

## Modul 9 Insert, Update and Delete

### 1.1. ApiClient.kt

Pada pertemuan kemarin, untuk url api masih perlu perbaikan dikarenakan akan terlalu panjang dibagian **ApiService.kt** endpoint. Ubah menjadi seperti ini :

```
package com.app.latihanrestapi.network

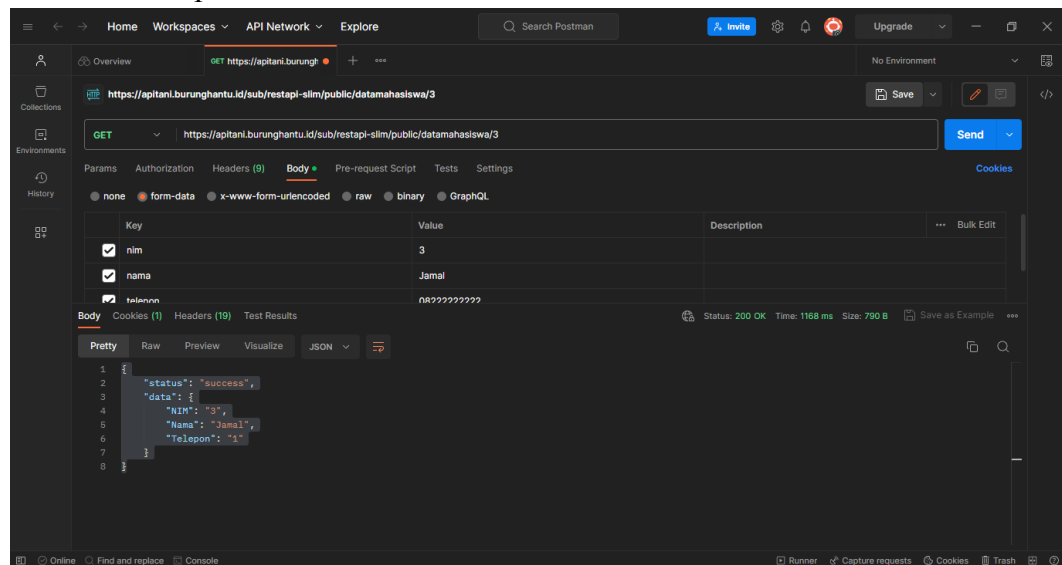
import retrofit2.Retrofit
import retrofit2.converter.gson.GsonConverterFactory

object ApiClient {
    const val BASE_URL = "https://apitani.burunghantu.id/sub/restapi-slim/public/"

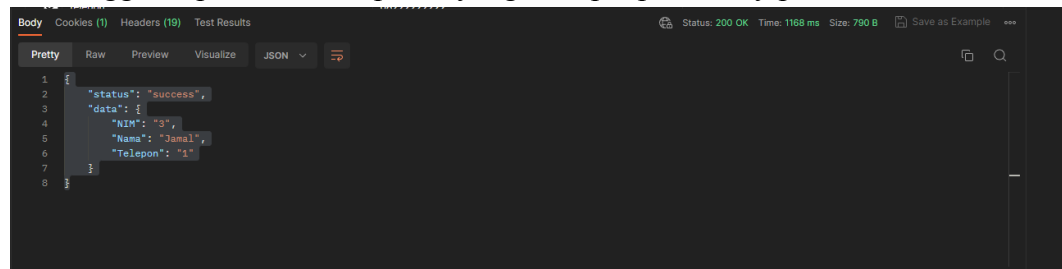
    val instance: ApiService by lazy {
        val retrofit = Retrofit.Builder()
            .baseUrl(BASE_URL)
            .addConverterFactory(GsonConverterFactory.create())
            .build()
        retrofit.create(ApiService :: class.java)
    }
}
```

### 1.2. Model Request dan Response

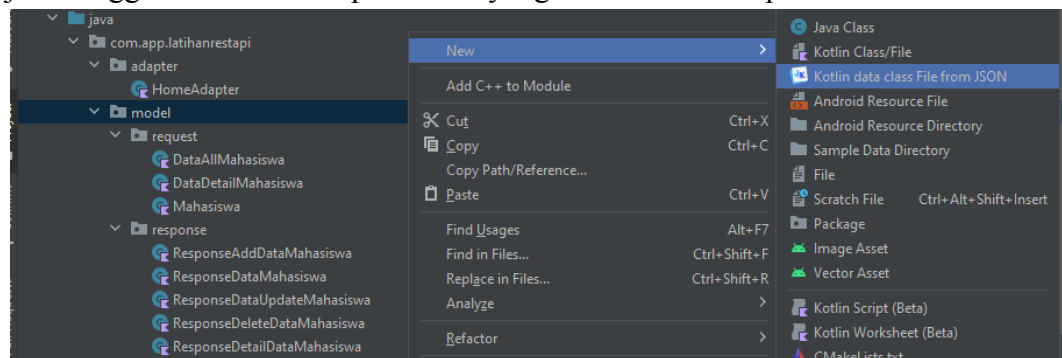
Pada folder ini kita harus menambahkan file response dan request tambahan yang akan digunakan untuk insert, update dan delete. Kita bisa menggunakan tools bernama postman.



Kita tinggal copas semua response yang terdapat pada body postman.



Lalu tambahkan file baru menggunakan plugin kotlin data class file from json hingga semua data response dan yang dibutuhkan didapatkan.



### Mahasiswa.kt

```
package com.app.latihanrestapi.model.request

data class Mahasiswa(val nim : String, val nama : String, val
telepon : String)
```

### ResponseAddDataMahasiswa.kt

```
package com.app.latihanrestapi.model.response

import com.google.gson.annotations.SerializedName

data class ResponseAddDataMahasiswa(
    @SerializedName("data")
    val data: String,
    @SerializedName("status")
    val status: String
)
```

### ResponseDataUpdateMahasiswa.kt

```
package com.app.latihanrestapi.model.response

import com.google.gson.annotations.SerializedName

data class ResponseDataUpdateMahasiswa(
    @SerializedName("data")
    val data: String,
```

```

        @SerializedName("status")
        val status: String
    )
}

```

### ResponseDeleteDataMahasiswa.kt

```

package com.app.latihanrestapi.model.response

import com.google.gson.annotations.SerializedName

data class ResponseDeleteDataMahasiswa(
    @SerializedName("data")
    val data: String,
    @SerializedName("status")
    val status: String
)

