

A. Pengenalan API Service

Sebelum masuk kedalam penggunaan API di android, yang harus dilakukan adalah mengenalkan mengenai webservice dalam hal ini adalah microservice, salah satu microservice yang terkenal adalah output JSON. Untuk mengakses API tersebut terlebih dahulu kita harus paham mengenai struktur dari API yang akan diakses. Untuk mengetahui struktur dari API yang akan diakses kita membutuhkan beberapa software tambahan yakni postman.

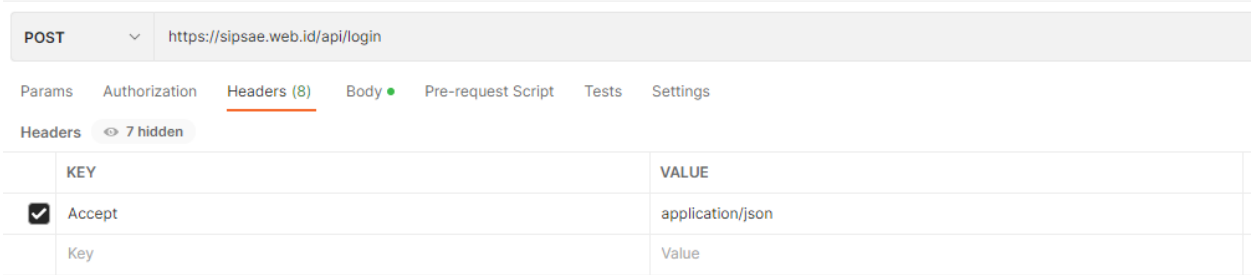
1. Install postman

Silahkan download Postman di link [Download Postman | Get Started for Free](#)

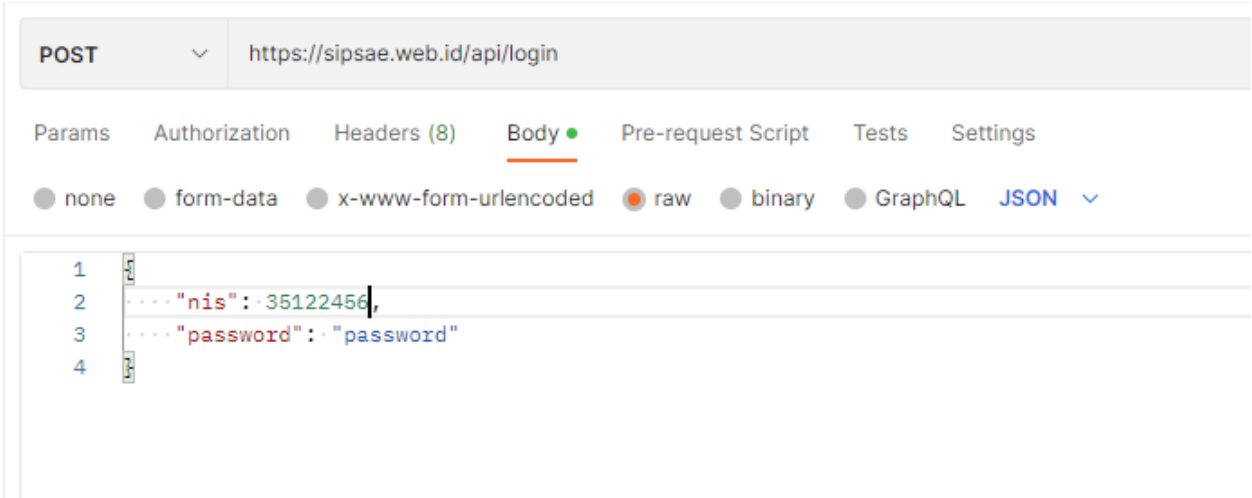
2. Akses API

Untuk akses API pada Postman silahkan masukkan link berikut pada postman
<https://sipsae.web.id/api/login>

3. Kemudian pada setting heads tambahkan berikut



4. Dan lakukan pengujian pada body sebagai berikut (untuk username yang bisa digunakan adalah 14562789,21345678, 35122456 dengan password semuanya adalah password)



