

BAB VI

Android: Database (Firebase)

1. Tujuan

Mahasiswa mampu membuat aplikasi CRUD pada database *Firebase*

2. Tools

Android Studio, Realtime Database Firebase

3. Dasar Teori

Firestore

Firestore adalah suatu layanan dari [Google](https://www.google.com) yang digunakan untuk mempermudah para pengembang aplikasi dalam mengembangkan aplikasi. Dengan adanya Firestore, pengembang aplikasi bisa fokus mengembangkan aplikasi tanpa harus memberikan usaha yang besar. Dua fitur yang menarik dari Firestore yaitu Firestore Remote Config dan Firestore Realtime Database. Selain itu terdapat fitur pendukung untuk aplikasi yang membutuhkan pemberitahuan yaitu Firestore Notification.

Fitur-Fitur Firestore

- **Authentication** : Sekarang sebagian besar aplikasi ingin mengetahui identitas penggunanya sehingga nanti aplikasi dapat menyimpan data pengguna secara aman di cloud dan memberikan pengalaman personal yang sama di setiap perangkat pengguna. Fitur ini menyediakan layanan backend dengan SDK yang mudah dan siap digunakan untuk mengautentikasi pengguna ke aplikasi anda. Jadi dengan menggunakan fitur ini kita bisa membuat login menggunakan gmail, facebook, twitter dan lainnya.
- **Hosting** : Firestore Hosting yaitu layanan hosting konten web yang berkelas produksi untuk para pengembang aplikasi. Kita bisa menerapkan aplikasi web serta menyajikan konten statis ataupun dinamis ke CDN global secara cepat hanya dengan satu perintah saja
- **Cloud Storage** : Fitur ini dibuat untuk para pengembang aplikasi yang ingin menyimpan dan menampilkan konten buatan pengguna seperti image dan video.
- **Realtime Database** : Fitur Firestore Realtime Database yaitu database yang di host di cloud. Nantinya data akan disimpan sebagai JSON kemudian disinkronkan secara realtime ke setiap client yang sudah terhubung. Saat kita membuat aplikasi lintas platform dengan

SDK Android, IOS, maupun JavaScript, semua client akan berbagi sebuah instance realtime database lalu menerima update data terbaru secara otomatis.

Firestore Realtime Database

Firestore Realtime Database adalah database yang di-host di cloud. Data disimpan sebagai JSON dan disinkronkan secara realtime ke setiap klien yang terhubung. Ketika Anda membangun aplikasi lintas platform dengan Android SDK, iOS SDK, dan JavaScript SDK, semua klien Anda akan berbagi sebuah instance Realtime Database dan menerima update data terbaru secara otomatis.

Kemampuan utama Firestore Realtime Database

1. **Realtime** : Sebagai ganti permintaan HTTP biasa, Firestore Realtime Database menggunakan sinkronisasi data—setiap kali data berubah, semua perangkat yang terhubung akan menerima update dalam waktu milidetik. Memberikan pengalaman yang kolaboratif dan imersif tanpa perlu memikirkan kode jaringan.
2. **Offline** : Aplikasi Firestore tetap responsif bahkan saat offline karena SDK Firestore Realtime Database menyimpan data ke disk. Setelah konektivitas pulih, perangkat klien akan menerima setiap perubahan yang terlewat dan melakukan sinkronisasi dengan status server saat ini
3. **Dapat Diakses dari Perangkat Klien** : Firestore Realtime Database dapat diakses secara langsung dari perangkat seluler atau browser web; server aplikasi tidak diperlukan. Keamanan dan validasi data dapat diakses melalui Aturan Keamanan Firestore Realtime Database yang merupakan kumpulan aturan berbasis ekspresi dan dijalankan ketika data dibaca atau ditulis.
4. **Menskalakan di beberapa database** : Dengan Firestore Realtime Database pada paket harga Blaze, Anda dapat mendukung kebutuhan data aplikasi Anda pada skala tertentu dengan membagi data Anda di beberapa instance database di project Firestore yang sama. Menyederhanakan autentikasi dengan Firestore Authentication pada project Anda dan mengautentikasi pengguna di instance database Anda. Mengontrol akses ke data di tiap database dengan Aturan Firestore Realtime Database khusus untuk tiap instance database.

Cara kerja Firebase Realtime Database

Firebase Realtime Database memungkinkan Anda untuk mem-build aplikasi kolaboratif dan kaya fitur dengan menyediakan akses yang aman ke database, langsung dari kode sisi klien. Data disimpan di drive lokal. Bahkan saat offline sekalipun, peristiwa realtime terus berlangsung, sehingga pengguna akhir akan merasakan pengalaman yang responsif. Ketika koneksi perangkat pulih kembali, Realtime Database akan menyinkronkan perubahan data lokal dengan update jarak jauh yang terjadi selama klien offline, sehingga setiap perbedaan akan otomatis digabungkan.

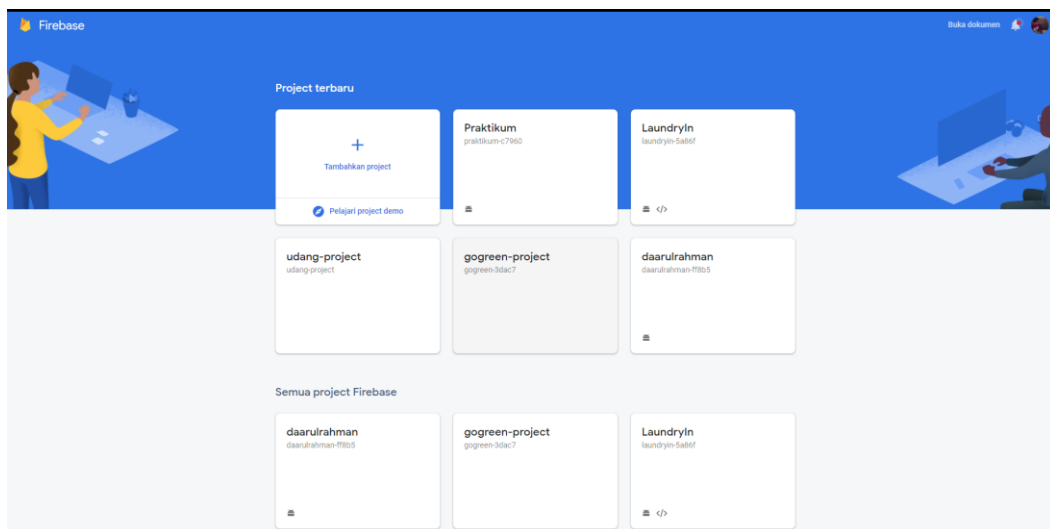
Realtime Database menyediakan bahasa aturan berbasis ekspresi yang fleksibel, atau disebut juga Aturan Keamanan Firebase Realtime Database, untuk menentukan metode strukturisasi data dan kapan data dapat dibaca atau ditulis. Ketika diintegrasikan dengan Firebase Authentication, developer dapat menentukan siapa yang memiliki akses ke data tertentu dan bagaimana mereka dapat mengaksesnya.

4. Langkah Praktikum

4.1 Persiapan

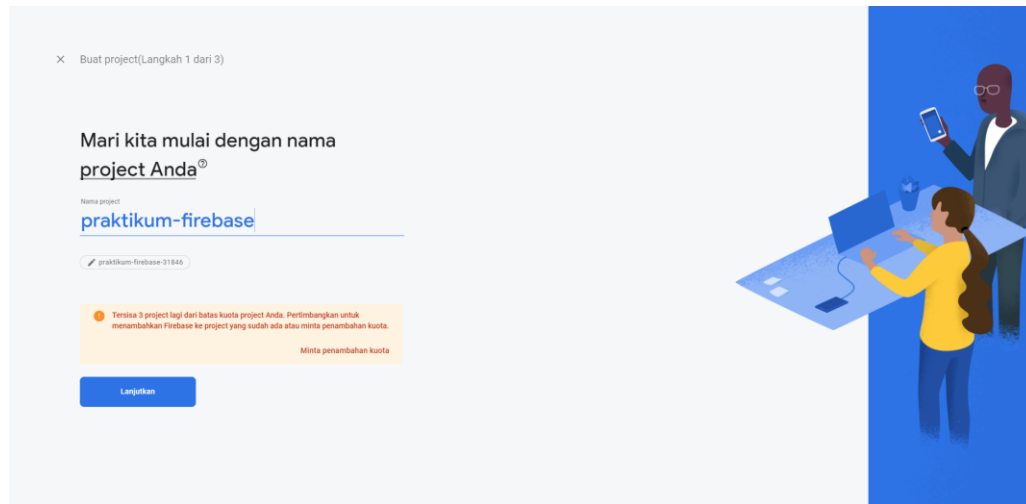
a. Masuk ke console firebase

Buka halaman pada link berikut <https://console.firebase.google.com/> apabila berhasil maka anda akan diminta untuk login menggunakan akun Gmail anda. Silahkan login maka akan muncul tampilan sebagai berikut.