```

### 1.3. ApiService.kt

Pada **ApiService.kt** kita tambahkan method dan endpoint yang digunakan untuk membuat fungsi insert, update dan delete. Kode seperti ini :

```

package com.app.latihanrestapi.network

import com.app.latihanrestapi.model.request.Mahasiswa
import com.app.latihanrestapi.model.response.ResponseAddDataMahasiswa
import com.app.latihanrestapi.model.response.ResponseDataMahasiswa
import com.app.latihanrestapi.model.response.ResponseDataUpdateMahasiswa
import com.app.latihanrestapi.model.response.ResponseDeleteDataMahasiswa
import com.app.latihanrestapi.model.response.ResponseDetailDataMahasiswa
import retrofit2.Call
import retrofit2.http.*

interface ApiService {
    @GET("datamahasiswa/")
    fun getDataMahasiswa() : Call<ResponseDataMahasiswa>

    @GET("datamahasiswa/{nim}")
    fun getDetailMahasiswa(@Path("nim") nim : String) :
    Call<ResponseDetailDataMahasiswa>

    @POST("datamahasiswa/")
    fun addDataMahasiswa(@Body data: Mahasiswa) : Call<ResponseAddDataMahasiswa>

    @DELETE("datamahasiswa/{nim}")
    fun deleteDataMahasiswa(@Path("nim") nim : String) :
    Call<ResponseDeleteDataMahasiswa>

    @POST("datamahasiswa/{nim}")
    fun updateDataMahasiswa(@Path("nim") nim : String, @Body data: Mahasiswa) :
    Call<ResponseDataUpdateMahasiswa>
}

```

Berikut ini merupakan pengertian dari fungsi-fungsi yang ada pada retrofit yang sedang kita gunakan :

1. **GET**: Metode GET adalah salah satu metode HTTP yang digunakan untuk mengambil data dari server. Permintaan GET digunakan ketika kita ingin mendapatkan data dari server tanpa mengubah atau memodifikasi data tersebut di server.

2. **POST**: Metode POST adalah metode HTTP yang digunakan untuk mengirimkan data baru ke server. Permintaan POST digunakan ketika kita ingin membuat entitas baru di server, seperti mengirimkan formulir atau mengirimkan data dari aplikasi ke server.

3. **PUT**: Metode PUT adalah metode HTTP yang digunakan untuk mengirimkan data yang menggantikan entitas yang ada di server. Permintaan PUT digunakan ketika kita ingin memperbarui atau menggantikan data yang sudah ada di server.

4. **DELETE**: Metode DELETE adalah metode HTTP yang digunakan untuk menghapus entitas yang ada di server. Permintaan DELETE digunakan ketika kita ingin menghapus data atau entitas dari server.

5. **PATH**: Anotasi **@Path** digunakan dalam Retrofit untuk menyertakan nilai dinamis dalam URL permintaan. Nilai dinamis ini dapat digunakan untuk mengganti bagian spesifik dari URL dengan nilai yang diambil dari variabel atau parameter dalam kode aplikasi.

6. **BODY**: Anotasi **@Body** digunakan dalam Retrofit untuk mengirimkan data dalam permintaan HTTP menggunakan metode POST atau PUT. Anotasi ini memungkinkan kita untuk mengirim objek serializable sebagai tubuh (body) permintaan.

#### 1.4. Tambahkan 2 view

Untuk membuat Insert dan Update, kita perlu membuat 2 buah view baru yaitu **TambahDataFragment** dan **UpdateFragment**.

##### TambahDataFragment.kt

```
package com.app.latihanrestapi.view.tambah

import android.os.Bundle
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.Toast
import androidx.fragment.app.Fragment
import androidx.lifecycle.ViewModelProvider
import androidx.navigation.fragment.findNavController
import com.app.latihanrestapi.R
```

```

import
com.app.latihanrestapi.databinding.FragmentTambahDataBinding
import com.app.latihanrestapi.viewmodel.ViewModelMahasiswa

class TambahDataFragment : Fragment() {
    lateinit var binding : FragmentTambahDataBinding
    lateinit var viewModel : ViewModelMahasiswa
    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View? {
        // Inflate the layout for this fragment
        binding =
FragmentTambahDataBinding.inflate(layoutInflater,container,false)
        return binding.root
    }

    override fun onViewCreated(view: View, savedInstanceState:
Bundle?) {
        super.onViewCreated(view, savedInstanceState)

        viewModel =
ViewModelProvider(requireActivity()).get(ViewModelMahasiswa::c
lass.java)

        binding.btnAdd.setOnClickListener{
            val nim = binding.inputNim.text.toString()
            val nama = binding.inputNama.text.toString()
            val telepon = binding.inputTelepon.text.toString()

viewModel.insertMahasiswa().observe(viewLifecycleOwner) {
                if (it != null){

findNavController().navigate(R.id.action_tambahDataFragment_to
_homeFragment)

                    Toast.makeText(context,"Data berhasil
ditambahkan",Toast.LENGTH_SHORT).show()
                }else{
                    Toast.makeText(context,"Data gagal
ditambahkan",Toast.LENGTH_SHORT).show()
                }
            }
            viewModel.insertDataMahasiswa(nim,nama,telepon)
        }

        binding.btnBack.setOnClickListener{
            findNavController().navigateUp()
        }
    }
}

```

### fragment\_tambah\_data.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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=".view.tambah.TambahDataFragment">

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/appBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="44dp"
        android:background="@color/white"
        app:layout_constraintBottom_toTopOf="@+id/textView"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent">
        <androidx.appcompat.widget.Toolbar
            android:layout_width="match_parent"
            android:layout_height="wrap_content">

            <ImageButton
                android:id="@+id/btnBack"
                android:backgroundTint="@color/white"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:src="@drawable/baseline_arrow_back_24"
                />

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginStart="30dp"
                android:text="Tambah Data Siswa"
                android:textColor="@color/black"
                android:textSize="20sp"
                android:textStyle="bold"
                />

        </androidx.appcompat.widget.Toolbar>
    </com.google.android.material.appbar.AppBarLayout>

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="100dp"
        android:text="Tambah Data Siswa"
        android:textAlignment="center"
        android:textColor="@color/black"
        android:textSize="24sp"
        android:textStyle="bold"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
```

```

        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

<com.google.android.material.textfield.TextInputLayout
    android:id="@+id/editTextNim"

style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="40dp"
    android:paddingStart="10dp"
    android:paddingEnd="10dp"
    app:layout_constraintTop_toBottomOf="@+id/textView">

<com.google.android.material.textfield.TextInputEditText
    android:id="@+id/inputNim"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:drawablePadding="10dp"
    android:hint="Nim"
    android:inputType="phone">

