *ImageView* ini berfungsi untuk menampilkan gambar, dimana lebar gambar diatur di `android:layout_width="100dp"`, sedangkan untuk ukuran tinggi gambar diatur di `android:layout_height="100dp"`, posisi dari gambar yang bergantung pada sumbu X dan Y diatur di `android:layout_gravity="center_horizontal"`. Dan gambar diakses dari `android:src="@drawable/logolab"`. *Tag* selanjutnya *LinearLayout* yang berfungsi untuk pengaturan layout dimana pengaturannya komponennya dapat rapi secara vertical dan horizontal. *Tag* *EditText* yang pertama ini digunakan untuk memasukan username kemudian yang kedua untuk mengisi *password*. Kemudian *tag* *button* yang pertama bertulisan *resigister* dengan baris koding `android:text="@string/register"`.

```

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.widget.Button;
import android.widget.EditText;

public class RegisterActivity extends AppCompatActivity {

```



```

private DatabaseHandler databaseHandler;
private User userModel;
private EditText etUsernameRegister;
private EditText etPasswordRegister;
private Button btRegister;
private String username,password;

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

    initView();
    register();
}

private void register() {
    btRegister.setOnClickListener(V ->
        initDataHandler()
    );
}

private void initView() {
    etUsernameRegister =
findViewById(R.id.etUsernameRegister);
    etPasswordRegister =
findViewById(R.id.etPasswordRegister);
    btRegister = findViewById(R.id.btRegister);
}

private void initUser() {
    username = etUsernameRegister.getText().toString();
    password = etPasswordRegister.getText().toString();

    userModel = new User();
    userModel.setUsername(username);
    userModel.setPassword(password);
}

private void initDataHandler() {
    initUser();

    databaseHandler = new DatabaseHandler(this);
    databaseHandler.addUser(userModel);
    User model = databaseHandler.getMahasiswa(1);
    Log.e("record", model.getUsername().toString());
    Intent admin = new Intent(RegisterActivity.this,
LoginActivity.class);
    startActivity(admin);
    finish();
}

```

```
}

```

Source code diatas dimasukan pada *file* RegisterActivity.java. dimana pada bagian awal kodingan adalah *import library* untuk pendukung fitur-fitur yang akan dipakai, yakni

```
import android.content.Intent; import
android.support.v7.app.AppCompatActivity; import android.os.Bundle; import
android.util.Log; import android.widget.Button; import
android.widget.EditText;
```

didalam *class* RegisterActivity berisikan variable-variabel yaitu databaseHandler yang dibuat dari *class* DatabaseHandler, *usermodel* yang dibuat dari *class* User, etUsernameRegister yang dibuat dari *class* Tag EditText, etPasswordRegister yang dibuat dari *class* Tag EditText, btRegister sebagai nama variable dari Button, username dan password yang bertipe data string dengan modifier private. Pada method Oncreate yang akan menginisialisasi program ini. method ini berisi method initView yang berfungsi untuk inisialisasi *view* dengann barisan kodingan

```
etUsernameRegister = findViewById(R.id.etUsernameRegister);
etPasswordRegister = findViewById(R.id.etPasswordRegister); btRegister
= findViewById(R.id.btRegister);
```

, kemudian register berisi proses ketika *button register* ditekan, setelah ditekan program akan mengalihkan proses ke initDataHandler(). Baris kodingan pada method ini akan mengambil data mahasiswa yang tersimpan pada database. Method terakhir adalah initUser yang berguna untuk inisialisasi user. Dimana hasil ketikan pada text field akan disimpan pada variable username dan password dan disimpan pada database.

3.5.3 Membuat MainActivity



Gambar 3.35 Tampilan MainActivity

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <android.support.v7.widget.CardView
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:id="@+id/cardView"
        style="@style/CardView.Light"
        android:layout_width="320dp"
        android:layout_height="400dp"
        android:layout_marginBottom="8dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.100000024">

        <android.support.constraint.ConstraintLayout
            xmlns:app="http://schemas.android.com/apk/res-auto"
            android:layout_width="match_parent"
            android:layout_height="wrap_content">
```

```

        <ImageView
            android:id="@+id/media_image"
            android:layout_width="0dp"
            android:layout_height="0dp"
            android:scaleType="fitCenter"
            android:src="@drawable/wall1"

app:layout_constraintBottom_toTopOf="@+id/primary_text"
            app:layout_constraintDimensionRatio="H,16:9"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"
            app:layout_constraintTop_toTopOf="parent"
            app:layout_constraintVertical_chainStyle="packed"
        />

        <TextView
            android:id="@+id/primary_text"
            android:layout_width="0dp"
            android:layout_height="61dp"
            android:layout_marginStart="16dp"
            android:layout_marginTop="24dp"
            android:layout_marginEnd="16dp"
            android:text="@string/Header"

android:textAppearance="@style/TextAppearance.AppCompat.Headline"
            android:textColor="#DE000000"

app:layout_constraintBottom_toTopOf="@+id/tvNameMain"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toBottomOf="@+id/media_image" />

        <TextView
            android:id="@+id/tvNameMain"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_marginStart="16dp"
            android:layout_marginTop="0dp"
            android:layout_marginEnd="16dp"
            android:text="@string/name"

android:textAppearance="@style/TextAppearance.AppCompat.Body1"
            android:textColor="#8A000000"
            android:textSize="15sp"
            app:layout_constraintBottom_toTopOf="parent"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintHorizontal_bias="0.0"
            app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toBottomOf="@+id/primary_text" />

```

```

        <TextView
            android:id="@+id/tvdescription"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_marginTop="32dp"
            android:layout_marginBottom="8dp"
            android:text="@string/description"

            android:textAppearance="@style/TextAppearance.AppCompat.Body1"
            android:textColor="#8A000000"
            android:textSize="20sp"
            app:layout_constraintBottom_toBottomOf="parent"
            app:layout_constraintEnd_toEndOf="@+id/tvNameMain"
            app:layout_constraintHorizontal_bias="0.0"

            app:layout_constraintStart_toStartOf="@+id/tvNameMain"
            app:layout_constraintTop_toTopOf="@+id/tvNameMain"
            app:layout_constraintVertical_bias="1.0" />

    </android.support.constraint.ConstraintLayout>

</android.support.v7.widget.CardView>

<Button
    android:id="@+id/btnLogoutMain"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="8dp"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="6dp"
    android:layout_marginBottom="8dp"
    android:background="@drawable/roundedbutton"
    android:minWidth="0dp"
    android:paddingStart="8dp"
    android:paddingEnd="8dp"
    android:text="@string/logout"
    android:textColor="#de101010"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="@+id/cardView"
    app:layout_constraintTop_toBottomOf="@+id/cardView"
    app:layout_constraintVertical_bias="0.753" />

