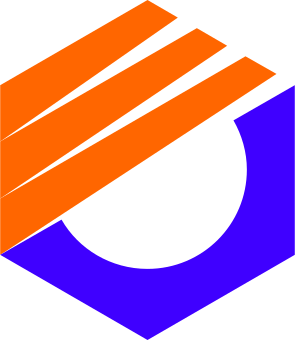
**LAPORAN TUGAS PERTEMUAN 15**

**“Mobile Programming”**



Disusun oleh

Nova Kensa Aura Dewi Oematan

201511054

Kelas 2B

**D3 TEKNIK INFORMATIKA**

**JURUSAN TEKNIK KOMPUTER DAN INFORMATIKA**

**POLITEKNIK NEGERI BANDUNG**

**2021**

**Lesson Learn**

Pertama kita harus mengsingkronisasikan gradle dengan retrofit, dengan cara mengimplementasikan :

implementation 'com.squareup.retrofit2:retrofit:2.4.0'  
implementation 'com.squareup.retrofit2:converter-gson:2.4.0'

Dalam pengimplementasian retrofit terdapat beberapa HTTP methods yang dapat dilakukan yaitu GET, POST, PUT, PATCH dan DELETE.

1. Penerapan GET request dilakukan pertama adalah membuat interface berisi method yang ber anotasi dengan @GET dan relative URL ke dalam API endpoint dari REST API yang akan kita lakukan query. Lalu setelah itu instance retrofit telah dibuat, baseURL pun telah didefinisikan dan juga menambahkan GSON sebagai converter dengan meneruskan GsonConverterFactory ke method pemanggilan objek dan nantinya akan dikembalikan oleh method GET dengan syarat 1 permintaan dan respon. Request tersebut dapat dijalankan asynchronous di latar belakang dan akan mengembalikan nilainya lagi ketika memanggil onResponse. Lalu ada pula onFailure yang akan dipanggil jika ada komunikasi dengan server atau proses respon yang salah. Dan ada pula isSuccesFull yang nantinya akan memeriksa kode respons antara 200 dan 300.
2. Untuk mengakses endpoint URL yang lebih dinamis dari REST API, kita bisa menggunakan blok pengganti dan parameter query. Ada beberapa anotasi pada retrofit diantaranya:

* Anotasi @Query argument yang akan ditambahkan retrofit bisa diteruskan pada URL sebagai string Query dengan syntax yang benar.
* Anotasi @URL, URL endpoint secara langsung diteruskan sebagai argument.
* Anotasi @POST digunakan untuk menentukan endpoint URL dan meneruskan objek yang akan dikirim ke server sebagai argument yang diberi anotasi sebagai @Body

GSON juga akan membuat serial objek kedalam format JSON sebelum dikirim ke sercer. Lalu jika kita ingin membuat serial objek dalam format lain seperti XML atau String, maka kita bisa menggunakan ConverterFactory yang lain agar formatnya terubah.

Retrofit juga mendukung format-urlencoded, dengan demikian keterangan method POST bisa ditambah dengan @FormUrlEncoded dan meneruskan argument @Field tunggal atau @FieldMap.

**Istilah pada penerapan retrofit**

* PUT dan PATCH merupakan method HTTP yang digunakan untuk mempebaharui data pada server, namun PATCH hanya mengubah field tertentu yang dikirim, sedangkan PUT menggantikan entity yang ada dengan body yang ada pada request.
* DELETE merupakan method yang digunakan untuk menghapus data pada server.
* PUT dan PATCH mengirim body pada permintaan yang beranotasu @Body dan di convert oleh GSON kedalam format JSON.
* Method DELETE bisa menentukan bahwa isi respons tidak ingin dibaca. Kode respon dari operasi DELETE jika berhasil dijalankan ada 200 (OK). Operasi DELETE juga dulakukan secara asynchronous dan hasilnya ditangani oleh onResponse dan onFailure.
* Interceptor logging digunakan untuk mencatat permintaan dan respons HTTP antara aplikasi dengan REST API. Untuk menggunakan interceptor logging, kita hanya perlu menambahkan dependency logging-interceptor OkHttp3 ke file gradle dan membuat instance HttpLoggingInterceptor, lalu setelahnya mengatur level log seperti (NONE, BASIC, HEADERS, atau BODY) dan menambahkannya ke instance OkHttpClient baru dengan addInterceptor dan diterukan ke Retrofit.Builder. Dengannya digunakan HttpLoggingInterceptor maka OkHttp akan secara otomatis mencatat permintaan dan respons HTTP yang masuk dan yang keluar ke logcat.
* Retrofit juga bisa menambahkan header ke method HTTP baik secara statis dengan menambahkan anotasi @Header atau @HeaderMap, atau menambahkan header ke semua method sekaligus menentukan OkHttp Interceptor.

Istilah atau konsep yang menurut saya penting adalah anotasi yang digunakan dalam retrofit seperti @GET, @ POST, @PUT, @PATCH, @ DELETE. Selain itu, dependency, method seperti onResponse, onFailure juga sangat penting dalam penerapan retrofit ini.

**Hasil Program**

|  |  |  |
| --- | --- | --- |
| Screenshot | Screenshot | Screenshot |
|  |  |  |
|  |  |  |
|  |  |  |