

**LAPORAN PRAKTIKUM**  
**PENGEMBANGAN APLIKASI BERGERAK**  
**PRAKTIKUM 09 – RETROFIT**



**L0122034**  
**BINTANG HARIDA RAMADHAN**

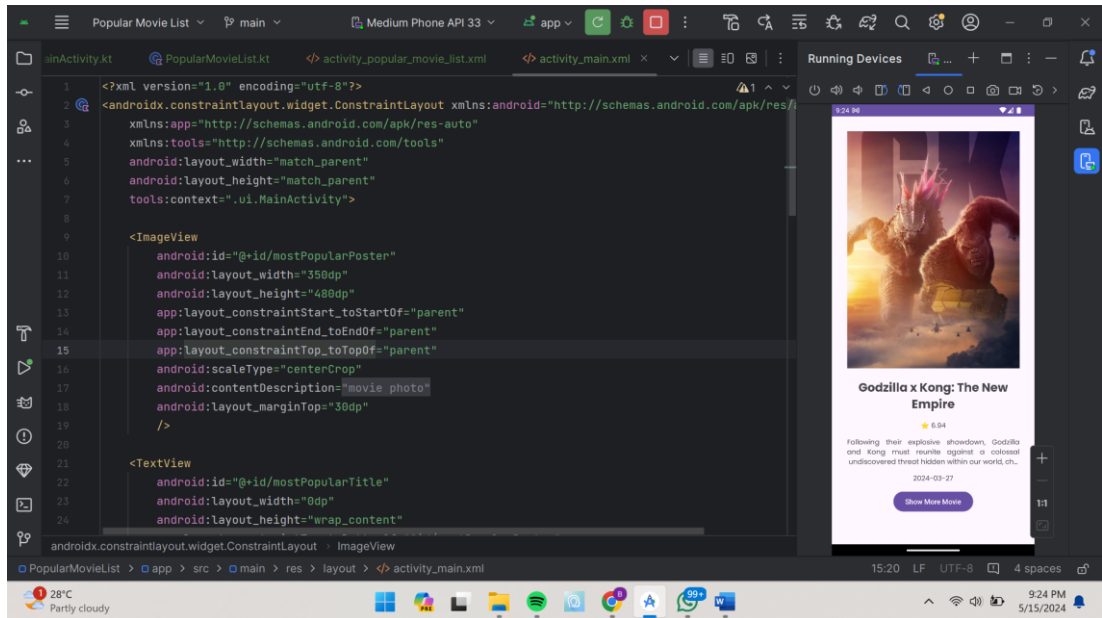
**PROGRAM STUDI INFORMATIKA**  
**FAKULTAS TEKNOLOGI INFORMASI DAN SAINS DATA**  
**UNIVERSITAS SEBELAS MARET**