</android.support.constraint.ConstraintLayout>

```

Pada tampilan MainActivity memiliki beberapa fitur seperti image, button dan paragraph. Paragraph diatur pada tag TextView dengan kodingan seperti diatas.

Dengan id primary text, dan tinggi layout 61dp, lebar 0 dp dll. Kemudian fitur button MainActivity.Java. Berikut adalah listing code MainActivity.java,

```
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.SharedPreferences;
import android.support.v7.app.AlertDialog;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.util.Log;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    private DatabaseHandler databaseHandler;
    private User userModel;
    private TextView tvNameMain;
    private Button btnLogoutMain;
    private Button exit;

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

        initView();
        initPreference();
        logout();
    }

    @Override
    public void onBackPressed() {
        showAlertDialog();
    }

    private void initView() {
        tvNameMain = findViewById(R.id.tvNameMain);
        btnLogoutMain = findViewById(R.id.btnLogoutMain);
        exit = findViewById(R.id.btnLogoutMain);
    }

    private void initPreference() {
        SharedPreferences preferences =
        getSharedPreferences("LoginPreference", MODE_PRIVATE);
        String username = preferences.getString("username", "");

        tvNameMain.setText(username);
    }
}
```

```

    }

    private void deletePreference(){
        SharedPreferences preferences =
        getSharedPreferences("LoginPreference", MODE_PRIVATE);
        preferences.edit().remove("username").commit();
        preferences.edit().remove("password").commit();
    }

    private void logout() {
        exit.setOnClickListener(view -> showAlertDialog());
    }

    public void showAlertDialog() {
        new AlertDialog.Builder(this)
            .setMessage("Apa kalian ingin Logout?")
            .setCancelable(false)
            .setPositiveButton("Yes", new
        DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog,
        int id) {
                deletePreference();
                Intent login = new
        Intent(MainActivity.this, LoginActivity.class);
                startActivity(login);
                finish();
            }
        })
            .setNegativeButton("No", null)
            .show();
    }
}

```

Pada *source code* diatas untuk *back end program* dari halaman MainActivity, yang berisi awalnya adalah mengimport *library* untuk mensupport fitur-fitur yang dipakai di MainActivity. *Class* ini berisikan variable *databaseHandler* yang dibuat dari *class DatabaseHandler*, *usermodel* dari *class User*, *tvNameMain* dari *tag TextView*, *btnLogout* *variable button*, dan *button exit*. Ketika menekan tombol *exit* maka *method logout* akan berfungsi untuk mengeksekusi ke *method showAlertDialog* yang berisi Secara garis besar ketika menekan *button exit (logout)* maka akan muncul dialog alert yang berisikan “apakah kalian ingin logout?” jika iya maka *preference* akan dihapus sehingga perlu login lagi untuk masuk ke program. Jika tidak maka tetap berada pada halaman ini.

3.5.4 Membuat Splashscreen



Gambar 3.36 Tampilan Splashscreen

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

    <ImageView
        android:src="@drawable/splashscreen"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:scaleType="fitXY"/>

    <ProgressBar
        android:id="@+id/progressBar1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_centerHorizontal="true"
        android:layout_marginBottom="58dp" />

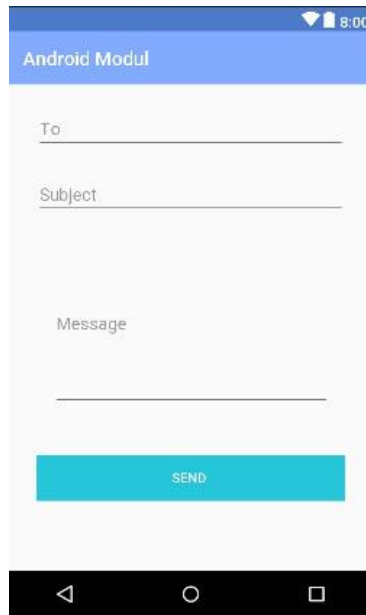
</RelativeLayout>
```

Pada *source code* ini untuk membuat tampilan seperti gambar di atasnya. Dengan 2 *tag* yang dipakai yaitu *ImageView* untuk menampilkan gambar dan *progressbar* untuk menampilkan *progress* ketika membuka program. Kemudian pada *tag ImageView* berisi `android:src="@drawable/Splashscreen"` mengambil

gambar yang berada pada *file drawable* dengan nama *file* png/jpg nya *SplashScreen*.

```
android:layout_width="fill_parent"    sesuai    dengan    parent    nya,
android:layout_height="fill_parent"  sesuai    dengan    parent    nya,
android:scaleType="fitXY"/>. Dan progressbar berisi baris kodingan
android:id="@+id/progressBar1"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_alignParentBottom="true"
android:layout_centerHorizontal="true"
android:layout_marginBottom="58dp" />.
```

3.5.5 Membuat EmailActivity



Gambar 3.37 Tampilan EmailActivity

```
import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;

public class EmailActivity extends AppCompatActivity {
```

```

Button buttonSend;
EditText textTo;
EditText textSbj;
EditText textMsg;

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

    buttonSend = (Button) findViewById(R.id.buttonSend);
    textTo = (EditText) findViewById(R.id.editTextTo);
    textSbj = (EditText) findViewById(R.id.editTextSbj);
    textMsg = (EditText) findViewById(R.id.editTextMsg);

    buttonSend.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {

            String to = textTo.getText().toString();
            String subject = textSbj.getText().toString();
            String message = textMsg.getText().toString();

            Intent email = new Intent(Intent.ACTION_SEND);
            email.putExtra(Intent.EXTRA_EMAIL, new String[]{
to});
            email.putExtra(Intent.EXTRA_SUBJECT, subject);
            email.putExtra(Intent.EXTRA_TEXT, message);

            email.setType("message/rfc822");

            startActivity(Intent.createChooser(email, "Choose
an Email client :"));
        }
    });

    @Override
    public void onBackPressed() {
        back();
    }

    private void back() {
        Intent reg = new Intent(this, MainActivity.class);
        startActivity(reg);
        finish();
    }
}

