

	LAPORAN PRAKTIKUM	PERTEMUAN 10
	PEMROGRAMAN PERANGKAT BERGERAK	
	NETWORKING FLUTTER	

A. IDENTITAS

NIM : 2103067
 Nama Lengkap : FATWAH FIRMANSYAH SLAMET RIYADI
 Kelas : D3TI2C
 Program Studi : D3 Teknik Informatika
 Jurusan : Teknik Informatika

B. WAKTU

Tanggal : 4 Mei 2023
 Waktu : 12.40 s/d 15.10

C. TUGAS

Pada tugas kali ini, saya membuat project yang hampir sama dengan paktikum, namun saya tambahkan properti tambahan sesuai dengan tugas yang diberikan. Pada tugas kalli ini untuk tampilan juga hampir sama dengan praktiku, saya hanya rubah sedikit tampilannya.

1. Membuat tabel

Untuk database sudah saya buat sebelumnya dengan nama db_mahasiswa, kemudian kita buat tabel didalamnya dengan struktur sebagai berikut.

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/> 1	nim	varchar(50)	utf8mb4_general_ci		No	None			Change Drop More
<input type="checkbox"/> 2	nama	text	utf8mb4_general_ci		No	None			Change Drop More
<input type="checkbox"/> 3	kelas	varchar(10)	utf8mb4_general_ci		No	None			Change Drop More
<input type="checkbox"/> 4	jurusan	text	utf8mb4_general_ci		No	None			Change Drop More
<input type="checkbox"/> 5	kehadiran	int(3)			No	None			Change Drop More
<input type="checkbox"/> 6	tugas	int(3)			No	None			Change Drop More
<input type="checkbox"/> 7	uts	int(3)			No	None			Change Drop More
<input type="checkbox"/> 8	uas	int(3)			No	None			Change Drop More
<input type="checkbox"/> 9	nilai_akhir	int(3)			No	None			Change Drop More

2. Membuat file PHP

File PHP yang saya gunakan ini merupakan file modifikasi dari file PHP praktikum.

File **config.php**

```
<?php
$conn = mysqli_connect("localhost","root","","db_mahasiswa");
```

File insert.php

```
<?php
include_once('config.php');
if($_SERVER['REQUEST_METHOD'] == "POST"){
    // Get data from the REST client
    $nim = isset($_POST['nim']) ? mysqli_real_escape_string($conn, $_POST['nim']) : "";
    $nama = isset($_POST['nama']) ? mysqli_real_escape_string($conn, $_POST['nama']) : "";
    $kelas = isset($_POST['kelas']) ? mysqli_real_escape_string($conn, $_POST['kelas']) : "";
    $jurusan = isset($_POST['jurusan']) ? mysqli_real_escape_string($conn, $_POST['jurusan']) : "";
    $kehadiran = isset($_POST['kehadiran']) ? mysqli_real_escape_string($conn, $_POST['kehadiran']) : "";
    $tugas = isset($_POST['tugas']) ? mysqli_real_escape_string($conn, $_POST['tugas']) : "";
    $uts = isset($_POST['uts']) ? mysqli_real_escape_string($conn, $_POST['uts']) : "";
    $uas = isset($_POST['uas']) ? mysqli_real_escape_string($conn, $_POST['uas']) : "";
    $nilai_akhir = ($kehadiran+$tugas+$uts+$uas) / 4;
    // Insert data into database
    $sql = "INSERT INTO tb_mahasiswa(nim,nama,kelas,jurusan,kehadiran,tugas,uts,uas,nilai_akhir) VALUES('$nim','$nama','$kelas','$jurusan',
    '$kehadiran','$tugas','$uts','$uas','$nilai_akhir')";
    $post_data_query = mysqli_query($conn, $sql);
    if($post_data_query){
        $json = array("status" => 1, "Success" => "Insert Success");
    }
    else{
        $json = array("status" => 0, "Error" => "Error Insert Data");
    }
}
else{
    $json = array("status" => 0, "Info" => "Request method not accepted!");
}
mysqli_close($conn);
// Set Content-type to JSON
header('Content-type: application/json');
echo json_encode($json);
```

File list_mahasiswa.php

```
<?php
include_once('config.php');

$sql = "SELECT * FROM tb_mahasiswa";
$get_data_query = mysqli_query($conn, $sql) or die(mysqli_error($conn));
if(mysqli_num_rows($get_data_query) != 0){
    $result = array();

    while($r = mysqli_fetch_array($get_data_query)){
        extract($r);
        $result[] = array("nim" => $nim, "nama" => $nama, 'kelas' => $kelas, 'jurusan' => $jurusan, 'kehadiran' =>
        $kehadiran, 'tugas' => $tugas, 'uts' => $uts, 'uas' => $uas, 'nilai_akhir' => $nilai_akhir);
    }
    $json = array("status" => 1, "info" => $result);
}
else{
    $json = array("status" => 0, "error" => "mahasiswa not found!");
}
mysqli_close($conn);
// Set Content-type to JSON
header('Content-type: application/json');
echo json_encode($json);
```

3. Persiapan di Flutter

Kita buat project baru flutter terlebih dahulu dengan menekan tombol **ctrl+shift+p** kemudian konfigurasi untuk app yang akan kalian buat.