**2024**

# BAB I

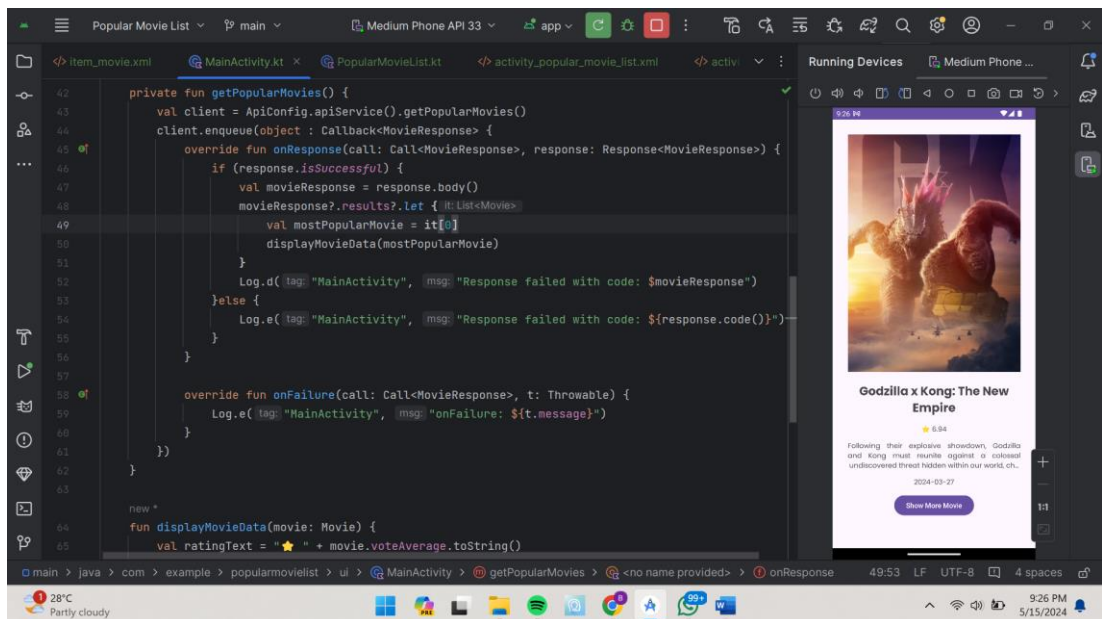
## SOURCE CODE

### 1. File “activity\_main.xml”



Layout ini merupakan layout dari aktivitas utama yang akan dijalankan. Di dalam layout ini dideklarasikan ImageView dan beberapa TextView yang akan menampilkan poster dari film dan beberapa data yang ada di dalam API sesuai dengan id masing masing View.

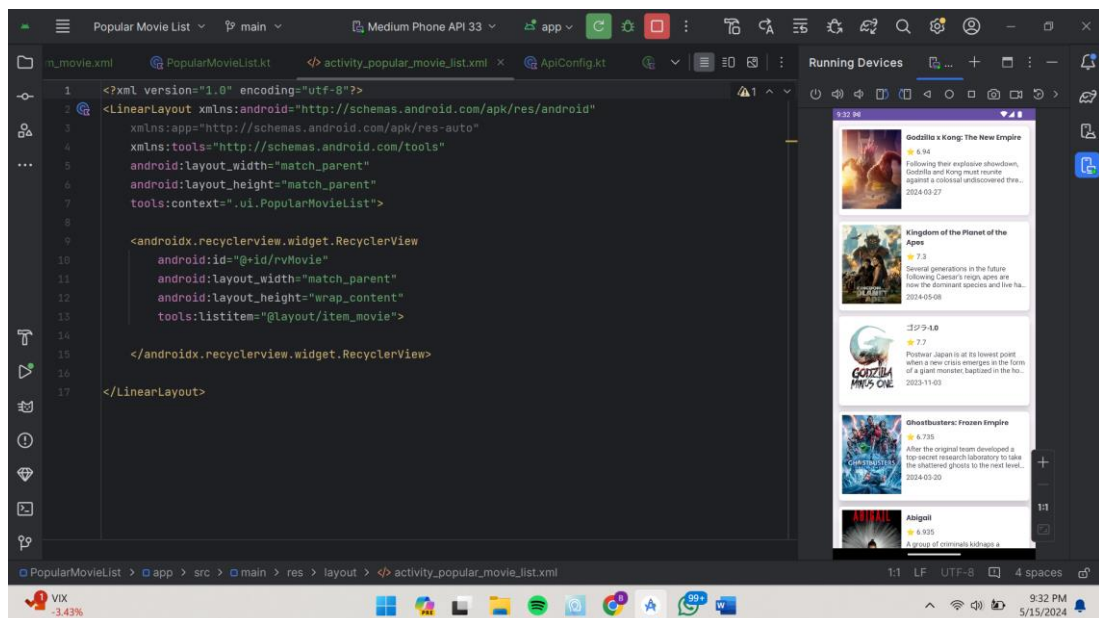
### 2. File “MainActivity.kt”



Activity main adalah activity utama pada program yang dibuat. Pada file ini digunakan untuk mengelola pengambilan data dari API dan menampilkannya. Sebelum

