

BAB IV

RECYCLER VIEW

4.1. Tujuan Pembelajaran

- Mahasiswa dapat menggunakan *RecyclerView* ke dalam aplikasi.
- Mahasiswa mengetahui berbagai macam *RecyclerView* seperti *List*, *Grid*, dan *Card*.
- Mahasiswa dapat membuat berbagai macam *RecyclerView* seperti *List*, *Grid*, dan *Card*.

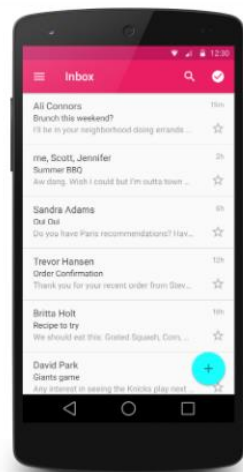
4.2. Software yang dibutuhkan

- Java JDK
- Android Studio 4.2
- SDK API 26

4.3. RecyclerView

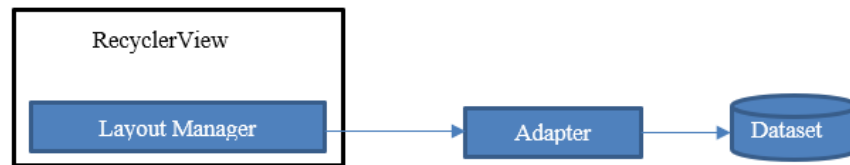
Untuk menampilkan data dalam bentuk *list*, biasanya kita menggunakan *ListView* misalnya untuk menampilkan data-data yang berasal dari kontak di *device* pengguna ke layar. Sangat sederhana, tetapi sangat berarti. Mengapa ? Pada dasarnya interaksi umum antara pengguna dengan aplikasi dalam menampilkan data dalam jumlah yang banyak adalah dengan menggunakan *list* yang bisa di *scroll* ke atas dan ke bawah.

ListView menjadi komponen pertama yang mengakomodasi hal tersebut, namun semenjak *Google* merilis pendekatan desain bernama *material design*, *RecyclerView* menjadi pilihan pertama yang harus *developer* gunakan. Gambar 1 merupakan contoh penggunaan *RecyclerView*. Anda masih bisa menggunakan kedua komponen tersebut secara berdampingan dalam satu aplikasi.



Gambar 1 : Penggunaan RecyclerView

Secara definisi *RecyclerView* adalah sebuah komponen tampilan (*widget*) yang lebih canggih ketimbang pendahulunya *ListView* dan bersifat lebih fleksibel. Yang paling penting adalah *RecyclerView* memiliki kemampuan untuk menampilkan data secara efisien dalam jumlah yang besar. Terlebih jika Anda memiliki koleksi data yang tiap elemennya mampu berubah-ubah sewaktu dijalankan (*runtime*) karena interaksi pengguna atau karena adanya pengaruh dari jaringan *internet*.