</com.google.android.material.textfield.TextInputEditText>
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:id="@+id/editTextNama"

style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_marginTop="10dp"
    android:paddingStart="10dp"
    android:paddingEnd="10dp"
    app:layout_constraintTop_toBottomOf="@+id/editTextNim">

<com.google.android.material.textfield.TextInputEditText
    android:id="@+id/inputNama"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:drawablePadding="10dp"
    android:hint="Nama"
    android:inputType="text">

</com.google.android.material.textfield.TextInputEditText>
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.textfield.TextInputLayout
    android:id="@+id/editTextTelepon"

```

```

style="@style/Widget.MaterialComponents.TextInputLayout.OutlinedBox"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:paddingStart="10dp"
        android:paddingEnd="10dp"

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

<com.google.android.material.textfield.TextInputEditText
        android:id="@+id/inputTelepon"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:drawablePadding="10dp"
        android:hint="Telepon"
        android:inputType="phone">

</com.google.android.material.textfield.TextInputEditText>
</com.google.android.material.textfield.TextInputLayout>

<com.google.android.material.button.MaterialButton
        android:layout_marginStart="10dp"
        android:layout_marginEnd="10dp"
        android:id="@+id/btnTambah"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="40dp"
        android:backgroundTint="#0095F6"
        android:padding="12dp"
        android:text="Tambah"

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

</androidx.constraintlayout.widget.ConstraintLayout>

```



← Tambah Data Siswa

**Tambah Data Siswa**

Nim

Nama

Telepon

TAMBAH

### UpdateFragment.kt

```
package com.app.latihanrestapi.view.update

import android.os.Bundle
import androidx.fragment.app.Fragment
import android.view.LayoutInflater
import android.view.View
import android.view.ViewGroup
import android.widget.Toast
import androidx.lifecycle.ViewModelProvider
import androidx.navigation.fragment.findNavController
import com.app.latihanrestapi.R
import com.app.latihanrestapi.databinding.FragmentUpdateBinding
import com.app.latihanrestapi.viewmodel.ViewModelMahasiswa

class UpdateFragment : Fragment() {
    lateinit var binding : FragmentUpdateBinding
    lateinit var viewModel : ViewModelMahasiswa

    override fun onCreateView(
        inflater: LayoutInflater, container: ViewGroup?,
        savedInstanceState: Bundle?
    ): View? {
        // Inflate the layout for this fragment
        binding =
            FragmentUpdateBinding.inflate(inflater, container, false)
        return binding.root
    }
}
```

```

        override fun onCreateView(view: View, savedInstanceState: Bundle?) {
            super.onCreateView(view, savedInstanceState)
            val nim = arguments?.getString("nim")
            val nama = arguments?.getString("nama")
            val telepon = arguments?.getString("telepon")

            viewModel =
                ViewModelProvider(this).get(ViewModelMahasiswa::class.java)

            binding.inputNim.setText(nim)
            binding.inputNama.setText(nama)
            binding.inputTelepon.setText(telepon)

            binding.btnUpdate.setOnClickListener{
                val updatedNim = binding.inputNim.text.toString()
                val updatedNama = binding.inputNama.text.toString()
                val updatedTelepon =
                    binding.inputTelepon.text.toString()

                viewModel.updateMahasiswa().observe(viewLifecycleOwner) {
                    if (it != null) {
                        Toast.makeText(context, "Data berhasil
                        diupdate", Toast.LENGTH_SHORT).show()

                        findNavController().navigate(R.id.action_updateFragment_to_homeFragment)
                    } else {
                        Toast.makeText(context, "Data gagal
                        diupdate", Toast.LENGTH_SHORT).show()
                    }
                }

                viewModel.updateDataMahasiswa(updatedNim, updatedNama, updatedTelepon)
            }

            binding.btnBack.setOnClickListener{
                findNavController().navigateUp()
            }
        }
    }
}

```

### fragment\_update.xml

```

<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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=".view.update.UpdateFragment">

    <com.google.android.material.appbar.AppBarLayout

```

```

        android:id="@+id/appBarLayout"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginBottom="44dp"
        android:background="@color/white"
        app:layout_constraintBottom_toTopOf="@+id/textView"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent">

        <androidx.appcompat.widget.Toolbar
            android:layout_width="match_parent"
            android:layout_height="wrap_content">

            <ImageButton
                android:id="@+id/btnBack"
                android:backgroundTint="@color/white"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:src="@drawable/baseline_arrow_back_24"

                />

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_marginStart="30dp"
                android:text="Update Data Siswa"
                android:textColor="@color/black"
                android:textSize="20sp"
                android:textStyle="bold"

                />

        </androidx.appcompat.widget.Toolbar>

    </com.google.android.material.appbar.AppBarLayout>

    <TextView
        android:id="@+id/textView"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="100dp"
        android:text="Update Data Siswa"
        android:textAlignment="center"
        android:textColor="@color/black"
        android:textSize="24sp"
        android:textStyle="bold"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

```

```

        <com.google.android.material.textfield.TextInputLayout
            android:id="@+id/editTextNim"

style="@style/Widget.MaterialComponents.TextInputLayout.Outlined
Box"

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="40dp"
        android:paddingStart="10dp"
        android:paddingEnd="10dp"
        app:layout_constraintTop_toBottomOf="@+id/textView">

        <com.google.android.material.textfield.TextInputEditText
            android:id="@+id/inputNim"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:drawablePadding="10dp"
            android:hint="Nim"
            android:inputType="none"
            android:enabled="false"
            >

</com.google.android.material.textfield.TextInputEditText>
</com.google.android.material.textfield.TextInputLayout>

        <com.google.android.material.textfield.TextInputLayout
            android:id="@+id/editTextNama"

style="@style/Widget.MaterialComponents.TextInputLayout.Outlined
Box"

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:paddingStart="10dp"
        android:paddingEnd="10dp"
        app:layout_constraintTop_toBottomOf="@+id/editTextNim">

        <com.google.android.material.textfield.TextInputEditText
            android:id="@+id/inputNama"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:drawablePadding="10dp"
            android:hint="Nama"
            android:inputType="text">

</com.google.android.material.textfield.TextInputEditText>
</com.google.android.material.textfield.TextInputLayout>

        <com.google.android.material.textfield.TextInputLayout
            android:id="@+id/editTextTelepon"

style="@style/Widget.MaterialComponents.TextInputLayout.Outlined
Box"

        android:layout_width="match_parent"

```

```

        android:layout_height="wrap_content"
        android:layout_marginTop="10dp"
        android:paddingStart="10dp"
        android:paddingEnd="10dp"
        app:layout_constraintTop_toBottomOf="@+id/editTextNama">