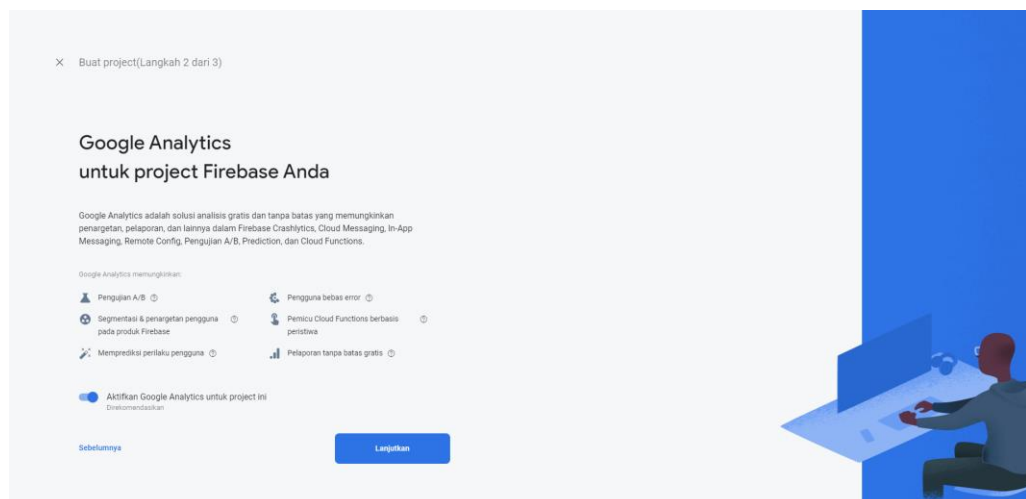


b. Membuat project baru di firebase

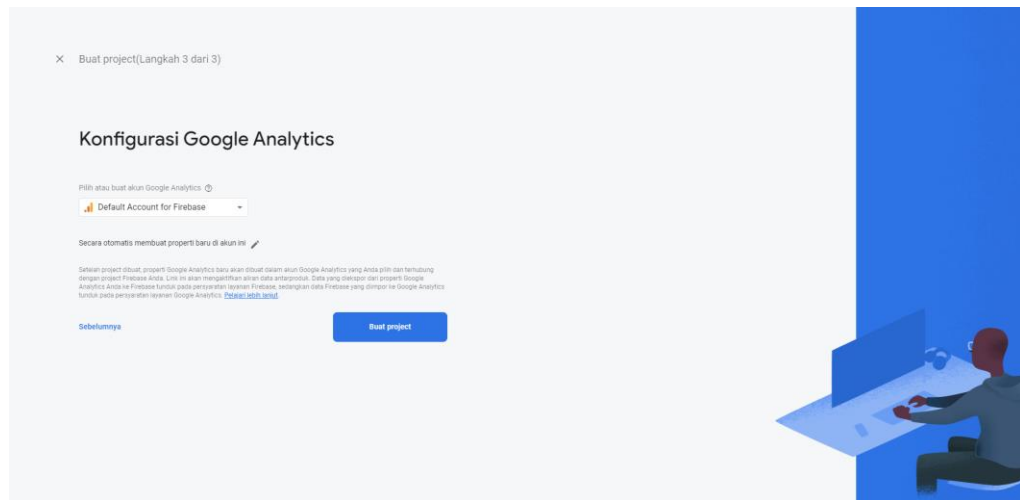
- Lalu pilih tambahkan project maka anda akan diminta untuk memberi nama project anda. Berilah nama pada project firebase anda 'praktikum-firebase' maka akan tampil sebagai berikut



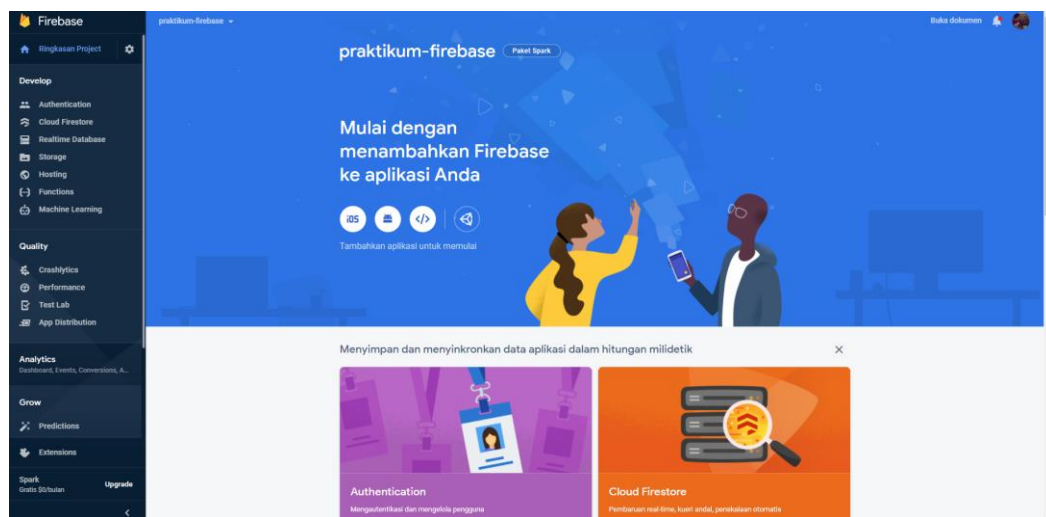
- Lalu pilih lanjutkan



- Lalu pilih ‘default account for firebase’ dan tekan buat project



- Tunggu hingga proses selesai. Apabila project berhasil dibuat maka tampilan akan seperti berikut, anda akan diarahkan kedalam console project di firebase

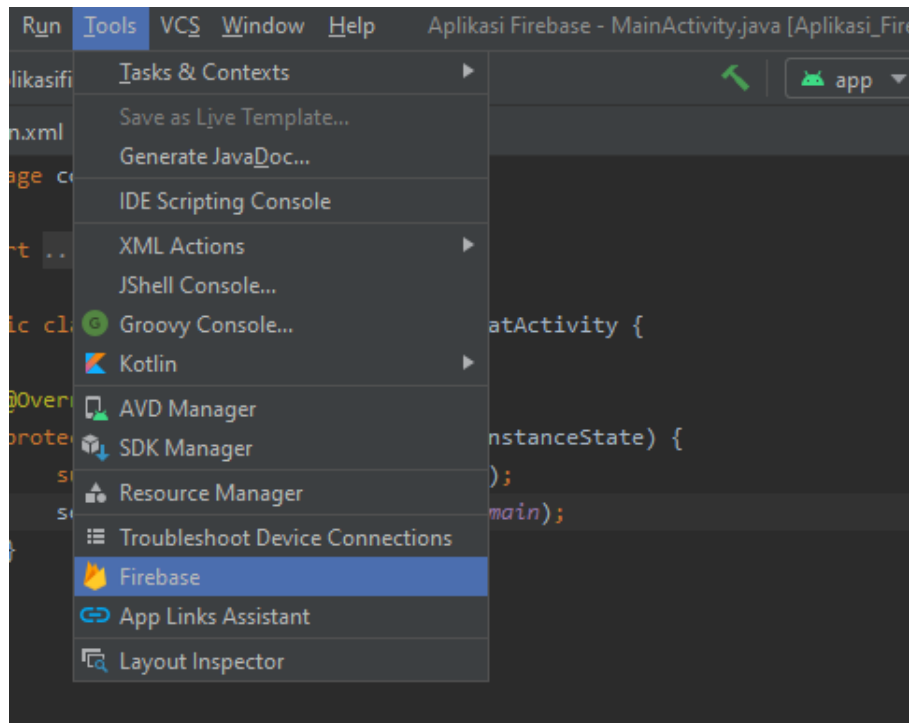


c. Membuat Project baru di Android Studio

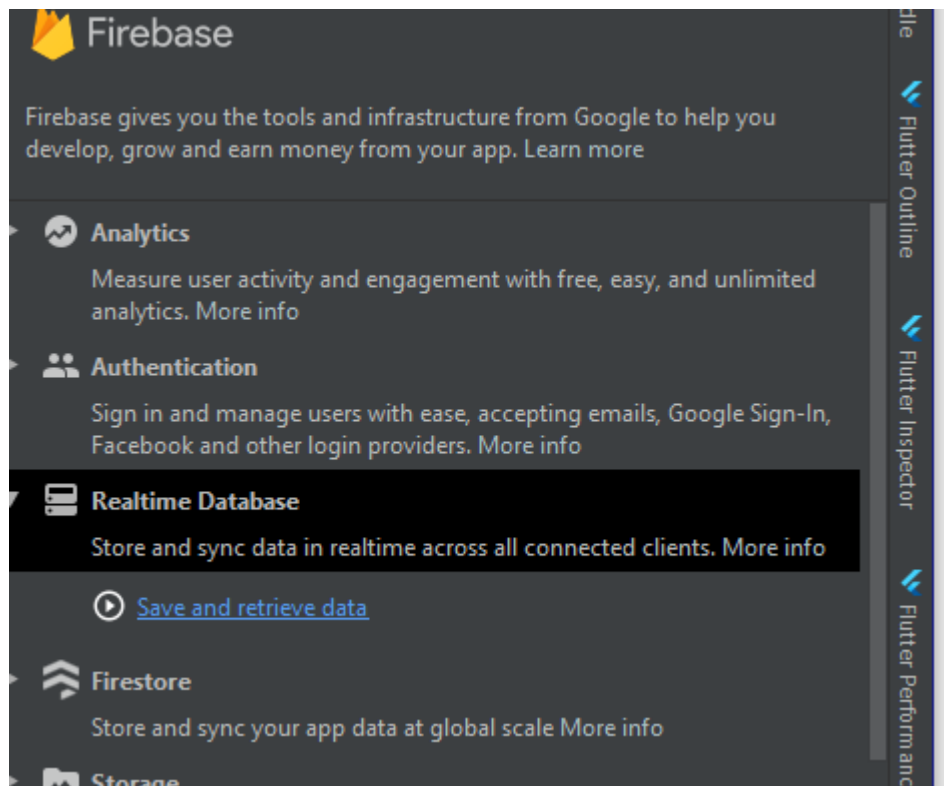
File→new project (pilih empty activity) →beri nama project kalian ‘Aplikasi Firebase’

d. Menghubungkan android studio dengan firebase dan membuat ‘*realtime database*’