Untuk tugas kali ini saya menambahkan beberapa library, yaitu Dio, HTTP, dan flutter_easyloading.

```
dependencies:
  cupertino_icons: ^1.0.5
  dio: ^5.1.1
  flutter:
    sdk: flutter
  flutter_easyloading: ^3.0.5
  http: ^0.13.6
```

4. Coding di Flutter

Pertama saya membuat sebuah file dart pada folder lib dengan nama **v_json.dart** kemudian kita buat class **statefulWidget** didalamnya.

```
class ViewJson extends StatefulWidget {  
  const ViewJson({super.key});  
  
  @override  
  State<ViewJson> createState() => _ViewJsonState();  
}  
  
class _ViewJsonState extends State<ViewJson> {  
  Dio dio = Dio();  
  List<Mahasiswa> listMahasiswa = [];  
  final TextEditingController _controllerNim = TextEditingController();  
  final TextEditingController _controllerNama = TextEditingController();  
  final TextEditingController _controllerKelas = TextEditingController();  
  final TextEditingController _controllerJurusan = TextEditingController();  
  final TextEditingController _controllerKehadiran = TextEditingController();  
  final TextEditingController _controllerTugas = TextEditingController();  
  final TextEditingController _controllerUTS = TextEditingController();  
  final TextEditingController _controllerUAS = TextEditingController();  
  final GlobalKey<ScaffoldState> _scaffold = GlobalKey<ScaffoldState>();  
  
  @override  
  void initState() { ...  
  
  Future<void> getMahasiswa() async { ...  
  
  Future<void> insertMahasiswa() async { ...  
  
  @override  
  Widget build(BuildContext context) {  
    return MaterialApp(  
  }  
}
```

Setelah itu kita buat class model dengan nama **Mahasiswa** ya berisikan prooerti sebagai berikut.

```

class Mahasiswa {
    final String nim;
    final String nama;
    final String kelas;
    final String jurusan;
    final String kehadiran;
    final String tugas;
    final String uts;
    final String uas;
    final String nilaiAkhir;
    Mahasiswa({
        required this.kehadiran,
        required this.tugas,
        required this.uts,
        required this.uas,
        required this.nilaiAkhir,
        required this.nim,
        required this.nama,
        required this.kelas,
        required this.jurusan});
}

```

Setelah itu pada class `_viewJsonState` kita tambahkan beberapa variabel yang digunakan untuk menampung instansiasi dari package tambahan, menampung controller untuk text field, dan lain sebagainya.

```

class _ViewJsonState extends State<ViewJson> {
    Dio dio = Dio();
    List<Mahasiswa> listMahasiswa = [];
    final TextEditingController _controllerNim = TextEditingController();
    final TextEditingController _controllerNama = TextEditingController();
    final TextEditingController _controllerkelas = TextEditingController();
    final TextEditingController _controllerjurusan = TextEditingController();
    final TextEditingController _controllerKehadiran = TextEditingController();
    final TextEditingController _controllerTugas = TextEditingController();
    final TextEditingController _controllerUTS = TextEditingController();
    final TextEditingController _controllerUAS = TextEditingController();
    final GlobalKey<ScaffoldState> _scaffold = GlobalKey<ScaffoldState>();
}

```

Kemudian kita buat sebuah fungsi yang digunakan untuk mendapatkan data, yang berupa data response json dari API kemudian data tersebut kita fetch dengan looping lalu dimasukan ke variabel `listMahasiswa` yang merupakan list of model mahasiswa.

```

Future<void> getMahasiswa() async {
  Response response;
  response = await dio
    .get("http://192.168.1.4:8080/api-mahasiswa/list_mahasiswa.php");
  if (response.data['status'] == 1) {
    setState(() {
      for (int i = 0; i < response.data['info'].length; i++) {
        listMahasiswa.add(Mahasiswa(
          nim: response.data['info'][i]['nim'],
          nama: response.data['info'][i]['nama'],
          kelas: response.data['info'][i]['kelas'],
          jurusan: response.data['info'][i]['jurusan'],
          kehadiran: response.data['info'][i]['kehadiran'],
          tugas: response.data['info'][i]['tugas'],
          uts: response.data['info'][i]['uts'],
          uas: response.data['info'][i]['uas'],
          nilaiAkhir: response.data['info'][i]['nilai_akhir'],
        ));
      }
    });
  }
}

```

Pada fungsi tersebut saya menggunakan dio package untuk koneksi dan get data dari API karena untuk proses fetching datanya lebih mudah. Untuk alamat IP itu tergantung alamat IP dari komputer kalian.

