



REST API

Pemrograman Aplikasi Perangkat Bergerak

REST API SERVER

- 1) Server yang mengelola resource
- 2) Berbasis web
- 3) Berkomunikasi menggunakan protokol HTTP
- 4) Menggunakan HTTP request dan HTTP response
- 5) Dibuat dengan laravel/php, CI/php, Spring/java, Django/python dll

JSON

Format data yang banyak digunakan REST API

```
{  
  "id": 1,  
  "judul": "Implementasi Sistem Informasi",  
  "deskripsi": "Buku Mata kuliah Implementasi Sistem Informasi"  
}
```

HTTP Method

- 1) GET, membaca data
- 2) POST, membuat/create resource
- 3) PUT, update resource
- 4) DELETE, menghapus resource



Android Studio

Android

- 1) Menambahkan dependency di build.gradle
- 2) Konfigurasi permission di AndroidManifest.xml
- 3) Membuat class : model, service dan retrofit client
- 4) Menggunakan retrofit client di activity

Dependency

Library yang menyediakan HTTP client

```
dependencies {  
    implementation 'com.squareup.retrofit2:converter-gson:2.9.0'  
    implementation 'com.squareup.retrofit2:retrofit:2.9.0'
```

Permission

AndroidManifest.xml

```
<uses-permission android:name="android.permission.INTERNET" />
<application
    .....
    android:usesCleartextTraffic="true">
```


SERVICE

```
public interface PerpustakaanService {  
    @GET("buku")  
    Call<List<Buku>> listBuku();  
  
    @GET("buku/{id}")  
    Call<Buku> getBuku(@Path("id") String id);  
}
```

Model

```
public class Buku {  
    private String id;  
    private String judul;  
    private String deskripsi;  
    .....  
    //getter dan setter
```

Retrofit Client

```
public class RetrofitClient {  
    public static final String BASE_URL = "http://192.168.100.55/api/";  
    private static Retrofit retrofit = null;  
    public static Retrofit getClient() {  
        if (retrofit==null) {  
            retrofit = new Retrofit.Builder()  
                .baseUrl(BASE_URL)  
                .addConverterFactory(GsonConverterFactory.create())  
                .build();  
        }  
        return retrofit;  
    }  
}
```

Activity

```
PerpustakaanService perpustakaanService = RetrofitClient.getClient().create(PerpustakaanService.class);
Call<List<Buku>> listRequest = perpustakaanService.listBuku();
listRequest.enqueue(new Callback<List<Buku>>() {
    @Override
    public void onResponse(Call<List<Buku>> call, Response<List<Buku>> response) {
        if (response.isSuccessful()) {
            List<Buku> list = response.body();
            listBuku.addAll(list);
            adapter.notifyDataSetChanged();
        } else {
            Log.d("error", "" + response.errorBody());
            Toast.makeText(MainActivity.this, "" + response.errorBody(), Toast.LENGTH_SHORT).show();
        }
    }

    @Override
    public void onFailure(Call<List<Buku>> call, Throwable t) {
        Log.d("error", "" + t.getMessage());
        Toast.makeText(getApplicationContext(), "Error : " + t.getMessage(), Toast.LENGTH_LONG).show();
    }
});
```

Kode sumber Android : https://gitlab.com/p989/java/rest_api

Kode sumber Rest Api Server PHP: <https://gitlab.com/p989/php/perpustakaan>



Thank you