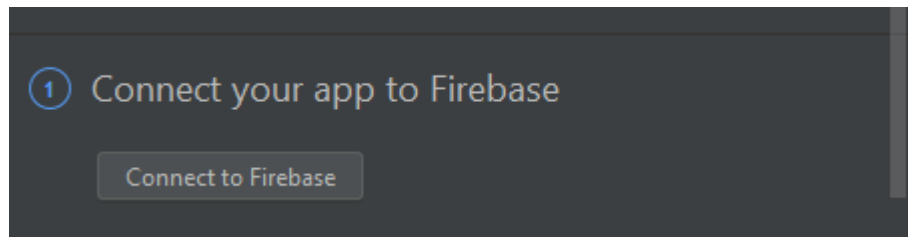
- Pilih 'tool' → 'firebase' pada topbar



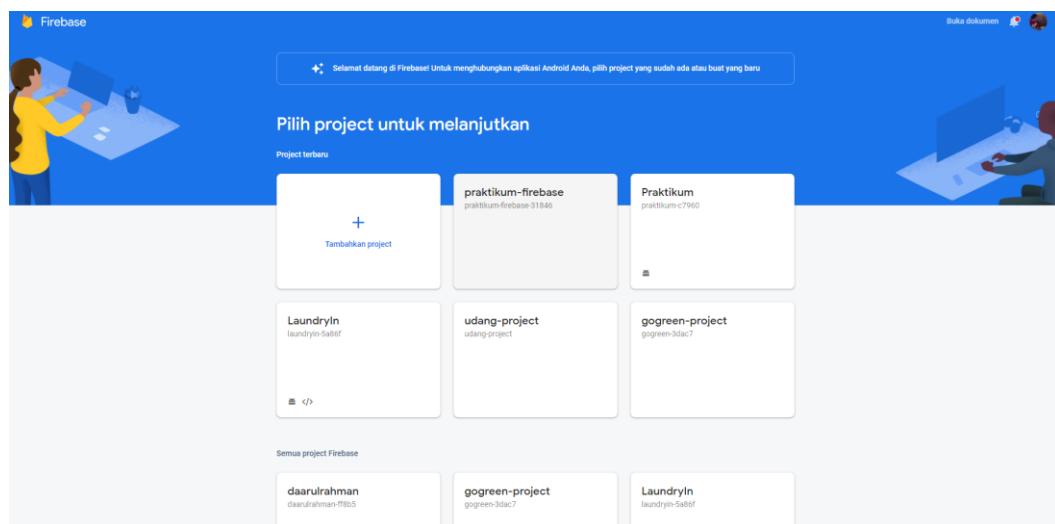
- Lalu pilih 'realtime database'



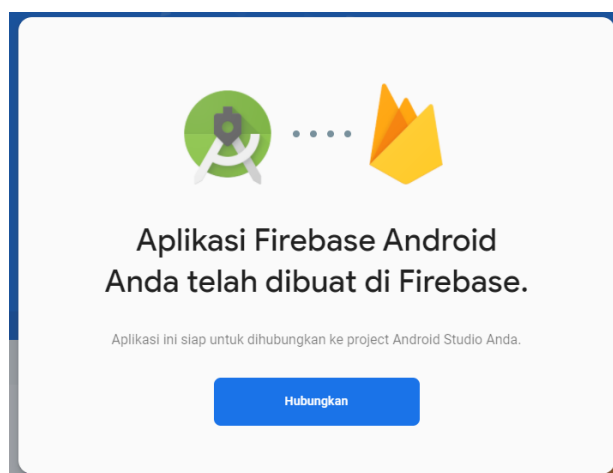
- Selanjutnya pilih ‘Connect your app to firebase’



- Maka anda akan diarahkan ke console firebase anda, lalu anda pilih project yang tadi telah dibuat.

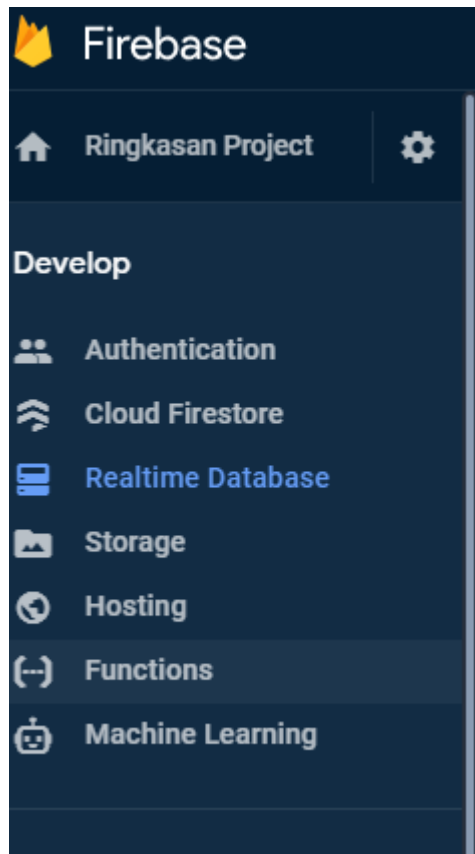


- Selanjutnya silahkan pilih hubungkan, maka project firebase anda telah terhubung dengan project android studio

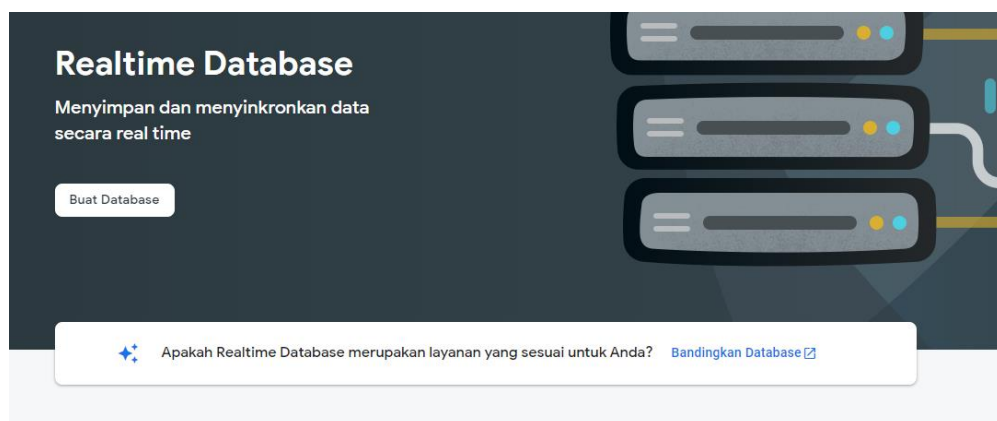


e. **Menambahkan Realtime Database kedalam aplikasi anda**

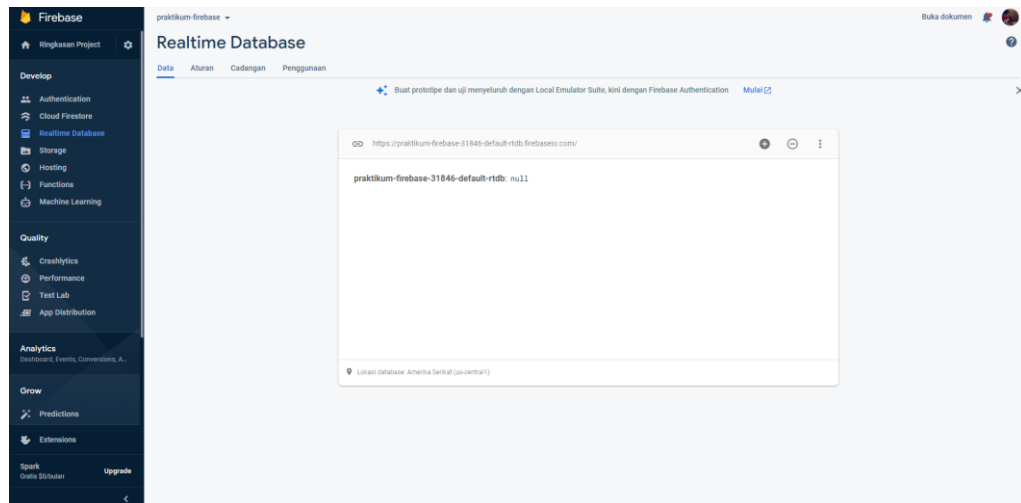
- Silahkan buka kembali consol firebase anda lalu pilih 'realtime database'



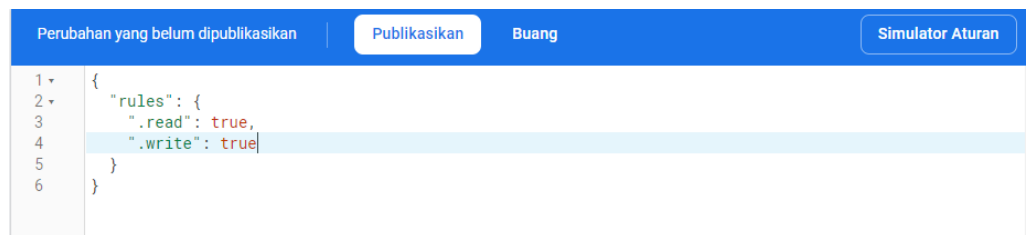
- Silahkan pilih 'Buat database' → lanjutkan



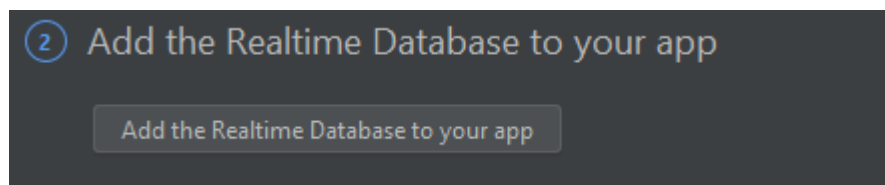
- Jika berhasil maka akan muncul tampilan sebagai berikut, lalu pilih tab aturan dan kita ubah aturan {read : true; write:true;}hal ini dilakukan agar aplikasi kita nanti dapat melakukan proses read and write pada 'realtime database'