```

Source code diatas merupakan tampilan pada EmailActivity. Fitur yang ada pada tampilan EmailActivity ada *Textfield* dan *button*. *Textfield* yang pertama

berikan alamat email yang dituju basis kodingannya
`android:id="@+id/editTextTo"` baris ini sebagai id dari *textfield* ini dengan nama
 id `editTextTo`. `android:layout_width="0dp"`, lebar dari text sebesar 0
 dp, `android:layout_height="44dp"`, Panjang dari *textfield* sebesar
 44dp `android:hint="To"` ini isi pada saat *textfield* diawal inisialisasi,
`android:inputType="textEmailAddress"` tipe inputan harus ada @ nya karena
email tipenya. Kemudian pada *textfield* yang kedua digunakan sebagai subjek saat
 mengirimkan *email* dengan baris kodingannya `android:id="@+id/editTextSbj"`
 baris ini menunjukan id dari *textfield* ini, `android:layout_width="0dp"`
 lebar dari *textfield*-nya adalah 0 dp, `android:layout_height="44dp"`
 panjang dari *textfield*nya adalah 44 dp, `android:inputType="textEmailSubject"`
 tipe inputannya *email* sehingga sesuai dengan sintak dari email
`android:hint="Subject "` ini isi pada saat *textfield* diawal inisialisasi. *Textfield*
 yang terakhir untuk menampung pesan yang ingin disampaikan
`android:id="@+id/editTextMsg"` nama id dari *textfield* ini `editTextMsg`,
`android:layout_width="291dp"` lebarnya dari *textfiel*nya adalah 219dp,
`android:layout_height="176dp"` panjangnya dari *textfiel*nya adalah 219dp,
`android:inputType="textMultiLine"` tipe inputannya *multi line* sehingga dapat
 menampung kata-kata/pesan-pesan yang ingin disampaikan,
`android:hint="Message "` ini isi pada saat *textfield* diawal inisialisasi. Terakhir
tag button yang nantinya akan berguna untuk memproses hasil dari inputan pada
textfield `android:id="@+id/buttonSend"` nama id dari *button* ini `buttonSend`
 dengan lebar 0 dp, dan tingginya sesuai dengan kontennya
`android:layout_width="0dp"` `android:layout_height=`
`"wrap_content"`. `android:background="#26c6da"` baris ini mengatur warna
 dari *background button* dimana hexa #26c6da bermakna warna dengan persentase
 4.9% red, 77.6% green and 85.5% blue. `android:text="Send "` ini tulisan yang

ada pada *button* `android:textColor="@android:color/background_light"`
barisan ini mengatur warna dari text nya.

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".EmailActivity">

    <EditText
        android:id="@+id/editTextSbj"
        android:layout_width="0dp"
        android:layout_height="44dp"
        android:layout_marginStart="30dp"
        android:ems="10"
        android:inputType="textEmailSubject"
        android:hint="Subject "
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        tools:layout_constraintLeft_creator="1"
        tools:layout_constraintTop_creator="1"
        tools:layout_constraintRight_creator="1"
        android:layout_marginEnd="30dp"
        android:layout_marginTop="25dp"
        app:layout_constraintTop_toBottomOf="@+id/editTextTo" />

    <EditText
        android:id="@+id/editTextTo"
        android:layout_width="0dp"
        android:layout_height="44dp"
        android:hint="To "
        android:fontFamily="@string/app_name"
        android:inputType="textEmailAddress"
        tools:layout_constraintTop_creator="1"
        android:layout_marginTop="25dp"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        tools:layout_constraintRight_creator="1"
        android:layout_marginStart="30dp"
        android:layout_marginEnd="30dp"
        tools:layout_constraintLeft_creator="1"
        app:layout_constraintTop_toTopOf="parent" />

    <EditText
        android:id="@+id/editTextMsg"
        android:layout_width="291dp"
```

```

        android:layout_height="176dp"
        android:layout_marginStart="30dp"
        android:ems="10"
        android:inputType="textMultiLine"
        android:hint="Message "
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        tools:layout_constraintLeft_creator="1"
        tools:layout_constraintTop_creator="1"
        tools:layout_constraintRight_creator="1"
        android:layout_marginEnd="30dp"
        android:layout_marginTop="25dp"
        app:layout_constraintTop_toBottomOf="@+id/editTextSbj"
        app:layout_constraintHorizontal_bias="0.516"
        android:layout_marginLeft="16dp"
        android:layout_marginRight="16dp"
        tools:layout_editor_absoluteX="62dp" />

<Button
    android:id="@+id/buttonSend"
    android:layout_width="0dp"
    android:layout_height="wrap_content"
    android:layout_marginEnd="30dp"
    android:layout_marginLeft="30dp"
    android:layout_marginRight="30dp"
    android:layout_marginStart="30dp"
    android:background="#26c6da"
    android:text="Send "
    android:textColor="@android:color/background_light"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    tools:layout_constraintBottom_creator="1"
    tools:layout_constraintLeft_creator="1"
    tools:layout_constraintRight_creator="1"
    tools:layout_constraintTop_creator="1"
    android:layout_marginTop="25dp"
    app:layout_constraintTop_toBottomOf="@+id/editTextMsg"
    app:layout_constraintVertical_bias="0.251" />

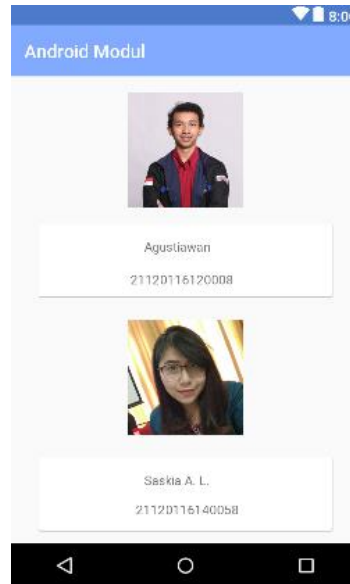
</android.support.constraint.ConstraintLayout>

```

Kemudian *source code* diatas sebagai *back end* nya, sehingga fitur-fiturnya bisa berjalan sesuai dengan kemauan *developer*. Bagian pertama mengimport *library-library* yang dibutuhkan `import android.content.Intent; import android.support.v7.app.AppCompatActivity; import android.os.Bundle; import android.view.View; import android.widget.Button; import android.widget.EditText; .` variable yang dipakai `buttonSend` variable `button`,

textTo dari EditText, textSbj dari EditText bagian Subjek, dan textMsg dari EditText bagian Message. Kemudian ketika buttonSend Diclick maka proses eksekusi akan dijalankan dengan baris koding , buttonSend = (Button) findViewById(R.id.buttonSend); textTo = (EditText) findViewById(R.id.editTextTo); textSbj = (EditText) findViewById(R.id.editTextSbj); textMsg = (EditText) findViewById(R.id.editTextMsg); baris ini yang akan mencocokkan dengan fitur-fitur yang dibuat pada xaml tadi sesuai dengan idnya. Kemudian String to = textTo.getText().toString(); inputan yang diisi pada textfield dengan id textTo ini akan disimpan pada variable to dengan tipe data StringString subject = textSbj.getText().toString(); inputan yang diisi pada textfield dengan id textSbj ini akan disimpan pada variable subject dengan tipe data String. String message = textMsg.getText().toString(); inputan yang diisi pada *textfield* dengan id textMsg ini akan disimpan pada variable message dengan tipe data String. Intent email = new Intent(Intent.ACTION_SEND); objek *email* ini yang akan menyimpan masukan pada textfieldnya dan akan dikirimkan sesuai dengan *email* tujuan, email.putExtra(Intent.EXTRA_EMAIL, new String[]{ to}); email.putExtra(Intent.EXTRA_SUBJECT, subject); email.putExtra(Intent.EXTRA_TEXT, message);. Method onBackPressed ketika diklik maka akan memproses *method back* yang berisi Intent reg = new Intent(this, MainActivity.class); akan memindahkan ke MainActivity startActivity(reg); menjalankan objek reg yang menyimpan eksekusi untuk pindah halaman ke MainActivity. finish(); kemudian selesai.

3.5.6 Membuat AboutActivity



Gambar 3.38 Tampilan activity_about

Berikut adalah source code AboutActivity.java yang digunakan untuk menampilkan tampilan pada activity_about.xml. Terdapat juga method onBackPressed yang digunakan agar ketika tombol back ditekan atau sentuh maka kembali ke halaman sebelumnya, yaitu MainActivity.