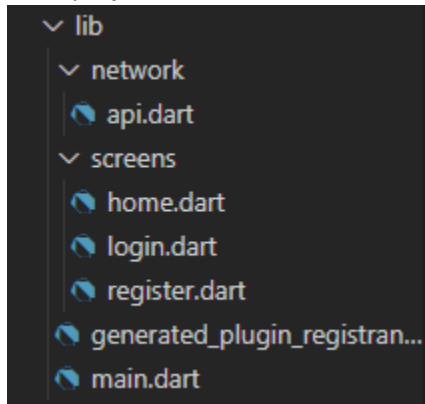
Perhatikan output pada aplikasi

B. Membuat aplikasi dengan flutter

1. Tambahkan dependencies sebagai berikut

```
dependencies:  
  shared_preferences: ^0.5.12+4  
  http: ^0.12.2
```

2. Pada project structure buat folder dan file sebagai berikut :



3. Pada api.dart

```
import 'dart:convert';  
import 'package:http/http.dart' as http;  
import 'package:shared_preferences/shared_preferences.dart';  
  
class Network {  
  final String _url = 'https://sipsae.web.id/api';  
  // 192.168.1.2 is my IP, change with your IP address  
  var token;  
  
  _getToken() async {  
    SharedPreferences localStorage = await SharedPreferences.getInstance();  
    token = jsonDecode(localStorage.getString('token').toString())['token'];  
  }  
  
  auth(data, apiURL) async {  
    var fullUrl = _url + apiURL;  
    print(fullUrl);  
    return await http.post(Uri.parse(fullUrl),  
      body: jsonEncode(data), headers: _setHeaders());  
  }  
  
  getData(apiURL) async {  
    var fullUrl = _url + apiURL;  
    await _getToken();  
    return await http.get(  
      Uri.parse(fullUrl),  
      headers: _setHeaders(),  
    );  
  }  
  
  _setHeaders() => {  
    'Content-type': 'application/json',  
    'Accept': 'application/json',  
    'Authorization': 'Bearer $token',  
  };  
}
```

4. Pada register.dart buat tampilan register yang terdiri dari student_nis dan password, kemudian tambahkan kode berikut untuk post kedalam APInya :

```
void _register() async {
  setState(() {
    _isLoading = true;
  });
  var data = {'name': name, 'email': email, 'password': password};

  var res = await Network().auth(data, '/register');
  var body = json.decode(res.body);
  if (body['success']) {
    SharedPreferences localStorage = await SharedPreferences.getInstance();
    localStorage.setString('token', json.encode(body['token']));
    localStorage.setString('user', json.encode(body['user']));
    Navigator.pushReplacement(
      context,
      MaterialPageRoute(builder: (context) => Home()),
    );
  } else {
    if (body['message']['name'] != null) {
      _showMsg(body['message']['name'][0].toString());
    } else if (body['message']['email'] != null) {
      _showMsg(body['message']['email'][0].toString());
    } else if (body['message']['password'] != null) {
      _showMsg(body['message']['password'][0].toString());
    }
  }

  setState(() {
    _isLoading = false;
  });
}
```

5. Pada login karena hanya dibutuhkan username dan password maka untuk request hanya akan mengirimkan data username dan password sehingga pada request dibutuhkan sebagai

berikut :

```
void _login() async {
  setState(() {
    _isLoading = true;
  });
  var data = {'nis': email, 'password': password};

  var res = await Network().auth(data, '/login');
  var body = json.decode(res.body);
  // var rest = body['data'] as List;
  print(body);
  if (res.statusCode == 200) {
    var data = json.decode(res.body);
    SharedPreferences localStorage = await SharedPreferences.getInstance();
    localStorage.setString('token', json.encode(body['token']));
    localStorage.setString('name', json.encode(body['name']));
    localStorage.setString('nis', json.encode(body['nis']));
    Navigator.pushReplacement(
      context,
      new MaterialPageRoute(builder: (context) => Home()),
    );
  } else {
    _showMsg(body['message']);
  }

  setState(() {
    _isLoading = false;
  });
}
```

6. Sedangkan pada home akan dilakukan pengambilan data sesuai dengan shared preference yang sudah didapat dari API, untuk kode pengambilannya akan sebagai berikut

```
_loadUserData() async {
  SharedPreferences localStorage = await SharedPreferences.getInstance();
  var user = jsonDecode(localStorage.getString('name').toString());

  if (user != null) {
    setState(() {
      name = user['name'];
    });
  }
}
```

Sedangkan untuk logout akan sebagai berikut :

```
void logout() async {  
  var res = await Network().getData('/logout');  
  var body = json.decode(res.body);  
  if (body['success']) {  
    SharedPreferences localStorage = await SharedPreferences.getInstance();  
    localStorage.remove('user');  
    localStorage.remove('token');  
    Navigator.pushReplacement(  
      context, MaterialPageRoute(builder: (context) => Login()));  
  }  
}
```