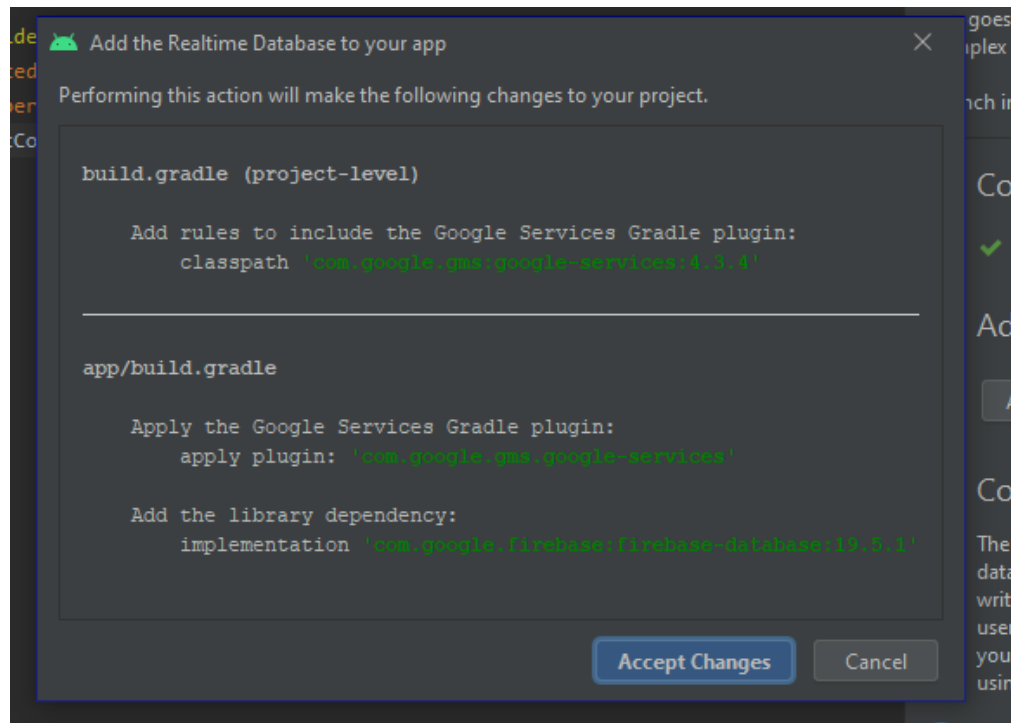
- Jika berhasil maka akan tampil sebagai berikut



- Selanjutnya tambahkan realtime database kedalam aplikasi anda



- Tambahkan dependencies dan rule yang dibutuhkan



4.2 Konfigurasi pada project android studio

- Penambahan dependencies yang dibutuhkan

Tambahkan dependencies berikut

```

{
  ...
  implementation "androidx.cardview:cardview:1.0.0"
  implementation 'androidx.recyclerview:recyclerview:1.1.0'
  ...
}
  
```

Pada project kali ini kita akan menggunakan cardview dan recyclerview olehkarena itu kite perlu menambahkan dependencies diatas.

- Penambahan permission pada AndroidManifest.xml

```

<manifest>
  ...
  <uses-permission android:name="android.permission.INTERNET"
    />
  ...
  
```

```
</manifest>
```

Pada project kali ini kita menggunakan 'relatime database' yang dimana database tersebut dapat diakses apabila terhubung dengan *INTERNET* oleh karena itu kita melakukan penambahan permission pada manifest agar aplikasi kita dapat terhubung dengan Internet

c. Mempersiapkan style dan icon yang akan digunakan

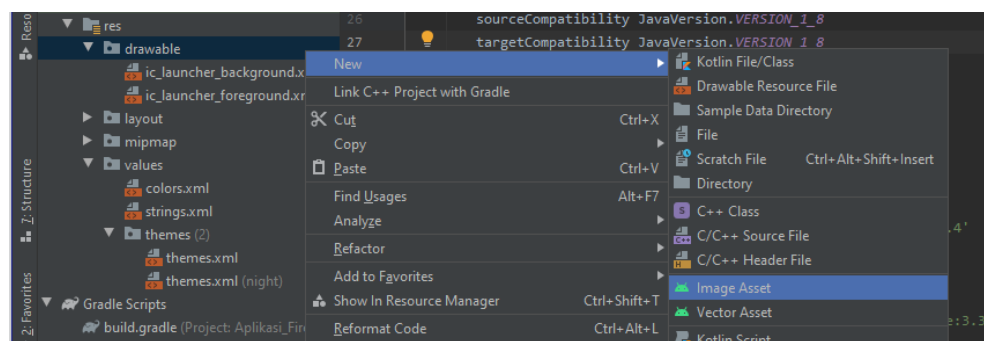
- Menghilangkan 'Action bar' (ini sifatnya optional)

Buka theme.xml dan theme.xml(night) lalu ubah menjadi seperti ini

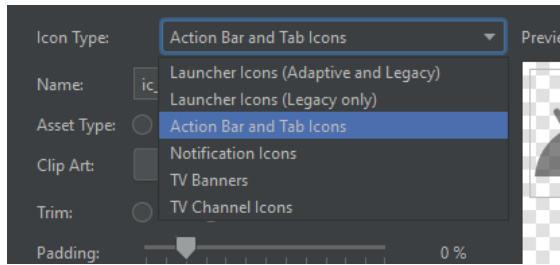
```
<resources xmlns:tools="http://schemas.android.com/tools">
  <!-- Base application theme. -->
  <style name="Theme.TestFirebase" parent="Theme.MaterialComponents.Light.NoActionBar">
    <!-- Primary brand color. -->
    <item name="colorPrimary">@color/purple_200</item>
    <item name="colorPrimaryVariant">@color/purple_700</item>
    <item name="colorOnPrimary">@color/black</item>
    <!-- Secondary brand color. -->
    <item name="colorSecondary">@color/teal_200</item>
    <item name="colorSecondaryVariant">@color/teal_200</item>
    <item name="colorOnSecondary">@color/black</item>
    <!-- Status bar color. -->
    <item name="android:statusBarColor" tools:targetApi="1">?attr/colorPrimaryVariant</item>
    <!-- Customize your theme here. -->
  </style>
</resources>
```

Cukup ubah yang diberi tanda merah saja, lakukan pada theme.xml dan theme.xml(night)

- Memasukkan icon yang dibutuhkan seperti ('person', 'people', 'add circle', 'arrow back')



Pilih image asset, selanjutnya pilih (Icon type : Action Bar and Tab Icons)



Selanjutnya select icon dengan cara menekan 'clip art' carilah icon yang bernama ('person', 'people', 'add circle', dan 'arrow back') lalu beri nama secara berurutan ('ic_mhs', 'ic_list_mhs', 'ic_add_mhs', dan 'ic_back') lalu beri warna sesuai dengan yang kalian inginkan, pada kasus kali ini kita akan memberi warna dengan (#3700B3)

- Membuat custom frame

Klik kanan pada 'drawable' → new → Drawable Resources File, lalu beri nama dengan 'frame'. Jika berhasil maka akan terbuat frame.xml. selanjutnya isikan code berikut pada frame.xml

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

    <!-- ini merupakan background frame yang dapat dicustom -->
    <corners
        android:radius="10dp"/>
    <solid
        android:color="@android:color/transparent"/>
    <stroke
        android:color="@color/purple_700"
        android:width="2dp"/>

</shape>
```

Frame ini akan kita gunakan pada list mahasiswa nantinya.

4.3 Mendesain activity_main.xml

Ubahlah activity_main.xml menjadi seperti berikut

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    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:background="#f2f2f2">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_gravity="center">

        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:padding="8dp"
            android:layout_marginBottom="10dp"
            android:layout_centerHorizontal="true">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:layout_marginTop="20dp">

                <androidx.cardview.widget.CardView

    xmlns:card_view="http://schemas.android.com/apk/res-auto"
        android:id="@+id/add_mhs"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="15dp"
```

```

        android:layout_marginRight="15dp"
        card_view:cardCornerRadius="10dp"
        android:clickable="true"
        android:onClick="pindah"
        android:background="@android:color/white"

        android:foreground="?android:attr/selectableItemBackground">

        <LinearLayout
            android:layout_width="150dp"
            android:layout_height="150dp"
            android:paddingTop="20dp"
            android:orientation="vertical">

            <ImageView
                android:layout_width="70dp"
                android:layout_height="70dp"
                android:layout_gravity="center"
                android:src="@drawable/ic_add_mhs"/>

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:textSize="12sp"
                android:layout_gravity="center"
                android:layout_marginTop="8dp"

                android:textColor="@android:color/black"
                android:text="Tambah Mahasiswa"/>
            </LinearLayout>

        </androidx.cardview.widget.CardView>

        <androidx.cardview.widget.CardView

xmlns:card_view="http://schemas.android.com/apk/res-auto"
        android:id="@+id/list_mhs"
        android:layout_width="wrap_content"