```
package com.example.mypc.aplikasimodul2kel37;

import android.content.Intent;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;

public class AboutActivity extends AppCompatActivity {

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

    @Override
    public void onBackPressed() {
        back();
    }

    private void back() {
```

```

        Intent reg = new Intent(this, MainActivity.class);
        startActivity(reg);
        finish();
    }
}

```

Berikut adalah listing code about_activity.xml

```

<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"

    tools:context="com.example.mypc.aplikasimodul2kel37.AboutActivity">

    <ImageView
        android:id="@+id/ivpraktikan1"
        android:layout_width="126dp"
        android:layout_height="145dp"
        android:layout_marginStart="8dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="8dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:srcCompat="@drawable/agus" />

    <android.support.v7.widget.CardView
xmlns:android="http://schemas.android.com/apk/res/android"
        android:id="@+id/cardView"
        style="@style/CardView.Light"
        android:layout_width="320dp"
        android:layout_height="80dp"
        android:layout_marginTop="8dp"
        android:layout_marginBottom="8dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/ivpraktikan1"
        app:layout_constraintVertical_bias="0.0">

        <android.support.constraint.ConstraintLayout
xmlns:app="http://schemas.android.com/apk/res-auto"
            android:layout_width="match_parent"
            android:layout_height="wrap_content">

            <TextView

```



```

        android:id="@+id/twnama1"
        android:layout_width="97dp"
        android:layout_height="24dp"
        android:layout_marginLeft="24dp"
        android:layout_marginTop="16dp"
        android:layout_marginEnd="108dp"
        android:text="@string/nameabout1"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintTop_toTopOf="parent"
    />

    <TextView
        android:id="@+id/twnim1"
        android:layout_width="127dp"
        android:layout_height="26dp"
        android:layout_marginTop="12dp"
        android:text="@string/nimabout1"
        app:layout_constraintEnd_toEndOf="@+id/twnama1"
        app:layout_constraintStart_toStartOf="@+id/twnama1"
        app:layout_constraintTop_toBottomOf="@+id/twnama1"
    />

</android.support.constraint.ConstraintLayout>

</android.support.v7.widget.CardView>

<ImageView
    android:id="@+id/ivpraktikan2"
    android:layout_width="126dp"
    android:layout_height="145dp"
    android:layout_marginTop="16dp"
    app:layout_constraintEnd_toEndOf="@+id/ivpraktikan1"
    app:layout_constraintHorizontal_bias="0.0"
    app:layout_constraintStart_toStartOf="@+id/ivpraktikan1"
    app:layout_constraintTop_toBottomOf="@+id/cardView"
    app:srcCompat="@drawable/saskia"
/>

<android.support.v7.widget.CardView
    android:id="@+id/cardView2"
    style="@style/CardView.Light"
    android:layout_width="320dp"
    android:layout_height="80dp"
    android:layout_marginTop="8dp"
    android:layout_marginBottom="8dp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="@+id/cardView"
    app:layout_constraintStart_toStartOf="@+id/cardView"
    app:layout_constraintTop_toBottomOf="@+id/ivpraktikan2">

    <android.support.constraint.ConstraintLayout
xmlns:app="http://schemas.android.com/apk/res-auto"

```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <TextView
            android:id="@+id/twnama2"
            android:layout_width="97dp"
            android:layout_height="24dp"
            android:layout_marginLeft="24dp"
            android:layout_marginTop="16dp"
            android:layout_marginEnd="108dp"
            android:text="@string/nameabout2"
            app:layout_constraintEnd_toEndOf="parent"
            app:layout_constraintLeft_toLeftOf="parent"
            app:layout_constraintTop_toTopOf="parent"
        />

        <TextView
            android:id="@+id/twnim2"
            android:layout_width="115dp"
            android:layout_height="32dp"
            android:layout_marginTop="8dp"
            android:text="@string/nimabout2"
            app:layout_constraintEnd_toEndOf="@+id/twnama2"
            app:layout_constraintStart_toStartOf="@+id/twnama2"
            app:layout_constraintTop_toBottomOf="@+id/twnama2"
        />

    </android.support.constraint.ConstraintLayout>

</android.support.v7.widget.CardView>

</android.support.constraint.ConstraintLayout>

```

Sourcecode diatas digunakan untuk membuat tampilan activity_about. Pada halaman ini ditampilkan data praktikan 1 dan data praktikan 2. Diatas tiap-tiap data praktikan terdapat foto masing-masing praktikan. Pada sourcecode diatas menggunakan beberapa view seperti imageView dan textView. Pada halaman about ini sudah menggunakan Construct Layout. Setelah halaman ini dibuat kemudian ditambahkan fungsi di class MainActivity untuk menambahkan fungsi ketika button about ditekan. Berikut adalah source code yang ditambahkan pada MainActivity.java

```

private void about() {
    btnAboutMain.setOnClickListener(V -> {
        Intent reg = new Intent(this, AboutActivity.class);
        startActivity(reg);
        finish();
    });
}

```

```
);  
}
```

Kemudian dibuat juga tampilan button About pada halaman utama. Berikut adalah source code yang ditambahkan pada main_activity.xml

```
<Button  
    android:id="@+id/btnAboutMain"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_marginStart="36dp"  
    android:layout_marginEnd="8dp"  
    android:layout_marginBottom="8dp"  
    android:background="@drawable/roundedbutton"  
    android:text="About"  
    app:layout_constraintBottom_toBottomOf="parent"  
    app:layout_constraintEnd_toEndOf="@+id/cardView"  
    app:layout_constraintHorizontal_bias="1.0"  
    app:layout_constraintStart_toEndOf="@+id/btnLogoutMain"  
    app:layout_constraintTop_toTopOf="@+id/btnLogoutMain"  
    app:layout_constraintVertical_bias="0.111" />
```

