

UAS  
MOBILE PROGRAMMING  
Ade Putra Prima Suhendri, S.Kom, M.Kom

SOAL

1. Jelaskan apa yang dimaksud dengan Mobile Programming? Point 5
2. Jelaskan apa yang dimaksud dengan User Interface (UI)? Point 5
3. Jelaskan apa yang dimaksud dengan API? jelaskan fungsinya! Point 5
4. Jelaskan perbedaan Native dan Hybrid pada mobile programming? Point 5
6. Jelaskan apa fungsi github! Point 5
7. Apa output dari script berikut ! Point 10:

```
ListView.builder(  
  itemCount: 10,  
  itemBuilder: (context, i){  
    return Text("$i");  
  },  
);
```

8. Apa output dari script berikut ! Point 10:

```
int timesTwo(int x) {  
  return x * 2;  
}  
  
int timesFour(int x) => timesTwo(timesTwo(x));  
  
int runTwice(int x, int Function(int) f) {  
  for (var i = 0; i < 2; i++) {  
    x = f(x);  
  }  
  return x;  
}  
  
void main() {  
  print("4 times two is ${timesTwo(4)}");  
  print("4 times four is ${timesFour(4)}");  
  print("2 x 2 x 2 is ${runTwice(2, timesTwo)}");  
}
```

9. Tuliskan sintak cara parsing JSON pada flutter ! Poin 55

Nama : Rizky Surya Putra

NIM : 181011401096

Kelas : 06TPLE011

## Bukti Pembayaran



**YAYASAN SASMITA JAYA**  
**UNIVERSITAS PAMULANG**  
**DATA PEMBAYARAN SEMESTER GENAP 2020/2021**  
FAKULTAS / PRODI : TEKNIK / TEKNIK INFORMATIKA  
NAMA MAHASISWA : RIZKY SURYA PUTRA  
NIM : 181011401096  
SHIFT : REGULER C

### DATA PEMBAYARAN TAGIHAN UANG KULIAH

NO	NOMOR TAGIHAN	NO URUT	PEMBAYARAN	JML BAYAR	STATUS BAYAR	TGL BAYAR	CHANNEL	TEMPAT BAYAR
1	2020113285402201	1	Registrasi	350000	LUNAS	2021-02-13 08:39:53.000	13000001	Bank BTN
2	2020113285402301	2	SKS2	300000	LUNAS	2021-04-04 08:59:56.000	13000001	Bank BTN
3	2020113285402401	3	SKS3	300000	LUNAS	2021-04-04 09:03:35.000	13000001	Bank BTN
4	2020113285400501	4	UTS	350000	LUNAS	2021-04-22 00:00:00.000	NEWMOBILE	Bank BNI
5	2020113285402501	5	SKS4	300000	LUNAS	2021-04-22 00:00:00.000	ATM	Bank BNI
6	2020113285402601	6	SKS5	300000	LUNAS	2021-05-31 00:00:00.000	NEWMOBILE	Bank BNI
7	2020113285402701	7	SKS6	300000	BELUM LUNAS			
8	2020113285400401	8	PRAKTEK	150000	BELUM LUNAS			
9	2020113285400601	9	UAS	350000	BELUM LUNAS			

### DATA PEMBAYARAN TAGIHAN LAINNYA

NO	NOMOR TAGIHAN	NO URUT	PEMBAYARAN	JML BAYAR	STATUS BAYAR	TGL BAYAR	CHANNEL	TEMPAT BAYAR
----	---------------	---------	------------	-----------	--------------	-----------	---------	--------------

1. Mobile Programming adalah pemrograman yang ditujukan untuk pembuatan aplikasi di perangkat mobile. Mobile Programming juga pemrograman yang diterapkan di Mobile atau Laptop.
2. User Interface adalah tampilan visual sebuah produk yang menjembatani sistem dengan pengguna (user). Tampilan UI dapat berupa bentuk, warna, dan tulisan yang didesain semenarik mungkin. Secara sederhana, UI adalah bagaimana tampilan sebuah produk dilihat oleh pengguna.
3. Application Programming Interface atau API adalah seperangkat antarmuka yang dapat berupa fungsi, method, maupun URL endpoint yang dapat digunakan untuk mengembangkan aplikasi baik dalam satu platform maupun lintas platform, API dapat menerima permintaan dari perangkat lunak lainnya seperti mobile apps / website yang memberikan layanan pertukaran data, dengan gunakan API memungkinkan developer untuk mengintegrasikan dan mengizinkan dua aplikasi yang berbeda saling terhubung secara bersamaan.