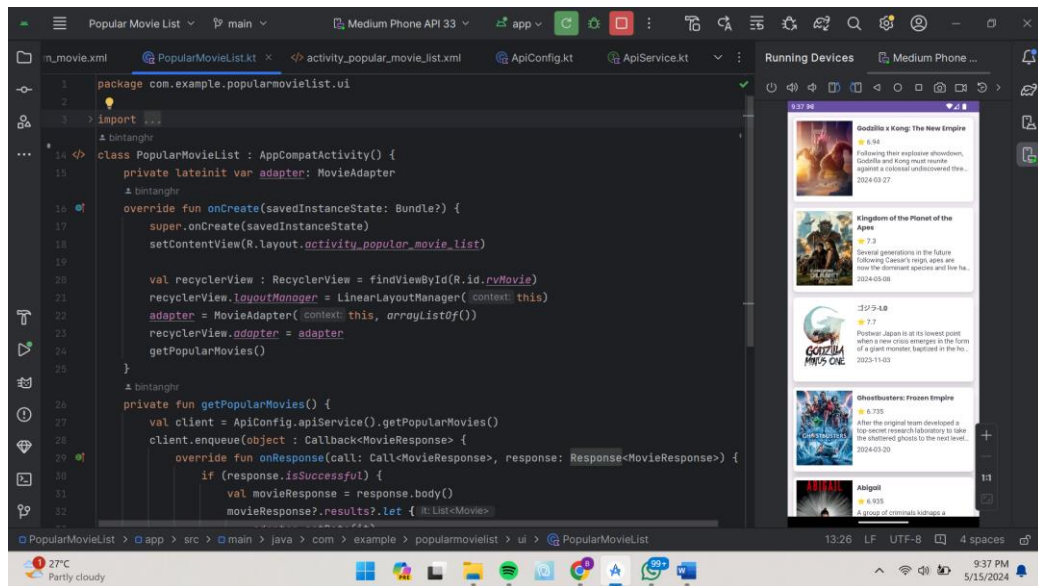
itu akan dilakukan inialisasi variable dari beberapa View pada method onCreate() yang nantinya masing masing View tersebut akan diisi dengan data dari API. Selanjutnya terdapat method getPopularMovies() yang berguna untuk mengambil data dari API menggunakan bantuan class ApiConfig, data hasil pengambilan dari API ini akan disimpan pada variable mostPopularMovie yang hanya mengambil satu data pertama dari list data yang ada. Data yang telah diambil tersebut selanjutnya akan digunakan untuk mengisi TextView dan ImageView di Activity Main. Untuk dapat menggunakan foto dari API, diperlukan bantuan library dari Glide. Selain itu juga terdapat penanganan untuk onClick pada button ShowMoreMovie yang bila ditekan akan melakukan intent ke PopularMovieList Activity.