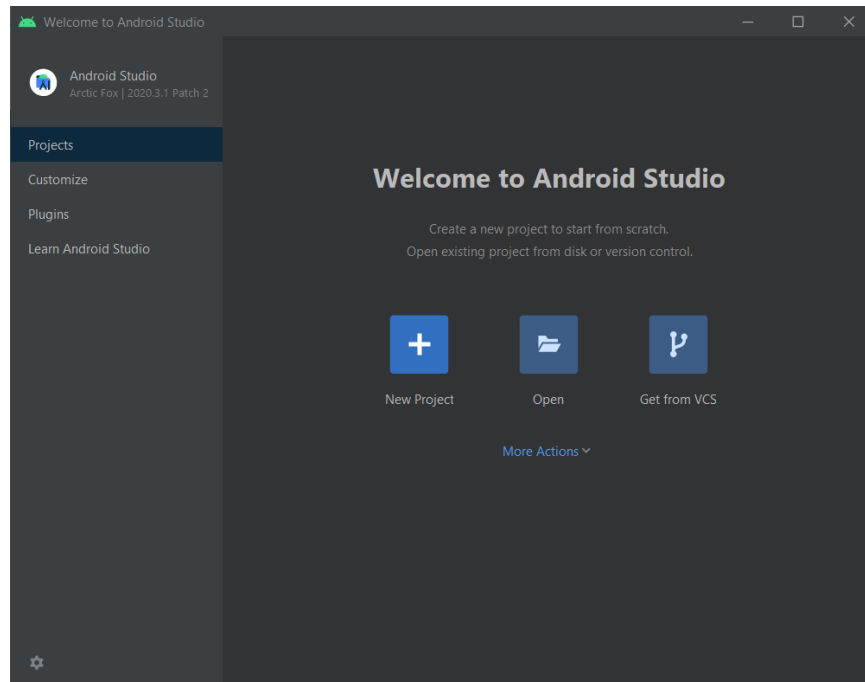


Gambar 2 : Komponen RecyclerView

1. *RecyclerView* dan *LayoutManager* : Komponen user interface yang bertugas untuk menampilkan data set yang dimiliki. Di dalamnya terdapat *LayoutManager* yang mana akan mengatur posisi tampilan data baik itu secara *List* (vertikal), *Grid* (baris dan kolom) atau *StaggeredGrid* (Grid yang memiliki susunan tak seragam/tak beraturan) .
2. *Adapter* : Komponen yang akan mengatur bagaimana tampilan data set ke dalam *RecyclerView*. Disinilah terjadi proses pengisian tampilan (*ViewInflate*) dari file layout xml untuk tiap elemen dari set data yang dimiliki sebelum dipasang (*bind*) ke dalam *RecyclerView*.
3. *Dataset* : Kumpulan data yang dimiliki dan ingin ditampilkan, bias berupa *Array*, *List* maupun objek map.