```

```

        android:layout_height="wrap_content"
        android:layout_marginLeft="15dp"
        android:layout_marginRight="15dp"
        android:clickable="true"

        android:foreground="?android:attr/selectableItemBackground"
        android:onClick="pindah"
        card_view:cardCornerRadius="10dp">

        <LinearLayout
            android:layout_width="150dp"
            android:layout_height="150dp"
            android:paddingTop="20dp"
            android:orientation="vertical">

            <ImageView
                android:layout_width="70dp"
                android:layout_height="70dp"
                android:layout_gravity="center"

                android:src="@drawable/ic_list_mhs"/>

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:textSize="12sp"
                android:layout_gravity="center"
                android:layout_marginTop="8dp"

                android:textColor="@android:color/black"
                android:text="List Mahasiswa"/>
            </LinearLayout>

        </androidx.cardview.widget.CardView>

    </LinearLayout>
</LinearLayout>

```

```
        </RelativeLayout>

    </ScrollView>
```

Pada MainActivity terdapat dua buah menu dimana menu pertama merupakan ‘tambah mahasiswa’ dan menu kedua merupakan ‘list mahasiswa’. Selanjutnya kita tambahkan beberapa baris code pada **MainActivity.java** tambahkan procedure *pindah()*

```
...
...

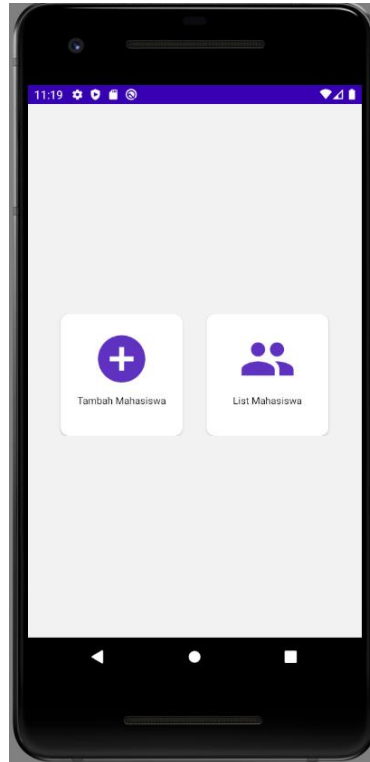
public class MainActivity extends AppCompatActivity {
    ...
    protected void onCreate(Bundle savedInstanceState) {
        ...
    }

    ...
}
```

Procedure **pindah()** akan dijalankan apabila salah satu menu dipilih. MainActivity akan berpindah ke activity lain sesuai dengan menu yang dipilih. A

4.4 Tambahkan 2 activity baru yaitu (*TambahMhsActivity*, dan *ListMhsActivity*)

Coba jalankan, apabila berhasil maka akan seperti berikut



4.5 Melakukan Insert data ke database

- Membuat form mahasiswa

Ubahlah code yang ada di **activity_tambah_mhs.xml** menjadi seperti berikut

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView
    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=".TambahMhsActivity"
    android:padding="8dp">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical">
        <ImageButton
            android:id="@+id/btn_back"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:src="@drawable/ic_back"
            android:background="@android:color/transparent"
            android:clickable="true"
```

```
android:foreground="?android:attr/selectableItemBackground"/>
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Tambah Mahasiswa"
    android:textSize="20sp"
    android:layout_marginBottom="20dp"
    android:textColor="@android:color/black"
    android:layout_gravity="center_horizontal"/>
```

```
<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="8dp"
    android:layout_gravity="center_horizontal">
```

```
<TextView
    android:layout_width="65dp"
    android:layout_height="wrap_content"
    android:text="NIM"
    android:textColor="@android:color/black"
    android:layout_marginRight="10dp"/>
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textColor="@android:color/black"
    android:text=":"/>
```

```
<EditText
    android:id="@+id/et_nim"
    android:layout_width="200dp"
    android:layout_height="wrap_content"
    android:hint="Masukkan NIM"
    android:textSize="12sp"
    android:layout_marginLeft="5dp"/>
```

```
</LinearLayout>
```

```
<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="20dp"
    android:layout_gravity="center_horizontal">
```

```
<TextView
    android:layout_width="65dp"
    android:layout_height="wrap_content"
    android:text="Nama"
    android:textColor="@android:color/black"
    android:layout_marginRight="8dp"/>
```

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textColor="@android:color/black"
    android:text=":"/>
```

```

        <EditText
            android:id="@+id/et_nama"
            android:layout_width="200dp"
            android:layout_height="wrap_content"
            android:hint="Masukkan Nama"
            android:textSize="12sp"
            android:layout_marginLeft="5dp"/>
    </LinearLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="20dp"
        android:layout_gravity="center_horizontal">

        <TextView
            android:layout_width="65dp"
            android:layout_height="wrap_content"
            android:text="Jurusan"
            android:textColor="@android:color/black"
            android:layout_marginRight="8dp"/>

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:textColor="@android:color/black"
            android:text=":"/>

        <EditText
            android:id="@+id/et_jurusan"
            android:layout_width="200dp"
            android:layout_height="wrap_content"
            android:hint="Masukkan Jurusan"
            android:textSize="12sp"
            android:layout_marginLeft="5dp"/>
    </LinearLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="20dp"
        android:layout_gravity="center_horizontal">

        <TextView
            android:layout_width="65dp"
            android:layout_height="wrap_content"
            android:text="Angkatan"
            android:textColor="@android:color/black"
            android:layout_marginRight="8dp"/>

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:textColor="@android:color/black"
            android:text=":"/>

        <EditText
            android:id="@+id/et_angkatan"
            android:layout_width="200dp"

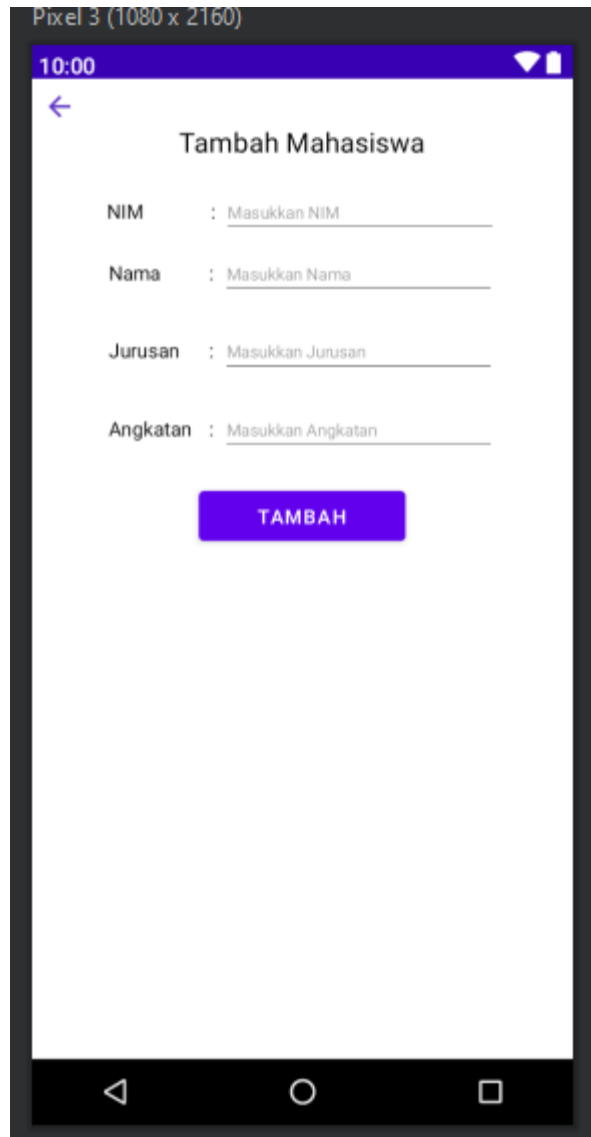
```

```
        android:layout_height="wrap_content"
        android:hint="Masukkan Angkatan"
        android:textSize="12sp"
        android:layout_marginLeft="5dp"/>
    </LinearLayout>

    <Button
        android:id="@+id/btn_tambah"
        android:layout_width="150dp"
        android:layout_height="wrap_content"
        android:text="Tambah"
        android:layout_gravity="center_horizontal"/>

    </LinearLayout>
</ScrollView>
```

Sehingga tampilan **activity_tambah_mhs.xml** akan seperti berikut



b. Membuat model 'Mahasiswa'

Buat java class baru bernama Mahasiswa.java, class ini akan digunakan sebagai model untuk proses di database, baik itu insert, edit, maupun delete. Buatlah atribut (nim,nama,jurusan,angkatan) yang bertipe '*String*', lalu buatlah constructor, setter dan getter. Maka isi dari Mahasiswa.java akan seperti berikut.

```
public class Mahasiswa {  
    private String nim;  
    private String nama;  
    private String jurusan;  
    private String angkatan;  
  
    public Mahasiswa(String nim, String nama, String jurusan,  
String angkatan) {  
        this.nim = nim;  
    }  
}
```

```
        this.nama = nama;
        this.jurusan = jurusan;
        this.angkatan = angkatan;
    }

    public Mahasiswa(String nim, String nama) {
        this.nim = nim;
        this.nama = nama;
    }

    public String getNim() {
        return nim;
    }

    public void setNim(String nim) {
        this.nim = nim;
    }

    public String getNama() {
        return nama;
    }

    public void setNama(String nama) {
        this.nama = nama;
    }

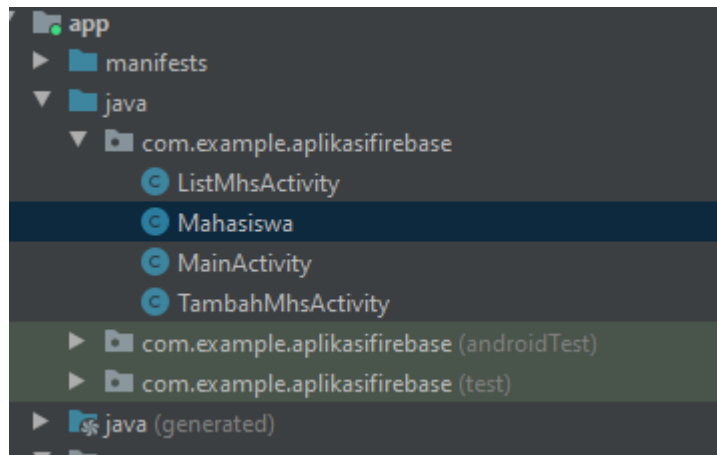
    public String getJurusan() {
        return jurusan;
    }

    public void setJurusan(String jurusan) {
        this.jurusan = jurusan;
    }

    public String getAngkatan() {
        return angkatan;
    }

    public void setAngkatan(String angkatan) {
        this.angkatan = angkatan;
    }
}
```