### 3. File “activity\_popular\_movie\_list.xml”



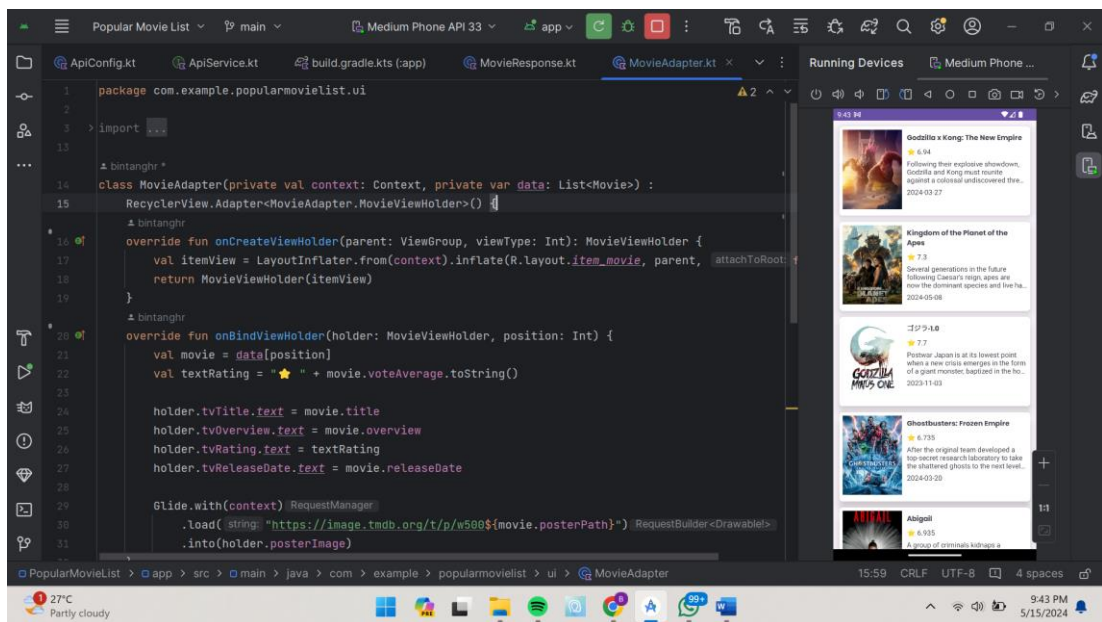
File ini adalah layout dari PopularMovieList Activity yang akan menampilkan RecyclerView dari list movie. Pada layout ini RecyclerView dideklarasikan dengan id rvMovie.

### 4. File “PopularMovieList.kt”



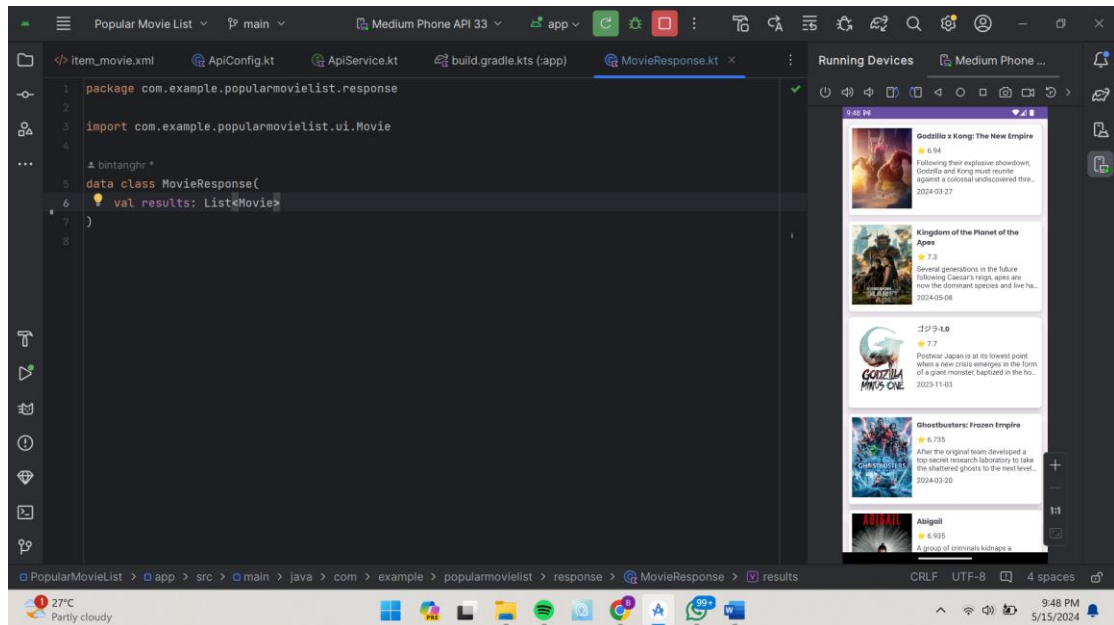
File ini akan digunakan untuk menangani pengambilan data dari API dan juga penggunaan data yang telah di ambil ke dalam RecyclerView. Pada method onCreate() diinisialisasikan variable recyclerView dan adapter yang keduanya akan saling berkaitan untuk dapat menampilkan RecyclerView. Selanjutnya terdapat method getPopularMovies() yang berguna untuk mengambil data dari API menggunakan bantuan class ApiConfig, data hasil pengambilan dari API ini akan disimpan pada adapter menggunakan method setData() yang akan menyimpan seluruh list data yang ada. Data yang telah diambil tersebut selanjutnya akan digunakan untuk mengisi TextView dan ImageView di RecyclerView yang akan diisi pada Adapter.