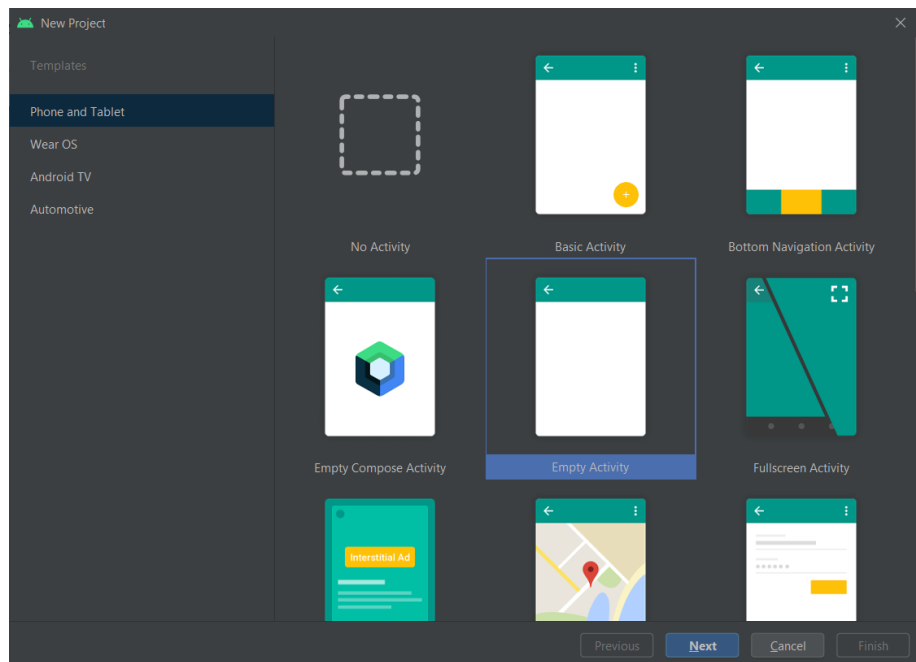
4.4. Langkah – langkah praktikum

1. Membuat project baru pada Android Studio dengan memilih New Project seperti pada gambar dibawah



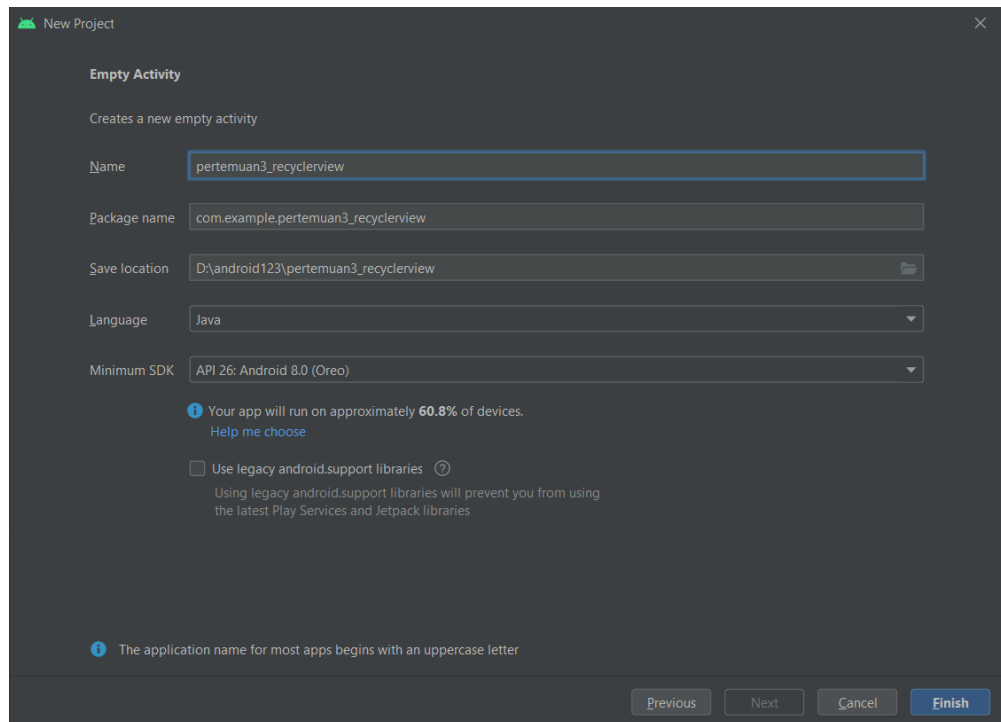
Gambar 3 : Halaman Awal Android Studio

2. Praktikan memilih “Empty Activity” pada “Phone and Tablet” seperti pada gambar dibawah



Gambar 4 : Empty Activity pada Phone and Tablet

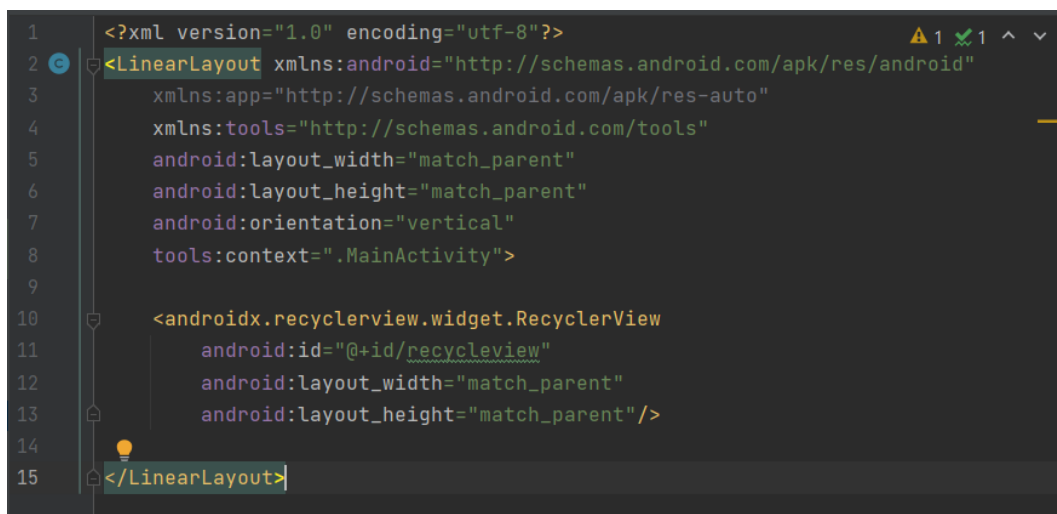
3. Mengisi identitas project seperti pada gambar. Disarankan memakai minimum SDK 26 (Android 8.0 Oreo). Jika sudah klik finish