Sehingga struktur file pada project kita akan seperti berikut



- c. Melakukan inisialisasi variable dan object yang dibutuhkan pada

TambahMhsActivity.java

Tambahkan beberapa baris code berikut pada **TambahMhsActivity.java**

```
...
...
public class TambahMhsActivity extends AppCompatActivity{
    EditText etNIM,etNama,etJurusan,etAngkatan;
    Button btnTambah;
    ImageButton btnBack;

    FirebaseDatabase database;
    DatabaseReference myref;

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

        //inisialisasi variable dengan id yang ada di layout
        etNIM = findViewById(R.id.et_nim);
        etNama = findViewById(R.id.et_nama);
        etJurusan = findViewById(R.id.et_jurusan);
        etAngkatan = findViewById(R.id.et_angkatan);

        btnTambah = findViewById(R.id.btn_tambah);
        btnBack = findViewById(R.id.btn_back);

        //memanggil fungsi untuk input data ke firebase
        insertData();

        btnBack.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                finish();
            }
        });
    }
}
```

Variable 'myref' dan 'database' merupakan variable yang akan menghubungkan ke realtime database

d. Membuat method *insertData()*

Method ini digunakan untuk menginput data ke realtime database, tambahkan beberapa baris code pada **TambahMhsActivity.java** berikut

```
...
...
public class TambahMhsActivity extends AppCompatActivity{
    ...
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_tambah_mhs);
        ...
        ...
    }
    private void insertData() {
        //inisialisasi object ke firebase
        database = FirebaseDatabase.getInstance();
        myref = database.getReference("mhs");

        //melakukan sesuatu jika btn ditekan
        btnTambah.setOnClickListener(new View.OnClickListener()
        {
            @Override
            public void onClick(View view) {

                //mengambil data dari setiap edittext dan di
                ubah kedalam string
                final String nim = etNIM.getText().toString();
                final String nama = etNama.getText().toString();
                final String jurusan =
                etJurusan.getText().toString();
                final String angkatan =
                etAngkatan.getText().toString();

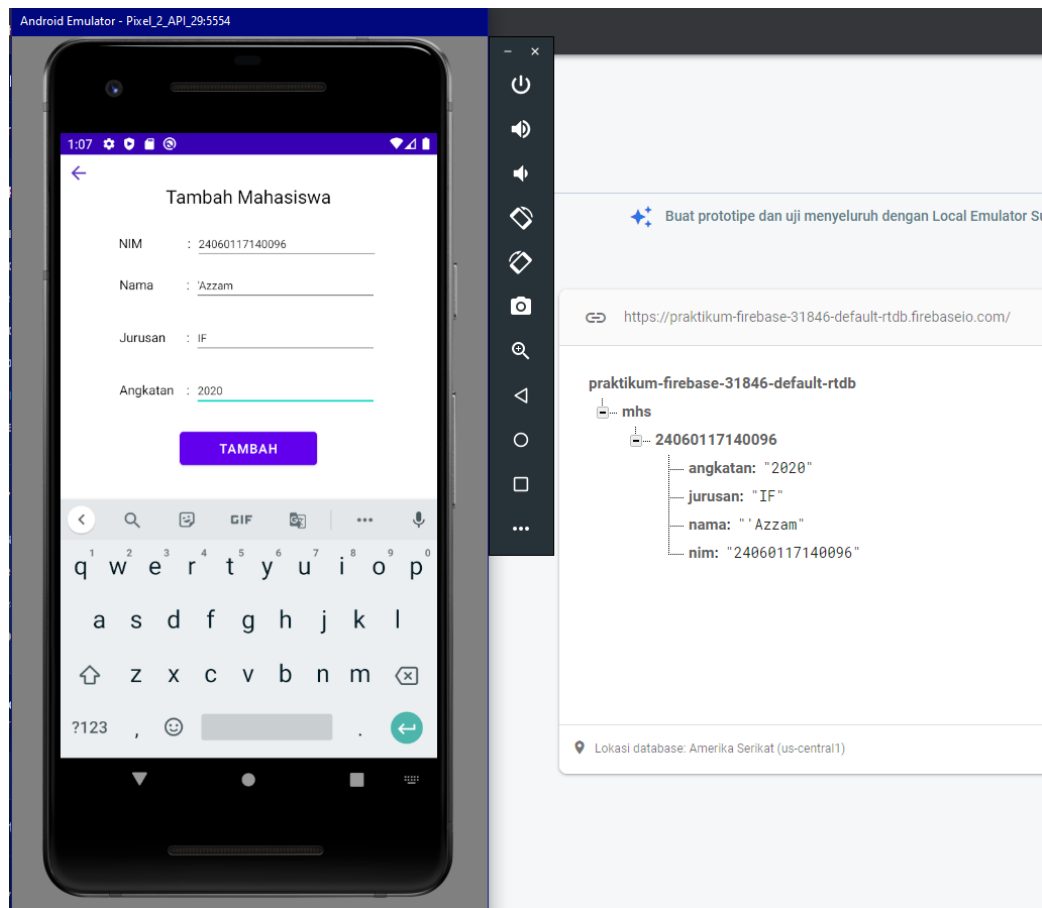
                //Validasi data yang diinput tidak boleh kosong
                if (nim.equals("") || nama.equals("") ||
                jurusan.equals("") || angkatan.equals("")){
                    Toast.makeText(TambahMhsActivity.this, "Data
                Harus diisi semua!!", Toast.LENGTH_SHORT).show();
                } else{
                    //memasukkan data - data tersebut kedalam
                    objek 'Mahasiswa'
                    Mahasiswa mhs = new
                    Mahasiswa(nim,nama,jurusan,angkatan);

                    //input data kedalam firebase

                    myref.child(nim).setValue(mhs).addOnSuccessListener(new
                    OnSuccessListener<Void>() {
```


Selanjutnya adalah memasukkan semua informasi yang ada di object 'Mahasiswa' kedalam realtime database. Perintah ini dapat ditemukan pada baris code `'myref.child(nim).setValue(mhs)'`

Coba jalankan dan coba masukkan data, jika berhasil maka akan tampil seperti berikut



Data akan secara realtime masuk kedalam database

4.6 Read data dari database

a. Membuat tampilan untuk **activity_list_mhs.xml**

Ubahlah code menjadi seperti berikut

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    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=".ListMhsActivity"
    android:orientation="vertical"
    android:padding="8dp">

    <ImageButton
        android:id="@+id/btn_back"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@drawable/ic_back"
        android:background="@android:color/transparent"
        android:clickable="true"

        android:foreground="?android:attr/selectableItemBackground"/>

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="List Mahasiswa"
        android:textSize="25sp"
        android:layout_gravity="center_horizontal"
        android:textColor="@android:color/black"
        android:layout_marginBottom="10dp"/>

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/recyclerview"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">
    </androidx.recyclerview.widget.RecyclerView>

</LinearLayout>
```

Pada code diatas kita menggunakan

‘<androidx.recyclerview.widget.RecyclerView...’ view ini yang digunakan untuk menampilkan data dari database dalam bentuk list

b. Membuat layout untuk stiap item dari list

Klik kanan pada layout → new → layout resources file lalu beri nama ‘item_mhs’

Lalu ubah item_mhs.xml menjadi seperti berikut

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

```

<!--Ini merupakan tampilan untuk setiap item pada sebuah
list-->

<LinearLayout

    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    android:layout_margin="8dp"
    android:background="@drawable/frame"
    android:clickable="true"

    android:foreground="?android:attr/selectableItemBackground
">

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

        <ImageView
            android:layout_width="80dp"
            android:layout_height="80dp"
            android:src="@drawable/ic_mhs"
            android:layout_gravity="center_vertical"/>

        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:orientation="vertical"
            android:layout_gravity="center_vertical">

            <TextView
                android:id="@+id/tv_nama"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Nama mahasiswa"
                android:textSize="20sp"
                android:textColor="@android:color/black"/>

            <TextView
                android:id="@+id/tv_nim"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="NIM Mahasiswa"
                android:layout_marginTop="5dp"/>

        </LinearLayout>

    </LinearLayout>

</LinearLayout>

```

Jika berhasil maka akan seperti berikut



- c. Tambah java class baru bernama **MyAdapter.java**

Class ini berguna untuk memproses data dalam bentuk *RecyclerView*.

Maka isi code dari MyAdapter.java adalah sebagai berikut

```
public class MyAdapter extends RecyclerView.Adapter {

    //List untuk menampung semua data mahasiswa dari firebase
    List<Mahasiswa> fetcgDataList;
    private OnItemClickListener mListener;

    public MyAdapter(List<Mahasiswa> fetcgDataList) {
        this.fetcgDataList = fetcgDataList;
    }

    @NonNull
    @Override
    public RecyclerView.ViewHolder onCreateViewHolder(@NonNull
    ViewGroup parent, int viewType) {
        View view =
        LayoutInflater.from(parent.getContext()).inflate(R.layout.item_mhs, par
        ent, false);
        ViewHolderClass viewHolderClass = new
        ViewHolderClass(view, mListener);
        return viewHolderClass;
    }
}
```