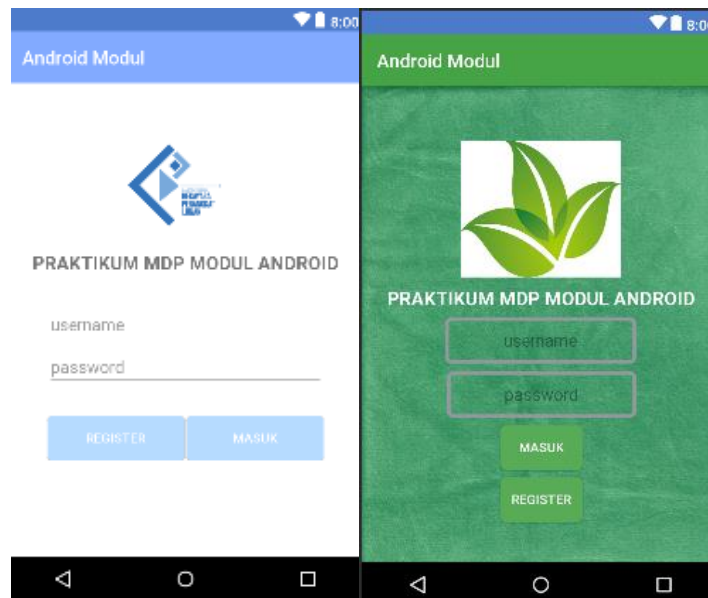
3.6 Tugas

3.6.1 Mengubah Logo

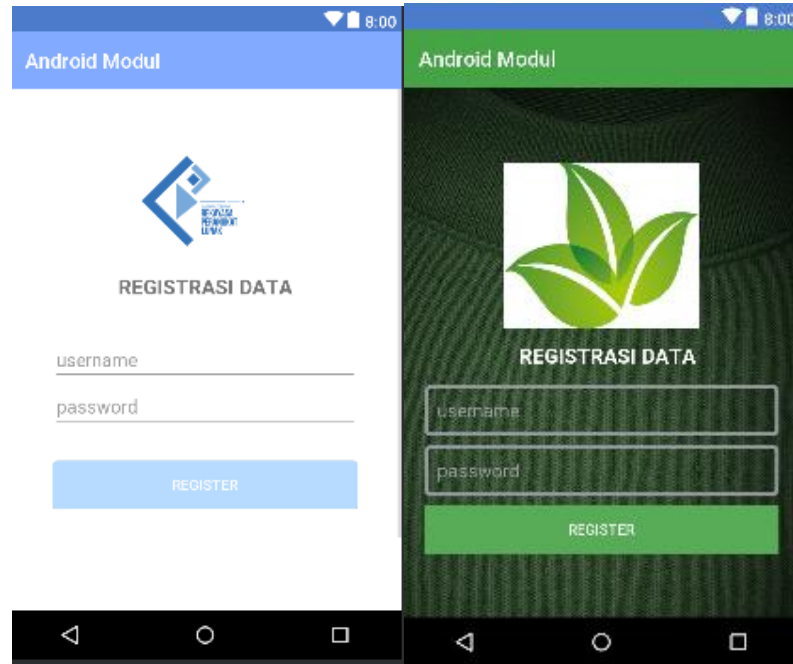
Langkah yang dilakukan untuk mengubah logo sangat sederhana, ada 2 metode dalam mengubah logo sebelumnya yaitu

1. menambahkan logo baru ke dalam package drawable dan mengubah penggunaan logo yang sebelumnya dengan logo baru pada file .xml yang menggunakan logo lama.
2. Menamakan logo baru dengan nama yang sama dengan logo lama, kemudian menghapus logo lama di drawable dilanjutkan dengan melakukan paste logo baru di package drawable.

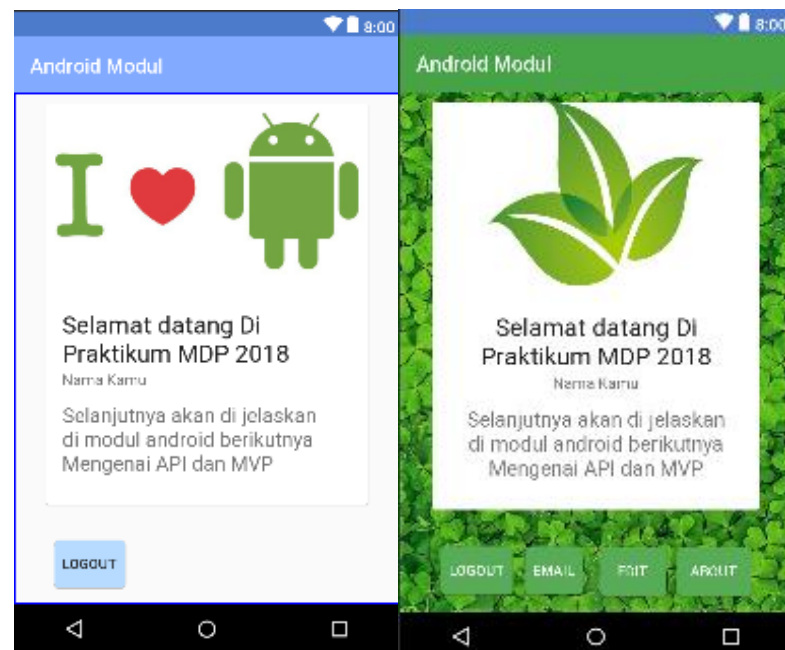
Dalam hal ini praktikan memilih metode yang kedua karena lebih mudah, berikut adalah tampilan sebelum dan sesudah logo diganti.