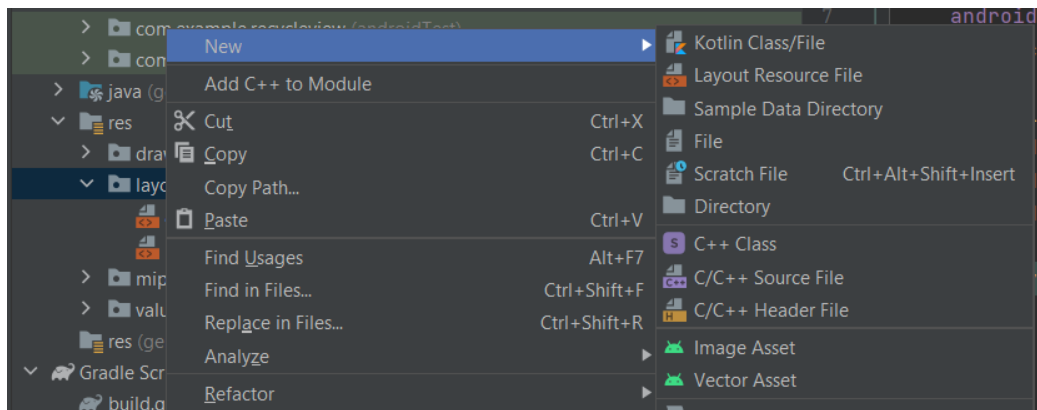
4. Buka Graddle Script pada build.gradle lalu tambahkan dependency dibawah ini, lalu klik “Sync Now” pada pop up graddle yang akan muncul.

```
implementation "androidx.recyclerview:recyclerview:1.2.1"
implementation "androidx.recyclerview:recyclerview-selection:1.1.0"
implementation "androidx.cardview:cardview:1.0.0"
```

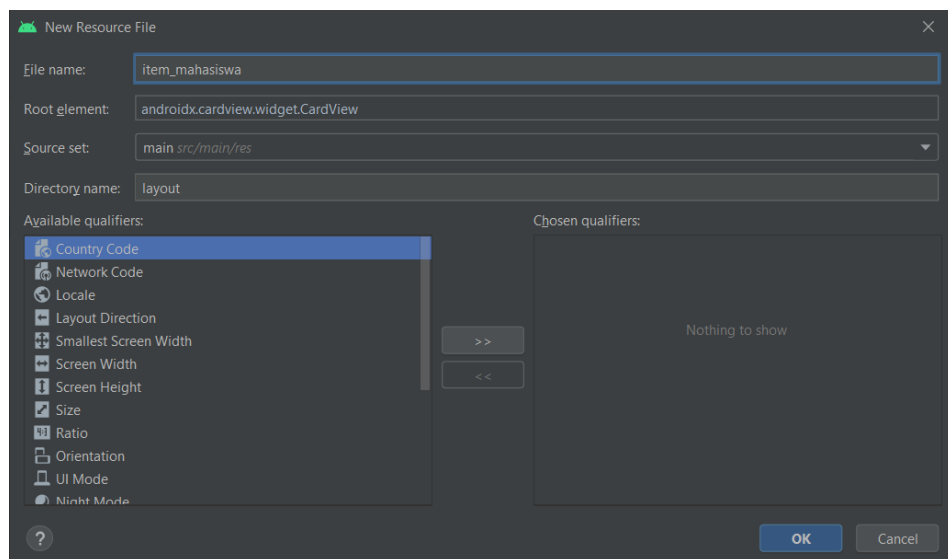
5. Ubahlah root element pada activity_main.xml pada Linear Layout, lalu tambahkan widget *RecyclerView* pada layout sehingga tampak seperti pada gambar dibawah :