        <com.google.android.material.textfield.TextInputEditText
            android:id="@+id/inputTelepon"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:drawablePadding="10dp"
            android:hint="Telepon"
            android:inputType="phone">

    </com.google.android.material.textfield.TextInputEditText>
    </com.google.android.material.textfield.TextInputLayout>

    <com.google.android.material.button.MaterialButton
        android:layout_marginStart="10dp"
        android:layout_marginEnd="10dp"
        android:id="@+id/btnUpdate"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="40dp"
        android:backgroundTint="#0095F6"
        android:padding="12dp"
        android:text="Update"

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

</androidx.constraintlayout.widget.ConstraintLayout>

```

The screenshot shows a mobile application interface for updating student data. At the top, there is a back arrow and the title 'Update Data Siswa'. Below the title, the same text 'Update Data Siswa' is centered. There are three text input fields stacked vertically, labeled 'Nim', 'Nama', and 'Telepon'. At the bottom of the form is a blue button with the text 'UPDATE' in white capital letters.

### 1.5. HomeFragment.kt

Selanjutnya pada **HomeFragment.kt** kita melakukan sedikit perubahan yaitu menambahkan button untuk tambah data

```
binding.btnAdd.setOnClickListener{
    findNavController().navigate(R.id.action_homeFragment_to_tambahDataFragment)
}
```

Tidak hanya itu kita juga melakukan perubahan dengan menambahkan toolbar, berikut adalah file xml layout Home.

#### fragment\_home.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.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=".view.home.HomeFragment">

    <com.google.android.material.appbar.AppBarLayout
        android:id="@+id/appBarLayout2"
        android:layout_width="match_parent"
```

```

        android:layout_height="wrap_content"
        android:background="@color/white"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent">

        <androidx.appcompat.widget.Toolbar
            android:layout_width="match_parent"
            android:layout_height="wrap_content">

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="@string/app_name"
                android:textColor="@color/black"
                android:textSize="20sp"
                android:textStyle="bold"
            />

            <ImageButton
                android:id="@+id/btnAdd"
                android:layout_gravity="end"
                android:layout_marginEnd="10dp"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:src="@drawable/baseline_add_24"
                android:backgroundTint="@color/white"
            />

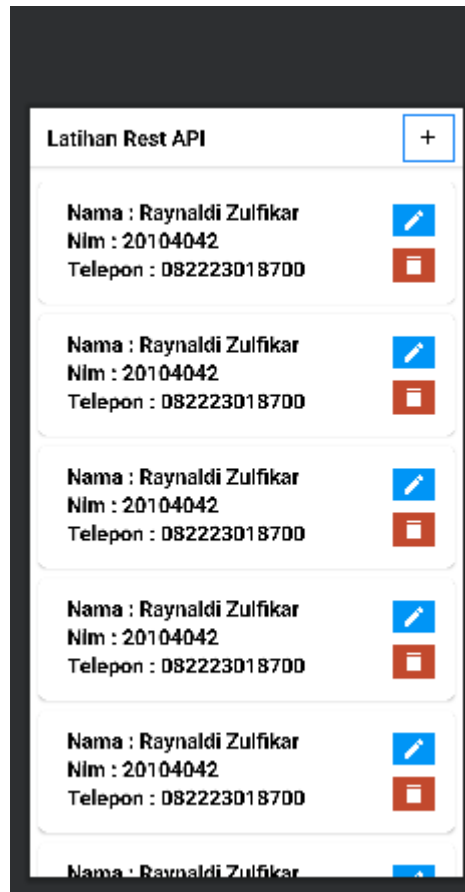
        </androidx.appcompat.widget.Toolbar>

    </com.google.android.material.appbar.AppBarLayout>

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/rvUser"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_marginTop="70dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/appBarLayout2"
        tools:listitem="@layout/user_list">

        </androidx.recyclerview.widget.RecyclerView>
    </androidx.constraintlayout.widget.ConstraintLayout>

```



### 1.6. user\_list.xml

Tambahkan 2 imagebutton pada **user\_list.xml** yang nantinya berguna untuk update dan hapus data. Berikut contoh codenya :

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.cardview.widget.CardView
    android:id="@+id/cardView"
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:elevation="6dp"
    app:cardCornerRadius="8dp"
    android:layout_height="wrap_content"
    android:layout_marginStart="8dp"
    android:layout_marginEnd="8dp"
    android:layout_marginBottom="10dp"
    xmlns:app="http://schemas.android.com/apk/res-auto">

    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_margin="15dp"
        android:layout_height="wrap_content">

        <ImageButton
            android:id="@+id/btnEdit"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_alignParentEnd="true"
            android:src="@drawable/baseline_edit_24"
```



```

        android:backgroundTint="#0095F6"
    />

    <ImageButton
        android:id="@+id/btnDellete"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_below="@id/btnEdit"
        android:src="@drawable/baseline_delete_24"
        android:backgroundTint="#C2492D"

    />

    <TextView
        android:id="@+id/txtNama"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="20sp"
        android:textStyle="bold"
        android:textColor="@color/black"
        android:layout_marginStart="12dp"
        android:text="Nama : Raynaldi Zulfikar"
    />

    <TextView
        android:id="@+id/txtNim"
        android:layout_below="@id/txtNama"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="20sp"
        android:textStyle="bold"
        android:textColor="@color/black"
        android:layout_marginStart="12dp"
        android:text="Nim : 20104042"
    />

    <TextView
        android:id="@+id/txtTelepon"
        android:layout_below="@id/txtNim"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:textSize="20sp"
        android:textStyle="bold"
        android:textColor="@color/black"
        android:layout_marginStart="12dp"
        android:text="Telepon : 082223018700"
    />