##### 5. File “MovieAdapter.kt”



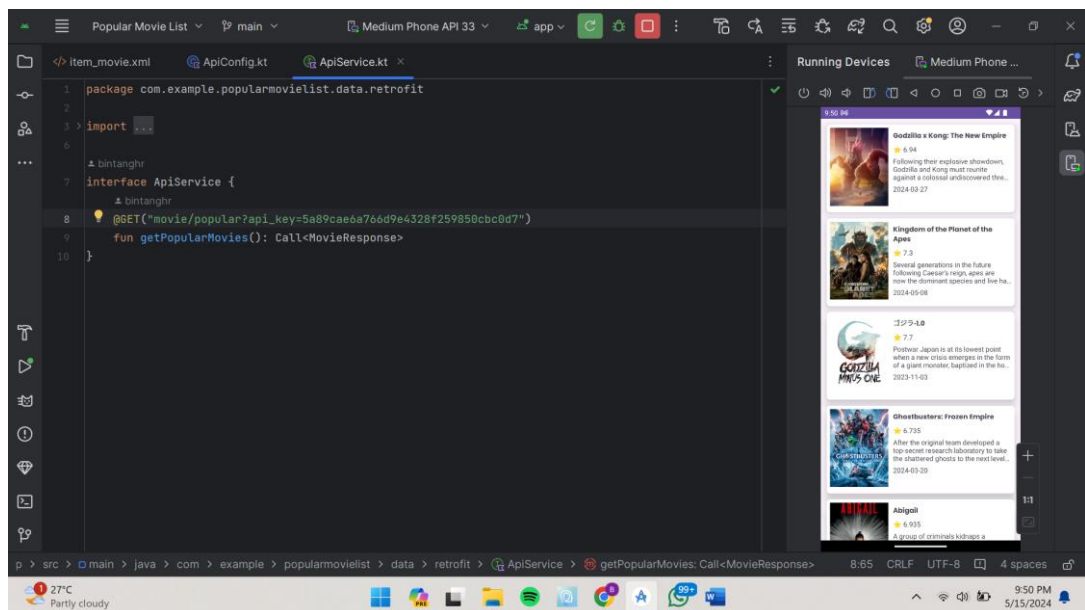
Pada class ini diberikan parameter konstruktor *context* dan *List* dari *Movie*. Di dalam definisi class tersebut terdapat class *MovieViewHolder* yang akan menampung masing masing dari *view* di dalam *card list (item\_movie.xml)*. Pada method *onBindViewHolder* akan diambil data *movie* dari posisi yang sesuai. Data data tersebut selanjutnya akan digunakan untuk memberi nilai pada masing masing *variable View*. Untuk dapat menggunakan foto dari API, diperlukan bantuan library dari Glide.

## 6. File “MovieResponse.kt”



Class data ini akan digunakan untuk membantu menyimpan data dalam bentuk List dari class *Movie*, hal ini dilakukan agar memudahkan dalam pengambilan data. Karena api yang saya gunakan seluruh datanya berada di dalam key “results”.