```

@Override
public void onBindViewHolder(@NonNull RecyclerView.ViewHolder
holder, int position) {
    ViewHolderClass viewHolderClass = (ViewHolderClass)holder;

    Mahasiswa mahasiswa = fetcgDataList.get(position);
    viewHolderClass.nim.setText(mahasiswa.getNim());
    viewHolderClass.nama.setText(mahasiswa.getNama());

}

@Override
public int getItemCount() {
    return fetcgDataList.size();
}

public class ViewHolderClass extends RecyclerView.ViewHolder{
    TextView nim,nama,jurusan,angkatan;

    public ViewHolderClass(@NonNull View itemView, final
OnItemClickListener listener) {
        super(itemView);
        nim=itemView.findViewById(R.id.tv_nim);
        nama=itemView.findViewById(R.id.tv_nama);

        //Dilakukan agar item pada list dapat berpindah ke
activity lain
        itemView.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                if (listener != null){
                    int position = getAdapterPosition();
                    if (position != RecyclerView.NO_POSITION){
                        listener.onItemClick(position);
                    }
                }
            }
        });
    }

}

public interface.OnItemClickListener{
    void onItemClick(int position);
}

public void setOnItemClickListener(OnItemClickListener listener){
    mListener = listener;
}
}

```

Langkah awal adalah melakukan inisialisasi variable. Pertama terdapat sebuah List dari 'Mahasiswa' yang bernama 'fetcgDataList', list ini berguna untuk menyimpan data mahasiswa dari database nantinya. Kedual adalah variable 'mListener'

merupakan variable yang nantinya digunakan sebagai listener apabila salasatu item dari list di pilih.

Kemudian pada 'onCreateViewHolder()' akan dihubungkan antara item_mhs.xml dengan recyclerView, jadi apabila ada data yang ter-retrieve dari database maka akan ditampilkan dalam bentuk list item_mhs.xml.

Kemudian pada 'onBindViewHolder()' data yang diperoleh akan di set kedalam TextView yang sudah kita buat pada item_mhs.xml.

Pada constructor 'ViewHolderClass()' mendefinisikan TextView pada item_mhs.xml berdasarkan id kedalam sebuah variable.

- d. Menampilkan data dari database ke aplikasi kita

Masuk kedalam **ListMhsActivity.java** lalu ubah isinya seperti berikut

```
public class ListMhsActivity extends AppCompatActivity {
    List<Mahasiswa> fetchData;
    RecyclerView recyclerView;
    MyAdapter myAdapter;
    DatabaseReference databaseReference;

    ImageButton btnBack;

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

        recyclerView = findViewById(R.id.recyclerview);
        recyclerView.setLayoutManager(new LinearLayoutManager(this));
        fetchData = new ArrayList<>();

        //inisialisasi objek untuk ke firebase
        databaseReference =
        FirebaseDatabase.getInstance().getReference("mhs");
        databaseReference.addListenerForSingleValueEvent(new
        ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot
            dataSnapshot) {
                //iterasi mengambil mahasiswa
                for (DataSnapshot ds : dataSnapshot.getChildren()){
                    //Mengambil value setiap child dari 'mahasiswa'
                    (nim,nama,jurusan,angkatan)
```

```

        String nim =
ds.child("nim").getValue().toString();
        String nama =
ds.child("nama").getValue().toString();
        Mahasiswa data = new Mahasiswa(nim,nama);
        fetchData.add(data);

    }
    myAdapter = new MyAdapter(fetchData);
    recyclerView.setAdapter(myAdapter);

    //Pindah activity apabila item pada list di click
    myAdapter.setOnItemClickListener(new
MyAdapter.OnItemClickListener() {
        @Override
        public void onItemClick(int position) {
            String nim = fetchData.get(position).getNim();
            startActivity(new
Intent(ListMhsActivity.this,EditMhsActivity.class).putExtra("nim",nim)
);
        }
    });

}

@Override
public void onCancelled(@NonNull DatabaseError
databaseError) {
    Toast.makeText(ListMhsActivity.this, "Maaf Terjadi
kesalahan... coba beberapa saat lagi!", Toast.LENGTH_SHORT).show();
}
});
btnBack = findViewById(R.id.btn_back);
btnBack.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        finish();
    }
});
}
}

```

Data dari database diambil menggunakan perintah berikut

```

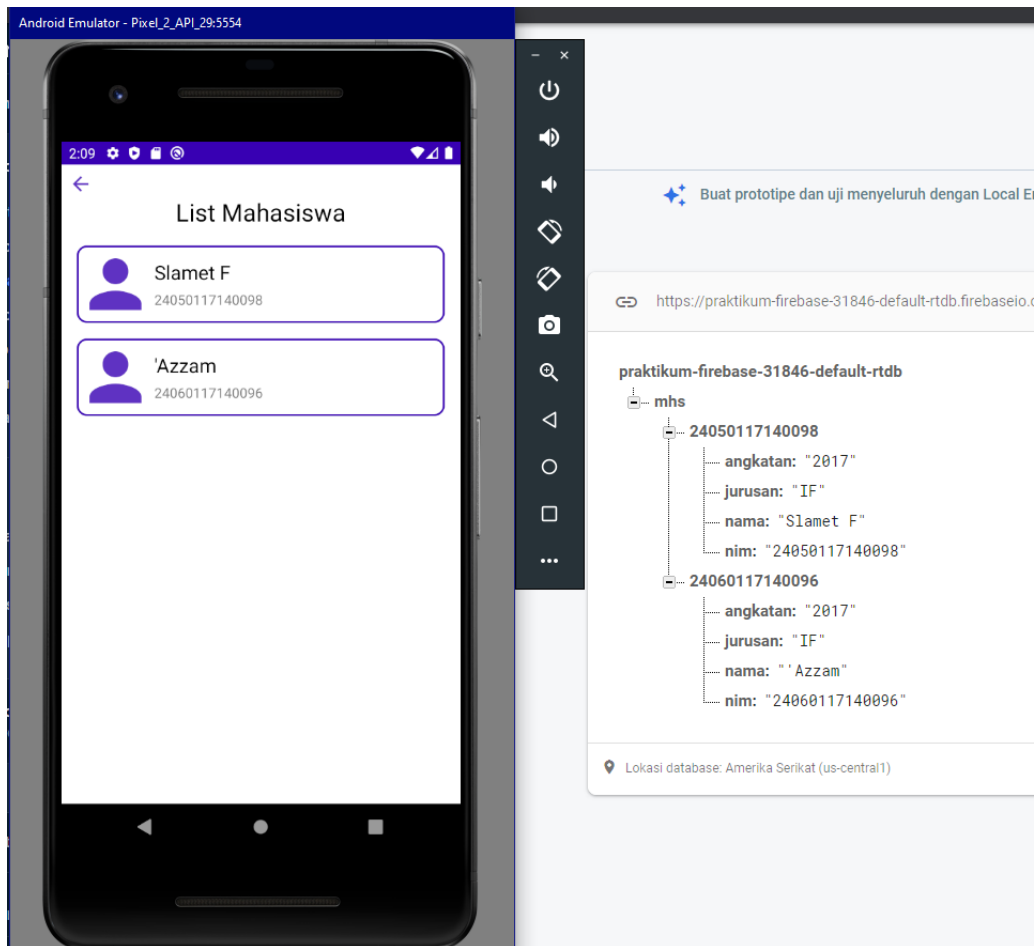
'databaseReference.addListenerForSingleValueEvent(new
ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot dataSnapshot)
{...}
});' apabila terjadi perubahan pada database maka akan melakukan proses

```


pengambilan data yang dapat kita temukan pada method `'onDataChange()'`. Data diambil menggunakan operasi perulangan for lalu disimpan kedalam 'List of Mahasiswa' pada kasus ini dapat ditemukan pada variable `'fetchData'`.

- e. Tambahkan Activity baru bernama **EditMhsActivity.java** untuk nanti digunakan pada Langkah selanjutnya.

Lalu coba jalankan maka akan tampil seperti berikut.



Data dari database akan ditampilkan dalam bentuk list menggunakan RecyclerView

4.7 Edit dan hapus data dari database

a. Membuat tampilan untuk **activity_edit_mhs.xml**

Ubahlah code menjadi seperti berikut

```
<?xml version="1.0" encoding="utf-8"?>
<ScrollView 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=".EditMhsActivity"
    android:padding="8dp">

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

        <ImageButton
            android:id="@+id/btn_back"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:src="@drawable/ic_back"
            android:background="@android:color/transparent"
            android:clickable="true"

            android:foreground="?android:attr/selectableItemBackground"/>

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Ubah Data Mahasiswa"
            android:textSize="20sp"
            android:layout_marginBottom="20dp"
            android:textColor="@android:color/black"
            android:layout_gravity="center_horizontal"/>

        <LinearLayout
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_marginBottom="8dp"
            android:layout_gravity="center_horizontal">

            <TextView
                android:layout_width="65dp"
                android:layout_height="wrap_content"
                android:text="NIM"
                android:textColor="@android:color/black"
                android:layout_marginRight="10dp"/>

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:textColor="@android:color/black"
```

```
        android:text=":"/>

        <EditText
            android:id="@+id/et_nim"
            android:layout_width="200dp"
            android:layout_height="wrap_content"
            android:hint="Masukkan NIM"
            android:textSize="12sp"
            android:layout_marginLeft="5dp"/>
    </LinearLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="20dp"
        android:layout_gravity="center_horizontal">

        <TextView
            android:layout_width="65dp"
            android:layout_height="wrap_content"
            android:text="Nama"
            android:textColor="@android:color/black"
            android:layout_marginRight="8dp"/>