</RelativeLayout>

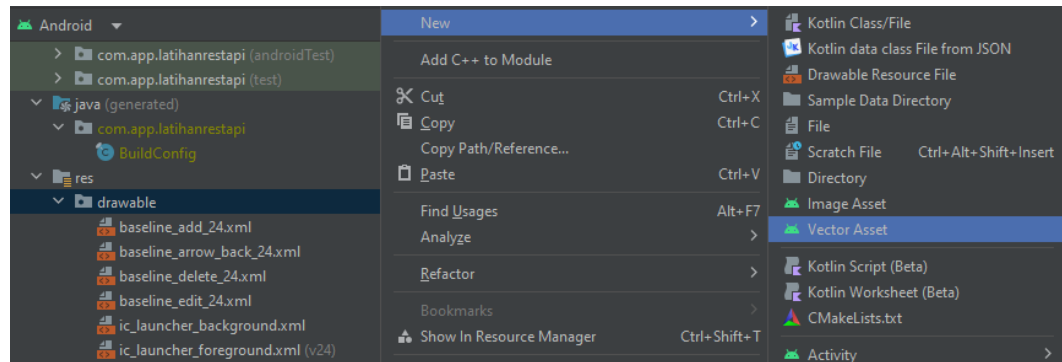
</androidx.cardview.widget.CardView>

```

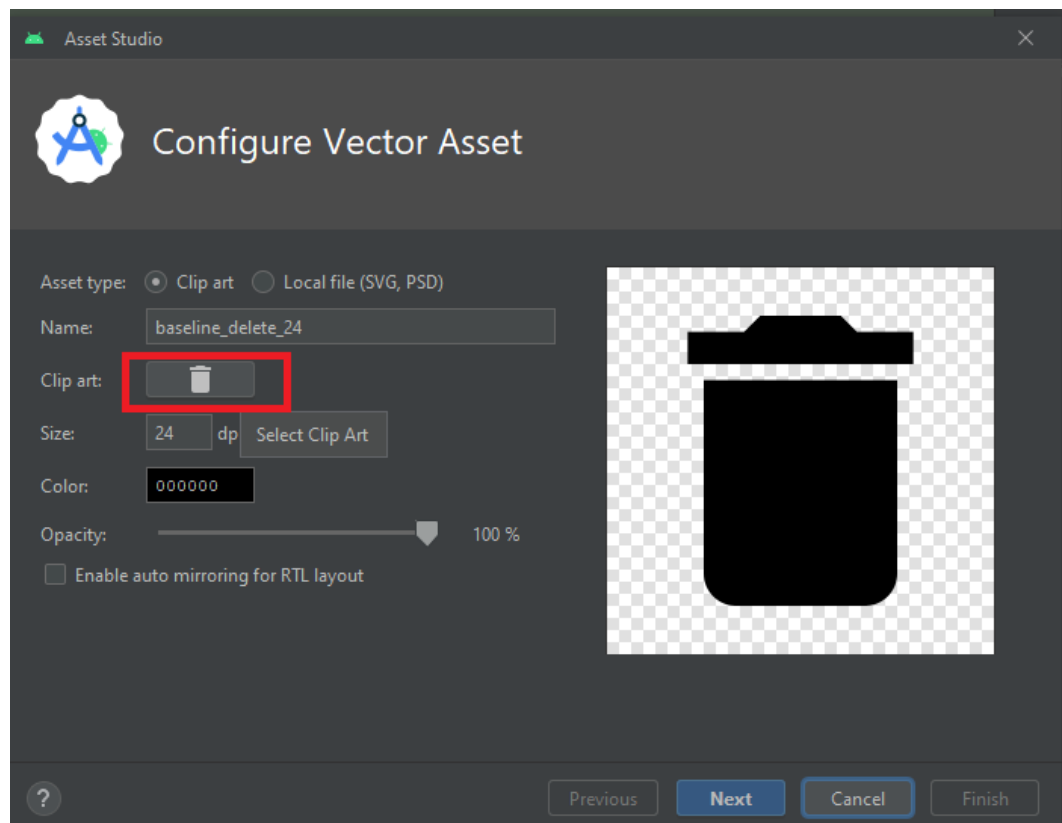
### 1.7. Icon atau drawable vector

Kita membutuhkan aset drawable vector yang berguna untuk button diantaranya, add, update atau edit, dan delete. Untuk menambahkannya kita

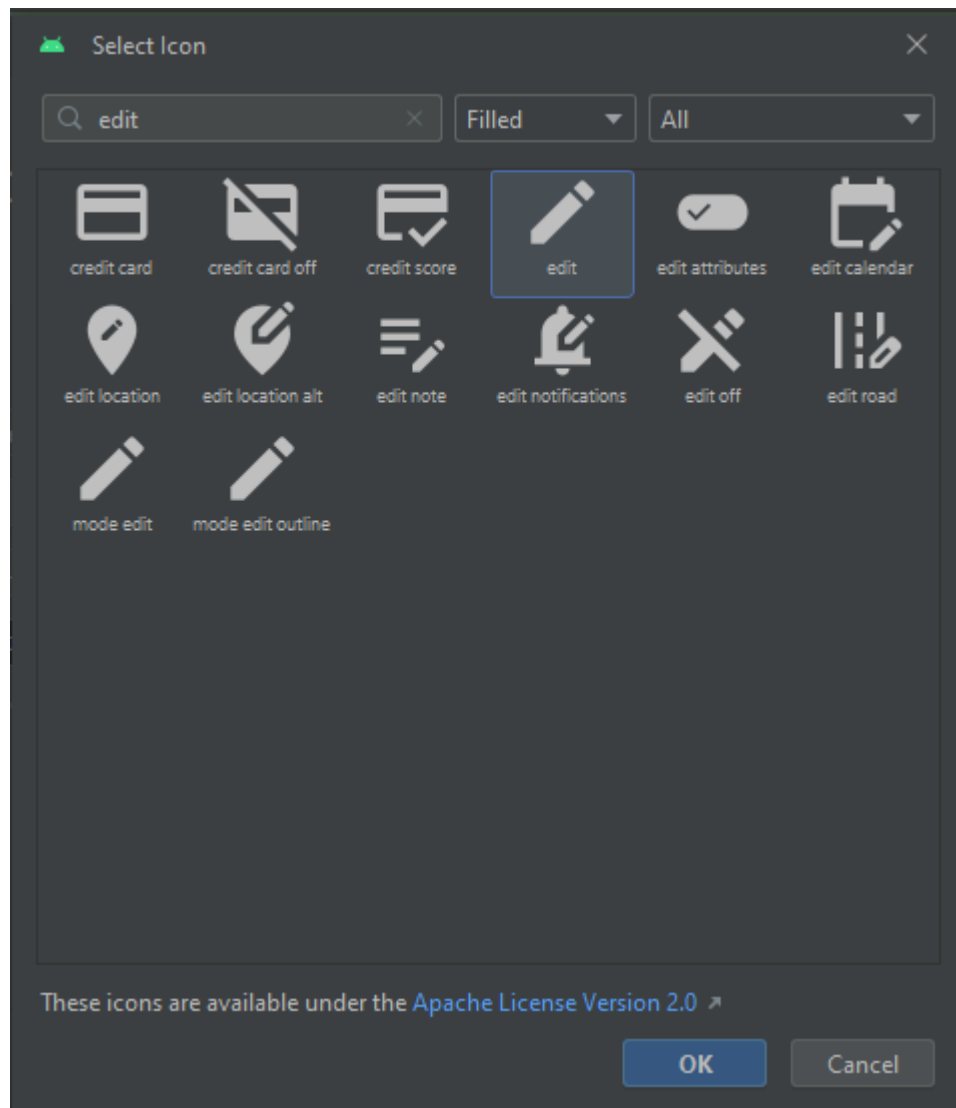
bisa klik kanan dibagian drawable dan memilih new vector. Seperti gambar dibawah ini :



Jika sudah klik bagian clip art ( yang dilingkari )



Jika sudah maka tekan ok, lalu next dan finish



### 1.8. HomeAdapter.kt

Pada **HomeAdapter.kt** kita tambahkan variabel baru yaitu api, yang berguna untuk mengambil fungsi instance dari api service delete data. Tidak hanya hapus data, kita juga menambahkan fungsi untuk edit data yang nantinya jika tombol di klik maka akan berpindah ke layout UdateFragment.kt. Berikut contoh codenya :