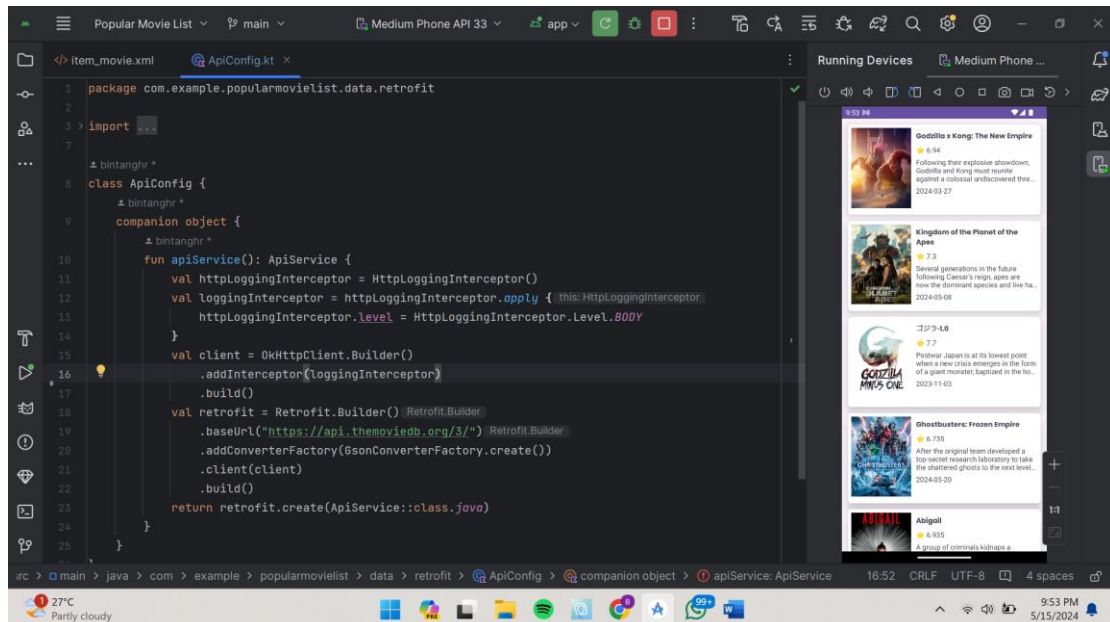
## 7. File “ApiService.kt”





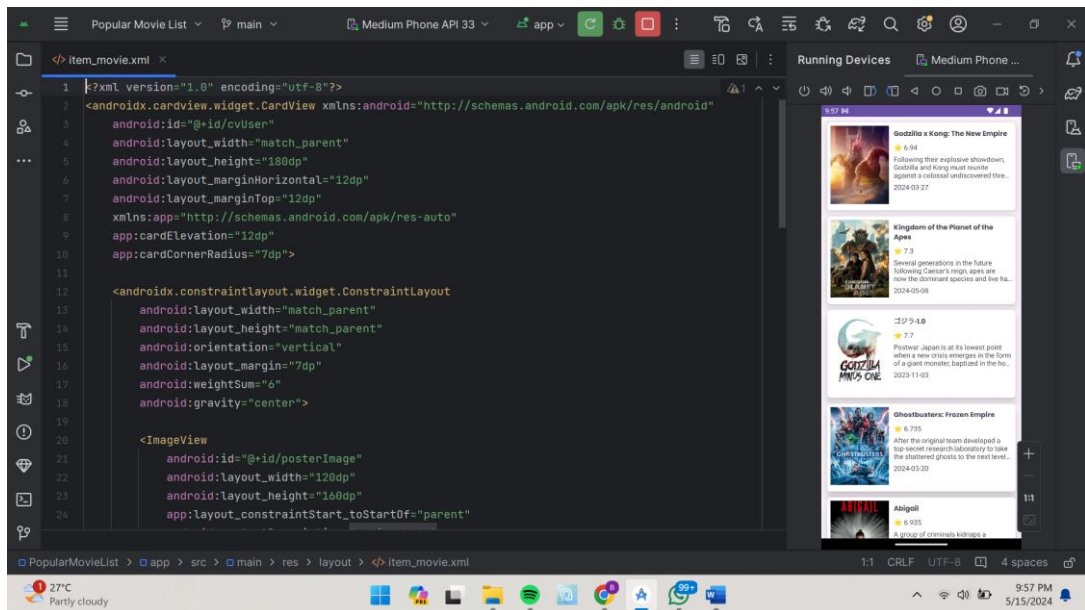
Interface ini akan digunakan di dalam ApiConfig. Interface ini berisi endpoint dari API yang saya gunakan. Karena API saya memerlukan API Key untuk dapat diakses, sehingga saya memberikan parameter api\_key pada endpoint. Endpoint ini akan memanggil fungsi getPopularMovies().