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:textColor="@android:color/black"
            android:text=":"/>

        <EditText
            android:id="@+id/et_nama"
            android:layout_width="200dp"
            android:layout_height="wrap_content"
            android:hint="Masukkan Nama"
            android:textSize="12sp"
            android:layout_marginLeft="5dp"/>
    </LinearLayout>

    <LinearLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="20dp"
        android:layout_gravity="center_horizontal">

        <TextView
            android:layout_width="65dp"
            android:layout_height="wrap_content"
            android:text="Jurusan"
            android:textColor="@android:color/black"
            android:layout_marginRight="8dp"/>

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:textColor="@android:color/black"
            android:text=":"/>
    </LinearLayout>
```

```
<EditText
    android:id="@+id/et_jurusan"
    android:layout_width="200dp"
    android:layout_height="wrap_content"
    android:hint="Masukkan Jurusan"
    android:textSize="12sp"
    android:layout_marginLeft="5dp"/>
</LinearLayout>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginBottom="20dp"
    android:layout_gravity="center_horizontal">

    <TextView
        android:layout_width="65dp"
        android:layout_height="wrap_content"
        android:text="Angkatan"
        android:textColor="@android:color/black"
        android:layout_marginRight="8dp"/>

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textColor="@android:color/black"
        android:text=":"/>

    <EditText
        android:id="@+id/et_angkatan"
        android:layout_width="200dp"
        android:layout_height="wrap_content"
        android:hint="Masukkan Angkatan"
        android:textSize="12sp"
        android:layout_marginLeft="5dp"/>
</LinearLayout>

<LinearLayout
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center_horizontal">
    <Button
        android:id="@+id/btn_simpan"
        android:layout_width="150dp"
        android:layout_height="wrap_content"
        android:layout_marginRight="5dp"
        android:backgroundTint="#00b300"
        android:text="Simpan"/>
    <Button
        android:id="@+id/btn_hapus"
        android:layout_width="150dp"
        android:layout_height="wrap_content"
        android:layout_marginLeft="5dp"
        android:backgroundTint="@android:color/holo_red_dark"
        android:text="Hapus"/>
</LinearLayout>
```

```

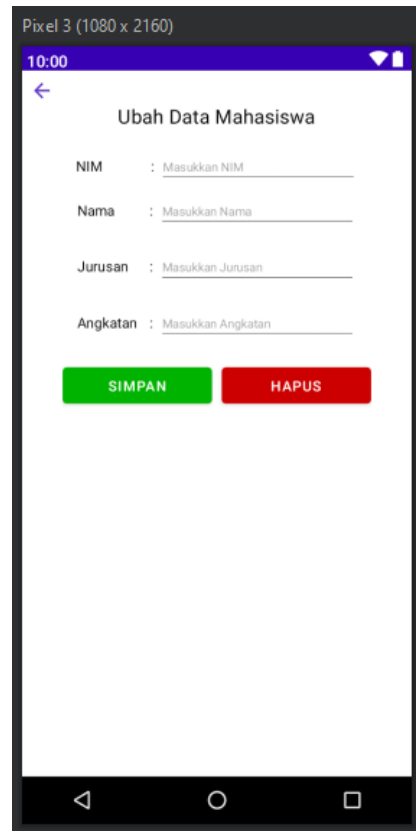
        </LinearLayout>

    </LinearLayout>

</ScrollView>

```

Maka tampilannya akan seperti berikut



- b. Ubah isi dari EditMhsActivity.java

Buka EditMhsActivity.java dan ubahlah code menjadi seperti berikut

```

public class EditMhsActivity extends AppCompatActivity {

    EditText etNIM,etNama,etJurusan,etAngkatan;
    Button btnSimpan,btnHapus;
    ImageButton btnBack;
    String temp_nim,temp_nama,temp_jurusan,temp_angkatan;

    DatabaseReference databaseReference;

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

        //Mengambil data ('nim') yang dikirim dari activity sebelumnya
    }
}

```

```

Intent intent = getIntent();
String nim = intent.getStringExtra("nim");

//inisialisasi variable dengan id yang ada di layout
etNIM = findViewById(R.id.et_nim);
etNama = findViewById(R.id.et_nama);
etJurusan = findViewById(R.id.et_jurusan);
etAngkatan = findViewById(R.id.et_angkatan);

btnSimpan = findViewById(R.id.btn_simpan);
btnHapus = findViewById(R.id.btn_hapus);
btnBack = findViewById(R.id.btn_back);

databaseReference =
FirebaseDatabase.getInstance().getReference().child("mhs").child(nim);
databaseReference.addValueEventListener(new
ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot
dataSnapshot) {
        temp_nim =
dataSnapshot.child("nim").getValue().toString();
        temp_nama =
dataSnapshot.child("nama").getValue().toString();
        temp_jurusan =
dataSnapshot.child("jurusan").getValue().toString();
        temp_angkatan =
dataSnapshot.child("angkatan").getValue().toString();

        etNIM.setText(temp_nim);
        etNama.setText(temp_nama);
        etJurusan.setText(temp_jurusan);
        etAngkatan.setText(temp_angkatan);
    }

    @Override
    public void onCancelled(@NonNull DatabaseError
databaseError) {
    }
});

btnSimpan.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        updateData();
        finish();
    }
});

btnHapus.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        hapusData();
    }
});

```

```

        finish();
    }
});

btnBack.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        finish();
    }
});

}

private void hapusData() {
    databaseReference.removeValue();
}

private void updateData() {
    String nim = etNIM.getText().toString();
    String nama = etNama.getText().toString();
    String jurusan = etJurusan.getText().toString();
    String angkatan = etAngkatan.getText().toString();

    if(!nim.equals(temp_nim) || !nama.equals(temp_nama) ||
    !jurusan.equals(temp_jurusan) || !angkatan.equals(temp_angkatan)){
        Mahasiswa mahasiswa = new
        Mahasiswa(nim,nama,jurusan,angkatan);

        databaseReference.setValue(mahasiswa).addOnSuccessListener(new
        OnSuccessListener<Void>() {
            @Override
            public void onSuccess(Void aVoid) {
                Toast.makeText(EditMhsActivity.this, "Data
                berhasil diubah", Toast.LENGTH_SHORT).show();

                etNIM.setText("");
                etNama.setText("");
                etJurusan.setText("");
                etAngkatan.setText("");

            }
        }).addOnFailureListener(new OnFailureListener() {
            @Override
            public void onFailure(@NonNull Exception e) {
                Toast.makeText(EditMhsActivity.this, "Data gagal
                diubah", Toast.LENGTH_SHORT).show();
            }
        });
    } else{
        Toast.makeText(this, "Data Tidak berubah",
        Toast.LENGTH_SHORT).show();
    }
}
}
}

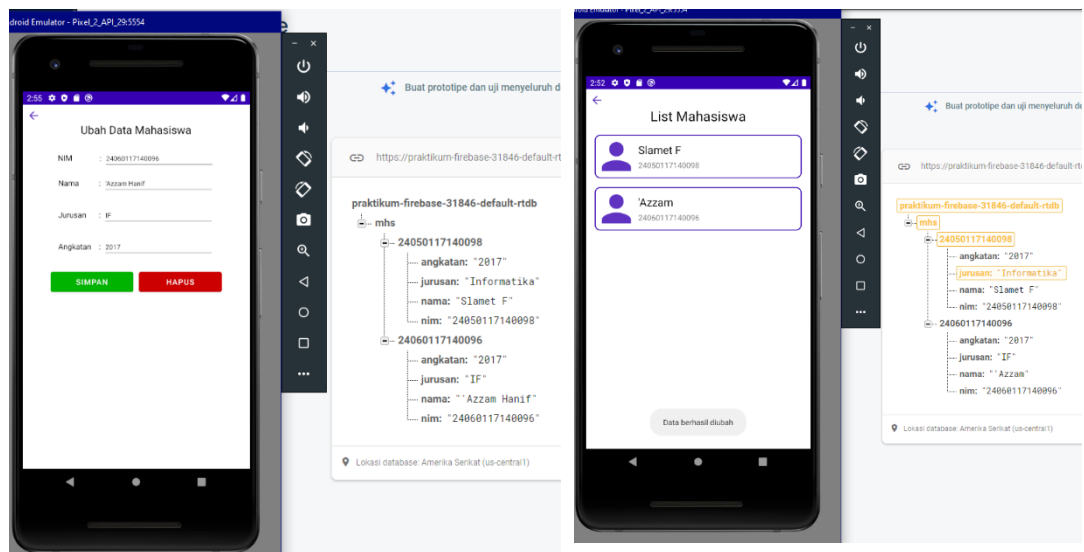
```

Langkah awal yang dilakukan adalah mengambil stringExtra yang telah dikirimkan

dari activity sebelumnya lalu disimpan kedalam variable 'nim'. Lalu menginisialisasi semua object yang dibutuhkan.

Kemudian membuat method 'updateData()', method ini digunakan untuk mengupdate data dari database. Selanjutnya dilakukan validasi apakah ada perubahan yang terjadi pada editText dengan data yang ada di database, jika terjadi perbedaan maka method ini akan melakukan update data, jika tidak maka akan memunculkan pesan 'data tidak berubah'.

Kemudian membuat method 'hapusData()', method ini digunakan untuk menghapus data dari database secara realtime. Kedua method tersebut di trigger oleh button yang sudah kita buat, yakni button simpan dan button hapus. Coba jalankan aplikasi, jika berhasil maka akan tampil seperti berikut



5. Tugas

1. Membuat Aplikasi sesuai dengan yang ada di modul
2. Membuat Video yang berisi tentang Penjelasan kode dan praktikum minggu ini