Setelah itu kita buat agar fungsi yang telah kita buat langsung di eksekusi, dengan memanggilnya pada fungsi override berikut.

```

@override
void initState() {
  getMahasiswa();
  super.initState();
}

```

Kemudian kita buat lagi sebuah fungsi yang digunakan untuk melakukan insert data ke database melalui API.

```

Future<void> insertMahasiswa() async {
  String? nim = _controllerNim.text;
  String? nama = _controllerNama.text;
  String? kelas = _controllerkelas.text;
  String? jurusan = _controllerjurusan.text;
  String? kehadiran = _controllerKehadiran.text;
  String? tugas = _controllerTugas.text;
  String? uts = _controllerUTS.text;
  String? uas = _controllerUAS.text;

  print([nim, nama, kelas, jurusan, kehadiran, tugas, uts, uas]);

  var response = await http.post(
    Uri.parse("http://192.168.1.4:8080/api-mahasiswa/insert.php"),
    body: {
      'nim': nim,
      'nama': nama,
      'kelas': kelas,
      'jurusan': jurusan,
      'kehadiran': kehadiran,
      'tugas': tugas,
      'uts': uts,
      'uas': uas,
    }
  );
}

```

```

if (response.statusCode == 200) {
  print("ini response ${response.body}");
  setState(() {
    _controllerNim.text = "";
    _controllerNama.text = "";
    _controllerkelas.text = "";
    _controllerjurusan.text = "";
    _controllerKehadiran.text = "";
    _controllerTugas.text = "";
    _controllerUTS.text = "";
    _controllerUAS.text = "";
    getMahasiswa();
  });
}
}

```

Fungsi tersebut akan mendapatkan data dari controller inputan, kemudian data yang berupa text tersebut kita simpan ke sebuah variabel, kemudian data pada variabel tersebut akan dikirimkan melalui API untuk dimasukan ke dalam database. Untuk koneksi dengan API saya menggunakan package HTTP karena entah kenapa jika menggunakan dio data yang masuk ke database itu kosong, data yang dibaca melalui controller tidak terdeteksi oleh dio.

Setelah itu kita buat UI untuk melakukan input data pada mobile device kita. Untuk tampilan UI hampir sama dengan yang ada pada praktikum saya hanya melakukan sedikit perubahan tampilan pada data yang ditampilkan

dan penambahan beberapa field inputan untuk input nilai mahasiswa.

```
Row(  
  mainAxisAlignment: MainAxisAlignment.spaceBetween,  
  crossAxisAlignment: CrossAxisAlignment.center,  
  children: [  
    Flexible(  
      child: TextField(  
        controller: _controllerKehadiran,  
        keyboardType: TextInputType.number,  
        decoration: const InputDecoration(hintText: "Kehadiran"),  
      )), // TextField // Flexible  
    const SizedBox(  
      width: 10,  
    ), // SizedBox  
    Flexible(  
      child: TextField(  
        controller: _controllerTugas,  
        keyboardType: TextInputType.number,  
        decoration: const InputDecoration(hintText: "Tugas"),  
      )), // TextField // Flexible  
  ],  
), // Row  
const SizedBox(  
  width: 10,  
), // SizedBox
```

```
Column(  
  mainAxisAlignment: MainAxisAlignment.start,  
  crossAxisAlignment: CrossAxisAlignment.start,  
  children: [  
    Text("Nama Mhs: ${listMahasiswa[i].nama}"),  
  ],  
) // Column  
const SizedBox(  
  height: 5,  
) // SizedBox  
const SizedBox(  
  width: 10,  
) // SizedBox
```

Berikut ini adalah tampilan UI nya.

8:28

Parsing Json

Form Input

Nim	kelas
Nama Mhs	Jurusan
Kehadiran	Tugas
UTS	UAS

Simpan

Data Mahasiswa

Data Kosong

Selanjutnya kita akan coba memasukan data melalui mobile device kita dengan data sebagai berikut.

8:29 53

Parsing Json

Form Input

2103067	D3TI2C
Fatwah Firmansyah SR	Teknik Informatika
85	80
90	90

Simpan

Data Mahasiswa

Data Kosong

Maka data yang kita buat tersebut akan langsung ditampilkan ketika kita berhasil melakukan insert data tersebut ke database.

8:29



Parsing Json

Form Input

Nim	kelas
Nama Mhs	Jurusan
Kehadiran	Tugas
UTS	UAS

Simpan

Data Mahasiswa

Nim Mhs: 2103067
Nama Mhs: Fatwah Firmansyah SR
Kelas Mhs: D3TI2C
Jurusan Mhs: Teknik Informatika
Nilai Kehadiran: 85
Nilai Tugas: 80
Nilai UTS: 90
Nilai UAS: 90
Nilai Akhir: 86