6. Buatlah layout pada folder resource seperti pada gambar dibawah



7. Isilah identitas layout pada gambar



8. Buatlah textview pada layout item_mahasiswa.xml seperti pada dibawah

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.cardview.widget.CardView
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:elevation="3dp"
    app:cardCornerRadius="5dp"
    app:cardUseCompatPadding="true">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:padding="5dp"
        android:layout_margin="5dp">
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:id="@+id/txt_nama"/>
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:id="@+id/txt_nim"/>
    </LinearLayout>
</androidx.cardview.widget.CardView>
```

```

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:id="@+id/txt_angkatan"/>
    </LinearLayout>
</androidx.cardview.widget.CardView>

```

9. Ubahlah Code pada MainActivity.java seperti dibawah :

```

public class MainActivity extends AppCompatActivity {
    private RecyclerView recyclerView;
    private MahasiswaAdapter adapter;
    private ArrayList<Mahasiswa> mahasiswaArrayList;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        addData();
        recyclerView =
        (RecyclerView) findViewById(R.id.recycleview);
        adapter = new MahasiswaAdapter(mahasiswaArrayList);
        RecyclerView.LayoutManager layoutManager = new
        LinearLayoutManager(MainActivity.this);
        recyclerView.setLayoutManager(layoutManager);
        recyclerView.setAdapter(adapter);
    }

    private void addData() {
        mahasiswaArrayList = new ArrayList<>();
        mahasiswaArrayList.add(new Mahasiswa("Agung", "1218019",
        "2012"));
        mahasiswaArrayList.add(new Mahasiswa("Abi", "1118002",
        "2011"));
        mahasiswaArrayList.add(new Mahasiswa("Yudis", "1118031",
        "2011"));
    }
}

```

10. Buatlah Class baru dengan nama Mahasiswa.java lalu isilah codenya dengan setter dan getter untuk array

```

public class Mahasiswa {
    private String nama,nim,angkatan;

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

    public String getNama(){
        return nama;
    }

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

```

```

    public String getNim(){
        return nim;
    }

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

    public String getAngkatan(){
        return angkatan;
    }

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

```

11. Lalu buatlah class baru dengan nama MahasiswaAdapter.java, isikan kode dibawah

```

public class MahasiswaAdapter extends
RecyclerView.Adapter<MahasiswaAdapter.MahasiswaViewHolder>{

    private ArrayList<Mahasiswa> dataList;

    public MahasiswaAdapter(ArrayList<Mahasiswa> dataList) {
        this.dataList = dataList;
    }

    @NonNull
    @Override
    public MahasiswaAdapter.MahasiswaViewHolder
onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        LayoutInflater inflater =
LayoutInflater.from(parent.getContext());
        View view =
inflater.inflate(R.layout.item_mahasiswa, parent, false);
        return new MahasiswaViewHolder(view);
    }

    public void onBindViewHolder(MahasiswaViewHolder holder, int
position) {
        holder.txtNama.setText(dataList.get(position).getNama());
        holder.txtNim.setText(dataList.get(position).getNim());
        holder.txtAngkatan.setText(dataList.get(position).getAngkatan());
    }

    @Override
    public int getItemCount() {
        return (dataList != null) ? dataList.size() : 0;
    }

    public class MahasiswaViewHolder extends
RecyclerView.ViewHolder{
        private TextView txtNama, txtNim, txtAngkatan;

        public MahasiswaViewHolder(View itemView) {
            super(itemView);
            txtNama = (TextView)
itemView.findViewById(R.id.txt_nama);

```

```
        txtNim = (TextView)
itemView.findViewById(R.id.txt_nim);
        txtAngkatan = (TextView)
itemView.findViewById(R.id.txt_angkatan);
    }
}
```

4.5. Tugas Rumah

Menambahkan gambar pada setiap item yang akan di tampilkan pada RecyclerView.