#### 8. File “ApiConfig.kt”



Pada file ini dilakukan konfigurasi Retrofit untuk melakukan panggilan networking ke API The Movie Database (TMDb) dengan URL dasar <https://api.themoviedb.org/3/>. Konfigurasi ini mencakup logging detail permintaan dan respons HTTP menggunakan HttpLoggingInterceptor dan mengonversi respons JSON menjadi objek Kotlin menggunakan Gson. Method apiService() akan mengembalikan instance dari interface ApiService yang dapat digunakan di seluruh aplikasi untuk melakukan panggilan jaringan.

#### 9. File “item\_movie.xml”

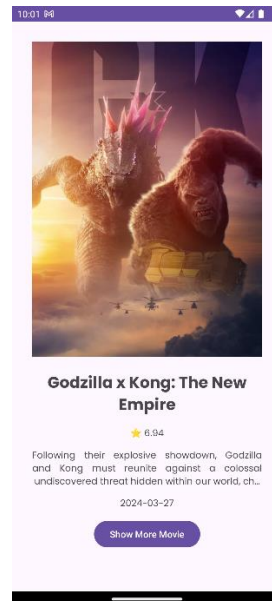


File layout ini digunakan untuk mendesain tampilan item (menggunakan CardView) dari Recycler View yang akan ditampilkan. Layout ini menggunakan ConstraintLayout untuk memudahkan dalam penataan Item View didalamnya. Card ini akan menampilkan foto dan beberapa informasi dari hasil pengambilan data melalui api. Masing masing View diberikan id yang unik untuk memudahkan dalam penampilan data di dalam Adapter.

## BAB II

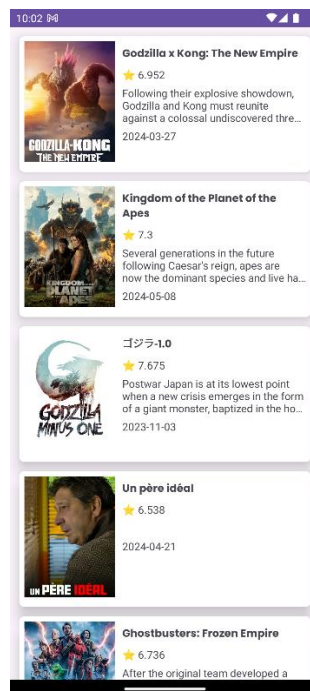
### USER INTERFACE

#### 1. Main Activity



Pada activity ini ditampilkan satu data dari api yang menampilkan poster movie, rating, overview, dan release date. Selain itu juga terdapat tombol yang akan menuju ke PopularMovieList Activity.

#### 2. PopulatMovieList Activity



Pada activity ini ditampilkan RecyclerView yang masing masing itemnya akan menampilkan gambar film, judul, rating, overview, dan tanggal rilis.



### **BAB III**

#### **KESIMPULAN**

Networking dengan Retrofit dapat memudahkan operasi jaringan di Android, menyediakan API sederhana untuk mengonfigurasi dan membuat permintaan HTTP. Retrofit mendukung konverter untuk mengonversi respons HTTP menjadi objek Kotlin dan mudah untuk menambahkan fungsionalitas tambahan seperti logging atau autentikasi.

API (Application Programming Interface) memungkinkan berbagai aplikasi untuk saling berkomunikasi, dengan RESTful API yang menggunakan HTTP untuk komunikasi. Metode GET dalam HTTP digunakan oleh klien untuk meminta data dari server tanpa mengubah sumber daya di server.