Gambar 3.39 Tampilan sebelum dan sesudah logo diganti pada halaman activity_login

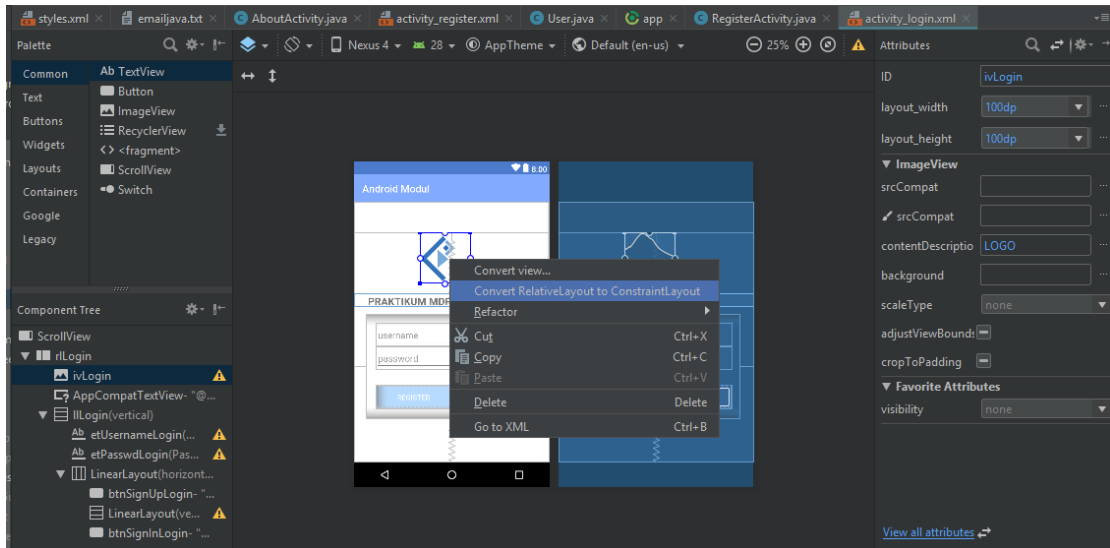


Gambar 3.40 Tampilan sebelum dan sesudah logo diganti pada halaman activity_register



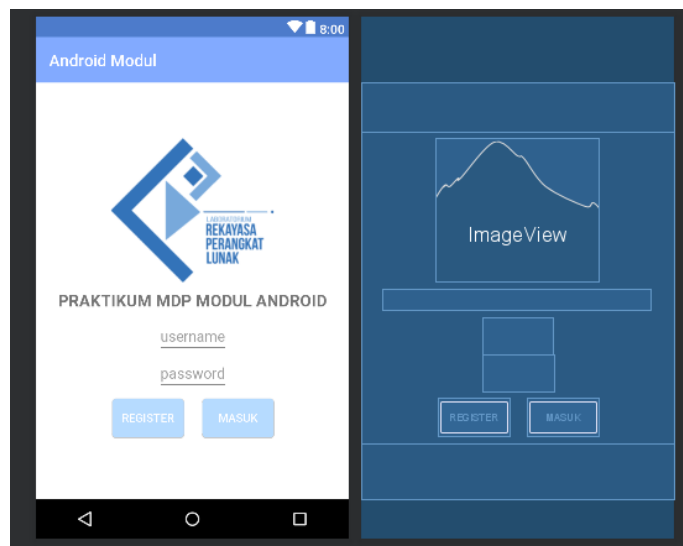
Gambar 3.41 Tampilan sebelum dan sesudah logo diganti pada halaman activity_main

3.6.2 Mengubah Struktur Layout Pada Login dan Registrasi Menjadi Constraint

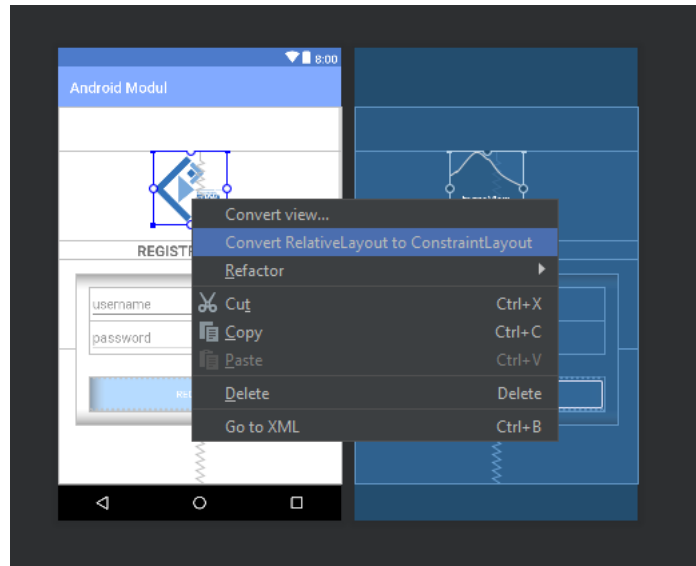


Gambar 3.42 Membuat construct layout (1)

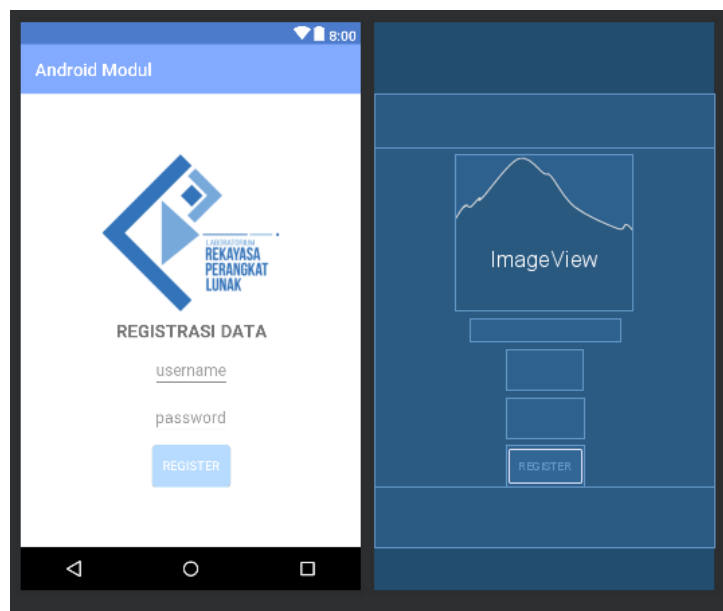
Cara untuk mengubah relative layout menjadi construct layout yaitu dengan cara klik kanan pada object seperti pada gambar diatas dan pilih ConvertRelativeLayout to ConstructLayout. Setelah itu tampilan akan berubah dan disusun sedemikian rupa sehingga menjadi seperti pada gambar dibawah ini,



Gambar 3.43 Tampilan setelah di convert menjadi construct layout pada activity_login



Gambar 3.44 Membuat construct layout (2)