```
package com.app.latihanrestapi.adapter

import android.annotation.SuppressLint
import android.os.Bundle
import android.view.LayoutInflater
import android.view.ViewGroup
import android.widget.Toast
import androidx.navigation.Navigation
import androidx.recyclerview.widget.RecyclerView
import com.app.latihanrestapi.R
import com.app.latihanrestapi.databinding.UserListBinding
import com.app.latihanrestapi.model.request.DataAllMahasiswa
```

```

import
com.app.latihanrestapi.model.response.ResponseDeleteDataMahasiswa
a
import com.app.latihanrestapi.network.ApiClient
import retrofit2.Call
import retrofit2.Callback
import retrofit2.Response

class HomeAdapter(private var dataMhs : List<DataAllMahasiswa>)
: RecyclerView.Adapter<HomeAdapter.ViewHolder>() {

    class ViewHolder(val binding : UserListBinding) :
RecyclerView.ViewHolder(binding.root) {
        var api = ApiClient.instance
    }

    override fun onCreateViewHolder(parent: ViewGroup, viewType:
Int): ViewHolder {
        val view =
UserListBinding.inflate(LayoutInflater.from(parent.context),
parent, false)
        return ViewHolder(view)
    }

    override fun getItemCount(): Int {
        return dataMhs.size
    }

    @SuppressWarnings("RecyclerView")
    override fun onBindViewHolder(holder: ViewHolder, position:
Int) {
        holder.binding.txtNama.text = dataMhs[position].nama
        holder.binding.txtNim.text = dataMhs[position].nim
        holder.binding.txtTelepon.text =
dataMhs[position].telepon
        holder.binding.cardView.setOnClickListener{
            val bundle = Bundle()
            bundle.putString("nim",dataMhs[position].nim)
            bundle.putString("nama",dataMhs[position].nama)
            bundle.putString("telepon",dataMhs[position].telepon)
            Navigation.findNavController(it).navigate(R.id.action_homeFragm
ent_to_detailFragment, bundle)
        }

        holder.binding.btnDelete.setOnClickListener{
            holder.api.deleteDataMahasiswa(dataMhs[position].nim).enqueue(ob
ject : Callback<ResponseDeleteDataMahasiswa>{
                override fun onResponse(
                    call: Call<ResponseDeleteDataMahasiswa>,
                    response:
Response<ResponseDeleteDataMahasiswa>
                ) {
                    if (response.isSuccessful){
                        dataMhs = dataMhs.toMutableList().apply
{ removeAt(position) }

```

```

        notifyDataSetChanged()
        Toast.makeText(holder.itemView.context,
            "Data berhasil dihapus!", Toast.LENGTH_SHORT).show()
    }
}

    override fun onFailure(call:
        Call<ResponseDeleteDataMahasiswa>, t: Throwable) {
        Toast.makeText(holder.itemView.context,
            "Data gagal dihapus!", Toast.LENGTH_SHORT).show()
        }

    })
}

holder.binding.btnEdit.setOnClickListener{
    val bundle = Bundle()
    bundle.putString("nim", dataMhs[position].nim)
    bundle.putString("nama", dataMhs[position].nama)
    bundle.putString("telepon", dataMhs[position].telepon)

    Navigation.findNavController(it).navigate(R.id.action_homeFragment_to_updateFragment, bundle)
}
}
}

```

### 1.9. ViewModelMahasiswa.kt

Pada viewmodel mahasiswa kita tambahkan variabel dan fungsi untuk insert, update, dan delete. Berikut contoh codenya :

```

package com.app.latihanrestapi.viewmodel

import androidx.lifecycle.MutableLiveData
import androidx.lifecycle.ViewModel
import com.app.latihanrestapi.model.request.DataAllMahasiswa
import com.app.latihanrestapi.model.request.Mahasiswa
import com.app.latihanrestapi.model.response.ResponseAddDataMahasiswa
import com.app.latihanrestapi.model.response.ResponseDataMahasiswa
import com.app.latihanrestapi.model.response.ResponseDataUpdateMahasiswa
import com.app.latihanrestapi.model.response.ResponseDetailDataMahasiswa
import com.app.latihanrestapi.network.ApiClient
import retrofit2.Call
import retrofit2.Callback
import retrofit2.Response

class ViewModelMahasiswa : ViewModel() {
    private val getDataMahasiswa =
        MutableLiveData<List<DataAllMahasiswa>>()
}

```

```

        private val detailMahasiswa =
MutableLiveData<ResponseDetailDataMahasiswa?>()
        private val insertMahasiswa =
MutableLiveData<ResponseAddDataMahasiswa?>()
        private val updateMahasiswa =
MutableLiveData<ResponseDataUpdateMahasiswa?>()

        fun getDataMahasiswa() :
MutableLiveData<List<DataAllMahasiswa?>>{
            return getDataMahasiswa
        }
        fun getDetailMahasiswa():
MutableLiveData<ResponseDetailDataMahasiswa?> {
            return detailMahasiswa
        }
        fun insertMahasiswa():
MutableLiveData<ResponseAddDataMahasiswa?> {
            return insertMahasiswa
        }
        fun updateMahasiswa():
MutableLiveData<ResponseDataUpdateMahasiswa?> {
            return updateMahasiswa
        }

        fun showDataMahasiswa() {
            ApiClient.instance.getDataMahasiswa().enqueue(object :
Callback<ResponseDataMahasiswa>{
                override fun onResponse(
                    call: Call<ResponseDataMahasiswa>,
                    response: Response<ResponseDataMahasiswa>)
                {
                    if (response.isSuccessful){

getDataMahasiswa.postValue(response.body()?.data)
                    }else{
                        getDataMahasiswa.postValue(null)
                    }
                }
                override fun onFailure(call:
Call<ResponseDataMahasiswa>,t: Throwable){
                    getDataMahasiswa.postValue(null)
                }
            })
        }

        fun getDetailData(nim: String) {
            ApiClient.instance.getDetailMahasiswa(nim).enqueue(object
: Callback<ResponseDetailDataMahasiswa> {
                override fun onResponse(
                    call: Call<ResponseDetailDataMahasiswa>,
                    response: Response<ResponseDetailDataMahasiswa>
                ) {
                    if (response.isSuccessful) {
                        detailMahasiswa.postValue(response.body())
                    } else {
                        detailMahasiswa.postValue(null)
                    }
                }
            })
        }

```