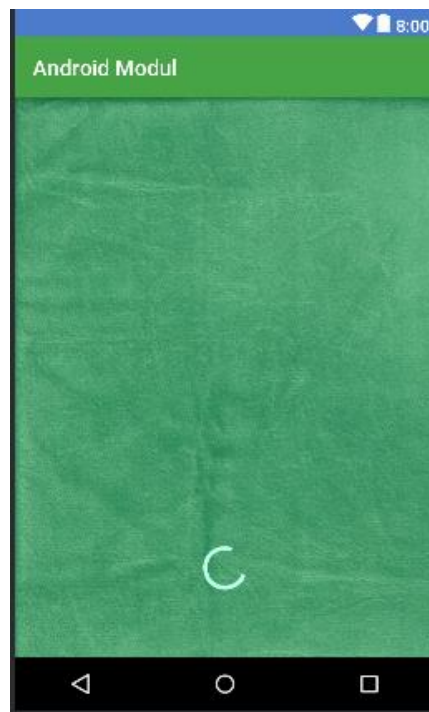
Gambar 3.45 Tampilan setelah di convert menjadi construct layout pada activity_register

3.6.3 Mengubah Tampilan Menjadi Material Design

Perubahan dilakukan pada background yang dipakai, pengaturan tombol atau button, dan juga pada pemilihan warna yang digunakan. Untuk mengubah background yang digunakan dengan cara mengubah pada bagian

`android:background="@drawable/baju"`. “baju” disini merupakan nama file yang terdapat pada folder drawable yang digunakan untuk background `activity_register`. Hal ini juga dilakukan dalam mengubah background activity yang lainnya.

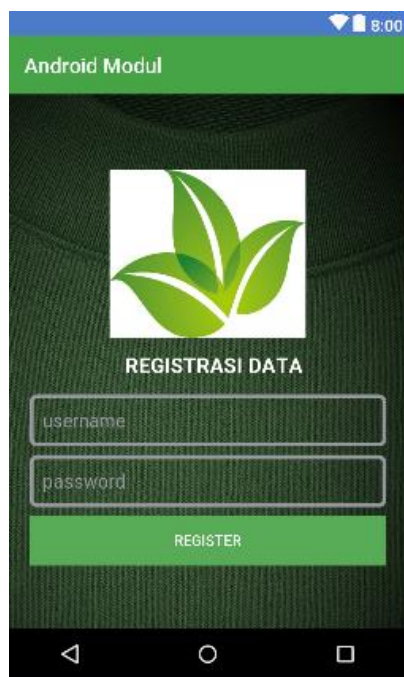
Untuk mengubah warna tema, button, dan lainnya yang dianggap perlu yaitu dengan mengubah pada folder `res/values/colors.xml`. Daftar warna yang dibutuhkan di list pada folder tersebut. Dalam hal ini praktikan mengubah warna sehingga menjadi sedemikian rupa. Berikut adalah tampilan-tampilan baru pada aplikasi ini,



Gambar 3.46 Tampilan `activity_splashscreen.xml`



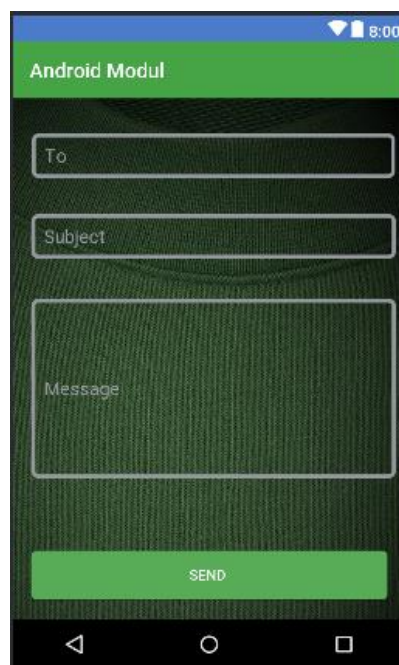
Gambar 3.47 Tampilan activity_login.xml



Gambar 3.48 Tampilan activity_register.xml



Gambar 3.49 Tampilan activity_main.xml



Gambar 3.50 Tampilan activity_email.xml



Gambar 3.51 Tampilan activity_about.xml

3.6.4 Membuat Menu Update dan Delete Untuk User

Membuat button edit pada activity_main.xml. Jika button edit ditekan maka nantinya akan muncul halaman edit data user. Berikut adalah listing code untuk menambah button edit pada activity_main.xml

```
<Button
    android:id="@+id/btnEditMain"
    android:layout_width="65dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="36dp"
    android:layout_marginEnd="8dp"
    android:layout_marginBottom="8dp"
    android:background="@drawable/roundedbutton"
    android:text="Edit"
    android:textColor="@color/white"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="@+id/cardView"
    app:layout_constraintHorizontal_bias="0.371"
    app:layout_constraintStart_toEndOf="@+id/btnLogoutMain"
    app:layout_constraintTop_toTopOf="@+id/btnLogoutMain"
    app:layout_constraintVertical_bias="0.111" />
```



Gambar 3.52 Tampilan activity_main.xml dengan button edit

Langkah selanjutnya yaitu membuat ActivityEdit.java dengan listing code seperti berikut,

```
package com.example.mypc.aplikasimodul2kel37;

import android.content.DialogInterface;
import android.content.Intent;
import android.content.SharedPreferences;
import android.support.v7.app.AlertDialog;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.widget.Button;
import android.widget.EditText;

public class EditActivity extends AppCompatActivity {

    private User userModel;
    private EditText etUserAbout, etPassAbout;
    private Button btnUpdate, btnDelete;
    DatabaseHandler cek = LoginActivity.getPresenter();
    User usr = LoginActivity.getUser();
    String name = usr.getUsername();
    String pass = usr.getPassword();
```

```

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_edit);
    initView();
    update();
    delete();
    deletePreference();
    etUserAbout.setText(name);
    etPassAbout.setText(pass);
    initUser();
}

private void initView() {
    etUserAbout = findViewById(R.id.etUserAbout);
    etPassAbout = findViewById(R.id.etPassAbout);
    btnUpdate = findViewById(R.id.btnUpdate);
    btnDelete = findViewById(R.id.btnDelete);
}

private void initUser() {
    name = etUserAbout.getText().toString();
    pass = etPassAbout.getText().toString();

    usr.setUsername(name);
    usr.setPassword(pass);
}

private void update() { //untuk update data
    btnUpdate.setOnClickListener(v -> {
        initUser();
        cek.updateMahasiswa(usr);
        Intent reg = new Intent(this, MainActivity.class);
        startActivity(reg);
        finish();
    });
}

private void delete() { //untuk delete data
    btnDelete.setOnClickListener(view ->
showAlertDialog());
}

public void showAlertDialog() {
    new AlertDialog.Builder(this)
        .setMessage("Apa ingin menghapus Akun ini?")
        .setCancelable(false)
        .setPositiveButton("Yes", new
DialogInterface.OnClickListener() {

```

```

        public void onClick(DialogInterface dialog,
int id) {
            deletePreference();
            Intent login = new
Intent(EditActivity.this, LoginActivity.class);
            startActivity(login);
            cek.deleteModel(usr);
            finish();
        }
    })
    .setNegativeButton("No", null)
    .show();
}

private void deletePreference() {
    SharedPreferences preferences =
getSharedPreferences("LoginPreference", MODE_PRIVATE);
    preferences.edit().remove("username").commit();
    preferences.edit().remove("password").commit();
}
@Override
public void onBackPressed() {
    back();
}

private void back() {
    Intent reg = new Intent(this, MainActivity.class);
    startActivity(reg);
    finish();
}
}

```

Pada *sourcecode* tersebut, dibuat metode/fungsi untuk button pada halaman menu edit. Pada halaman edit ini, nantinya data user dapat diupdate maupun dihapus. Di mana pada update kita bisa mengubah username dan password sesuai dengan yang kita mau. Selanjutnya kita juga bisa menghapus username dan password yang telah terdaftar, dan ketika username dan password yang sudah di hapus tidak akan bisa kita gunakan untuk login kembali, kecuali kita regis lagi. Berikut adalah listing code `activity_edit.xml`

```

<?xml
version="1.0"
encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:tools="http://schemas.android.com/tools"

```

```

android:layout_width="match_parent"
android:layout_height="match_parent"
android:background="@drawable/seagreen"
tools:context=".MainActivity">

<Button
    android:id="@+id/btnUpdate"
    android:layout_width="140dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="36dp"
    android:layout_marginTop="8dp"
    android:layout_marginBottom="44dp"
    android:background="@color/colorPrimaryDark"
    android:text="Update"
    android:textColor="@color/white"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/etPassAbout"
    app:layout_constraintVertical_bias="1.0" />

<Button
    android:id="@+id/btnDelete"
    android:layout_width="140dp"
    android:layout_height="wrap_content"
    android:layout_marginTop="8dp"
    android:layout_marginEnd="40dp"
    android:layout_marginBottom="44dp"
    android:background="@color/colorPrimaryDark"
    android:text="DELETE"
    android:textColor="@color/white"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/etPassAbout"
    app:layout_constraintVertical_bias="1.0" />

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginStart="36dp"
    android:layout_marginTop="72dp"
    android:text="Username"
    android:textSize="24sp"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="34dp"
    android:layout_marginStart="36dp"

```

```

        android:layout_marginTop="96dp"
        android:text="Password"
        android:textSize="24sp"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView3" />

<EditText
    android:id="@+id/etUserAbout"
    android:layout_width="280dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="8dp"
    android:layout_marginTop="124dp"
    android:layout_marginEnd="8dp"
    android:ems="10"
    android:inputType="textEmailAddress"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintHorizontal_bias="0.318"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toTopOf="parent" />

<EditText
    android:id="@+id/etPassAbout"
    android:layout_width="280dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="36dp"
    android:layout_marginTop="80dp"
    android:ems="10"
    android:inputType="textPassword"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/etUserAbout"
/>

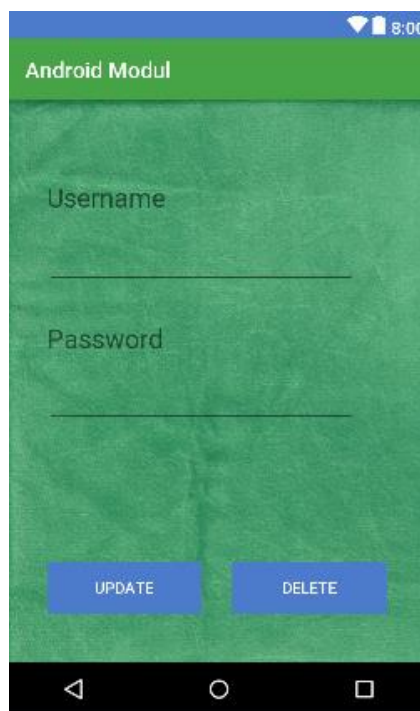
</android.support.constraint.ConstraintLayout>

```

Sourcecode activity_edit.xml digunakan untuk membuat tampilan pada halaman edit. Pada halaman ini terdapat textview untuk username dan password, textedit yang digunakan untuk mengedit username maupun password, dan juga terdapat button update dan delete.



Gambar 3.53 Tampilan activity_main.xml



Gambar 3.54 Tampilan activity_edit.xml

3.6.5 Membatasi Password Minimal 6 Karakter

```
private void initDataHandler() {
    initUser();
    String pass = etPasswordRegister.getText().toString();
    if (TextUtils.isEmpty(pass) || pass.length() < 6) {
        etPasswordRegister.setError("You must have 6 characters
in your password");
        return;
    } else {
        databaseHandler = new DatabaseHandler(this);
        databaseHandler.addUser(usermodel);
        User model = databaseHandler.getMahasiswa(1);
        Log.e("record", model.getUsername());
        Intent admin = new Intent(RegisterActivity.this,
LoginActivity.class);
        startActivity(admin);
        finish();
    }
}
```

Ditambahkan sourcecode diatas pada ActivityRegister.java. Source code diatas digunakan untuk emngatur agar password yang diisikan oleh user saat register data minimal berjumlah 6 karakter. Jika password yang dimasukkan kurang dari 6 karakter maka akan muncul tulisan “You must have 6 characters in your password” dan akan gagal menambahkan data pada menu register.



Gambar 3.55 Tampilan Halaman Registrasi

(Link Tugas: <https://github.com/agustiawan0798/modul2kelompok37.git>)

3.7 Kesimpulan

1. SharedPreferences digunakan untuk menyimpan database username dan password user.
2. SharedPreferences mirip seperti database yang lainnya namun digunakan untuk menyimpan data dengan skala kecil dan sederhana.
3. Ketika membuat activity pada android, maka akan secara otomatis terbuat activity.xml pada layout.
4. Class Database Handler berfungsi untuk melakukan operasi CRUD (Create, Read, Update, dan Delete).
5. Untuk membuat suatu resources yang nantinya dapat dipakai secara berulang-ulang seperti roundedbutton, dapat dibuat suatu Drawable resources file pada res/drawable dengan cara klik kanan dan pilih new kemudian pilih Drawable resource file.
6. Pada res/values/colors.xml merupakan suatu lokasi yang digunakan untuk database warna yang akan digunakan dan warna tersebut dapat diberi nama agar lebih mudah untuk digunakan pada kelas-kelas yang lain.
7. AndroidManifest.xml menyediakan informasi penting tentang aplikasi ke sistem Android yang harus dimiliki sistem agar bisa menjalankan setiap kode aplikasi.
8. Background pada suatu halaman dapat diubah dengan mudah dengan cara menambahkan atau mengubah pada `android:background="@drawable/nama_file_background"` di layout halaman yang ingin kita ubah backgroundnya.