```

        }

        override fun onFailure(call:
Call<ResponseDetailDataMahasiswa>, t: Throwable) {
            detailMahasiswa.postValue(null)
        }
    })
}

fun insertDataMahasiswa(nim : String, nama :String,telepon :
String){
ApiClient.instance.addDataMahasiswa (Mahasiswa (nim,nama,telepon))
.enqueue(object : Callback<ResponseAddDataMahasiswa>{
    override fun onResponse(
        call: Call<ResponseAddDataMahasiswa>,
        response: Response<ResponseAddDataMahasiswa>
    ) {
        if (response.isSuccessful){
            insertMahasiswa.postValue(response.body())
        }else{
            insertMahasiswa.postValue(null)
        }
    }

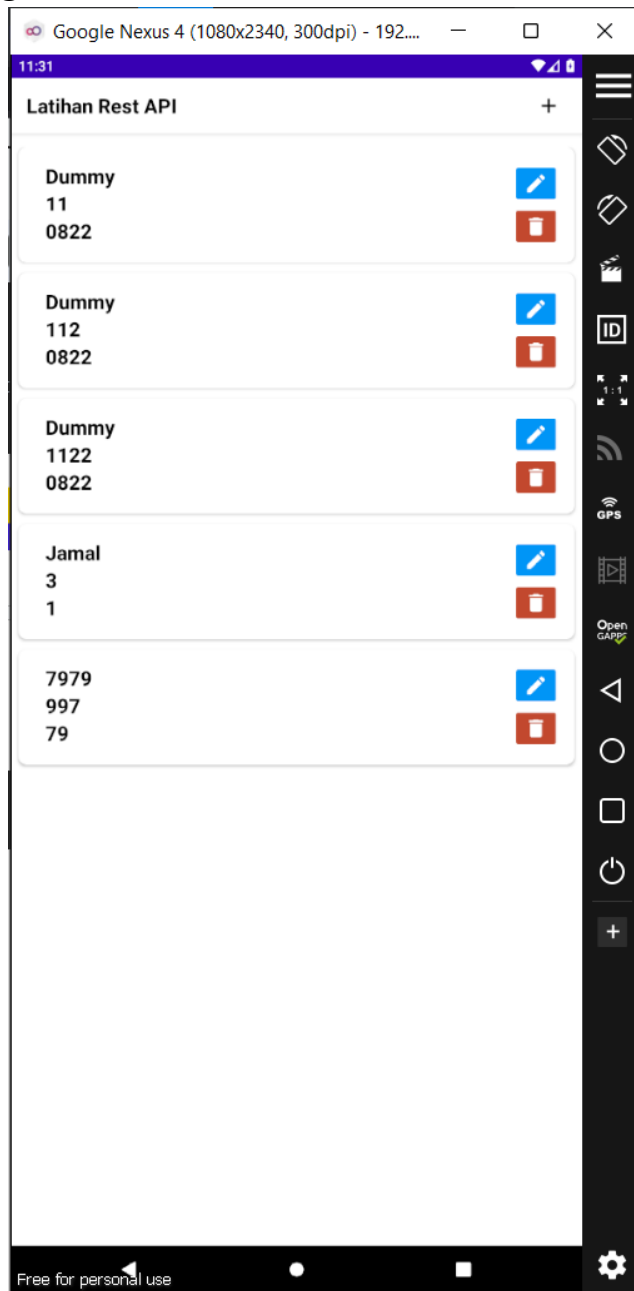
    override fun onFailure(call:
Call<ResponseAddDataMahasiswa>, t: Throwable) {
        insertMahasiswa.postValue(null)
    }
    })
}

fun updateDataMahasiswa(nim : String, nama :String,telepon :
String){
ApiClient.instance.updateDataMahasiswa (nim,Mahasiswa (nim,nama,te
lepon)).enqueue(object : Callback<ResponseDataUpdateMahasiswa>{
    override fun onResponse(
        call: Call<ResponseDataUpdateMahasiswa>,
        response: Response<ResponseDataUpdateMahasiswa>
    ) {
        if (response.isSuccessful){
            updateMahasiswa.postValue(response.body())
        }else{
            updateMahasiswa.postValue(null)
        }
    }

    override fun onFailure(call:
Call<ResponseDataUpdateMahasiswa>, t: Throwable) {
        updateMahasiswa.postValue(null)
    }
    })
}
}

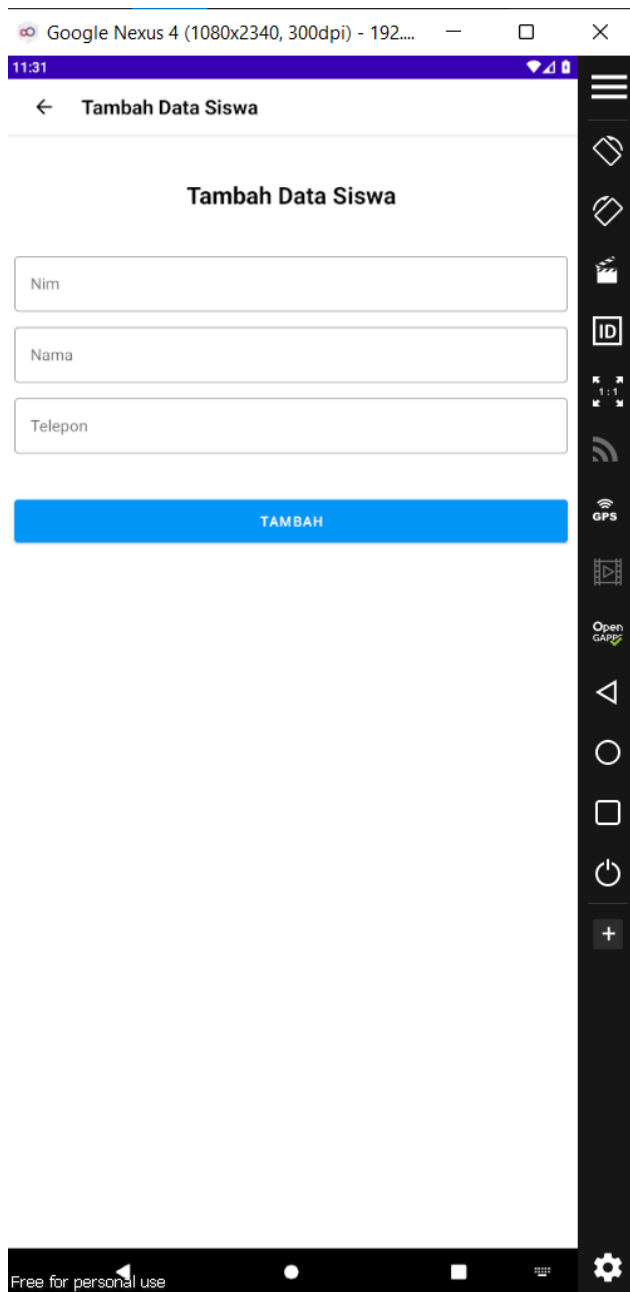
```

### 1.10. Hasil Run Program

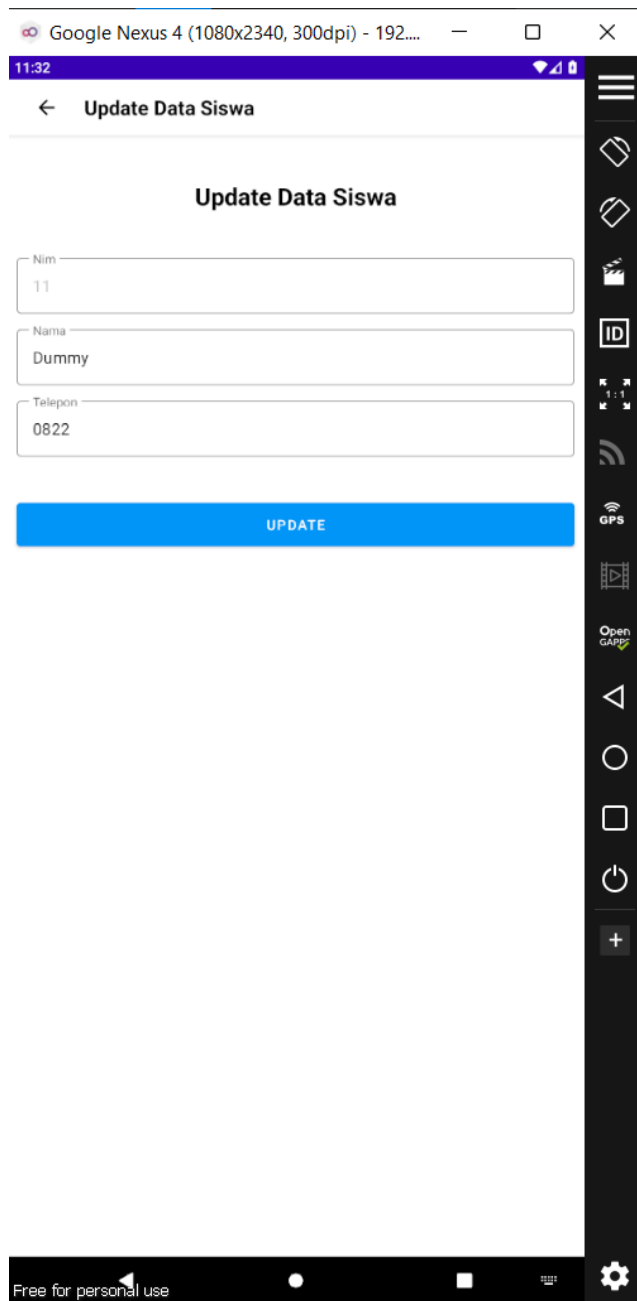


Home

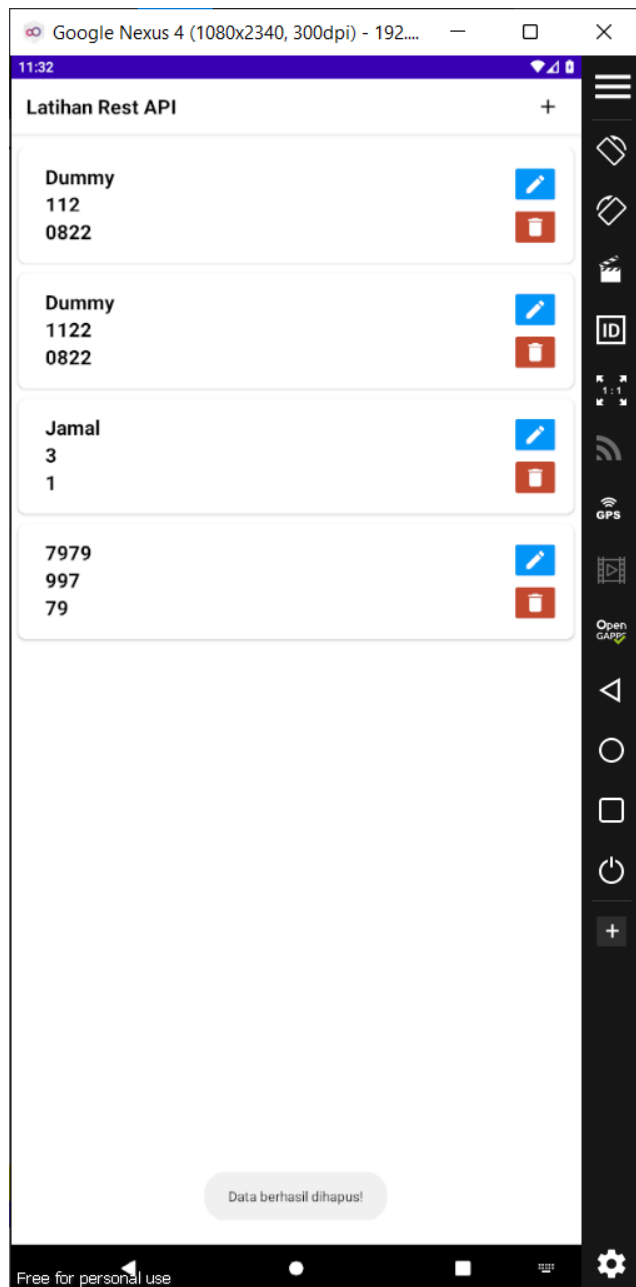




**Tambah**



**Update**



**Delete**