#### Fungsi API:

- Memudahkan membangun aplikasi yang fungsional: Dengan menggunakan API akan lebih mudah untuk membuat aplikasi yang fungsional dan kompleks. Tanpa perlu menambahkan data secara manual, aplikasi yang dikembangkan akan memiliki fitur dari aplikasi tujuan.
  - Pengembangan aplikasi menjadi lebih efisien: Dengan adanya API kita tidak perlu untuk melakukan komunikasi langsung dengan aplikasi lain yang ingin dihubungkan. Cukup dengan komunikasi melalui API hal ini sangat membantu, terutama jika anda ingin membangun aplikasi lintas platform (Cross Platform) dengan berbagai layanan sekaligus.
  - Meringankan beban server: Dengan menggunakan API kita tidak perlu menyimpan semua data yang dibutuhkan di server kita sendiri, cukup meminta API untuk mendapatkan data terbaru dari server aplikasi asal. Dengan kondisi seperti ini server tidak akan terbebani dan pada akhirnya mengurangi resiko website tidak dapat diakses karena server yang down.
4. Native di Mobile Programming adalah sebuah aplikasi yang dibangun dengan bahasa pemrograman yang spesifik untuk platform-platform tertentu. Contoh populernya penggunaan bahasa pemrograman Objective-C atau swift untuk platform Ios (apple) adapun platform android yang menggunakan bahasa pemrograman seperti hal-nya (JAVA).

Hybrid di Mobile Programming adalah aplikasi web yang di transformasikan menjadi kode native pada platform seperti ios atau android. Aplikasi hybrid biasanya menggunakan browser untuk mengizinkan aplikasi web mengakses berbagai fitur di device mobile seperti hal-nya Push Notification, Contacts, atau Offline Data Storage. Beberapa tools untuk mengembangkan aplikasi hybrid antara lain yaitu, Phonegap, Rubymotion dan lain lain.

6. Fungsi utama github adalah membantu penyimpanan repository. Namun tak hanya sebatas itu saja, masih ada lebih banyak fungsi dari github untuk mendukung project yang Anda garap. Beberapa fungsi github adalah:

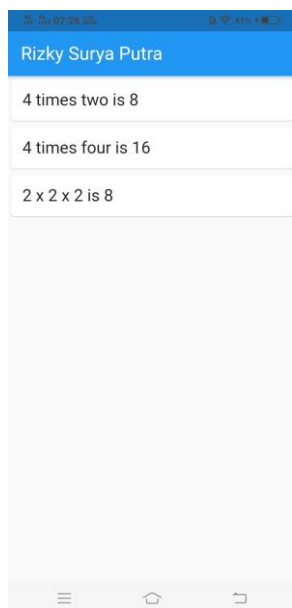
- Memungkinkan Anda untuk berkolaborasi dengan orang lain
- Menyimpan dan mengawasi repository
- Merencanakan, menyimpan dan melacak proses kerja dari proyek
- Berkomunikasi dengan sesama programmer
- Melacak bug dan manajemen tugas. hingga
- Menampilkan profil dan update dari Anda ke khalayak banyak.

7.



The screenshot shows a mobile application interface. At the top, there is a blue header bar with the text "Rizky Surya Putra" in white. Below the header, there is a list of 10 empty, rectangular input fields, each containing a number from 1 to 10. At the bottom of the screen, there is a white navigation bar with three icons: a hamburger menu, a home icon, and a back arrow.

8.



The screenshot shows a mobile application interface. At the top, there is a blue header bar with the text "Rizky Surya Putra" in white. Below the header, there is a list of three input fields containing the following text: "4 times two is 8", "4 times four is 16", and "2 x 2 x 2 is 8". Below these fields, there is a large, empty rectangular area. At the bottom of the screen, there is a white navigation bar with three icons: a hamburger menu, a home icon, and a back arrow.

```
9. [  
  {  
    "id": 1,  
    "name": "Leanne  
Graham", "username":  
    "Bret",  
    "email":  
    "Sincere@april.biz",  
    "address": {  
      "street": "Kulas  
Light", "suite": "Apt.  
556", "city":  
      "Gwenborough",  
      "zipcode": "92998-  
3874", "geo": {  
        "lat": "-37.3159",  
        "lng": "81.1496"  
      }  
    },  
    "phone": "1-770-736-8031 x56442",  
    "website":  
    "hildegard.org",  
    "company": {  
      "name": "Romaguera-Crona",  
      "catchPhrase": "Multi-layered client-server neural-  
net", "bs": "harness real-time e-markets"  
    }  
  },  
  {  
    "id": 2,  
    "name": "Ervin Howell",  
    "username": "Antonette",  
    "email":  
    "Shanna@melissa.tv",  
    "address": {  
      "street": "Victor Plains",
```

```

import 'package:flutter/material.dart';

void main() { runApp(new
MaterialApp(  title: "My
Apps",    home: new
HalamanJson(),
));
}

class HalamanJson extends StatefulWidget {
  @override
  _HalamanJsonState createState() => _HalamanJsonState();

  "suite": "Suite 879",
  "city": "Wisokyburgh",
  "zipcode": "90566-7771",
  "geo": {
    "lat": "-43.9509",
    "lng": "-34.4618"
  }
},
  "phone": "010-692-6593 x09125",
  "website": "anastasia.net",
  "company": {
    "name": "Deckow-Crist",
    "catchPhrase": "Proactive didactic contingency",
    "bs": "synergize scalable supply-chains"
  }
},

```

Tahap Pertama kita akan membuat file halamanJson.dart dalam project kita. Kemudian isi dengan kode di bawah ini.

```

}

class _HalamanJsonState extends State {

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text("Data JSON"),
      ),
      drawer: DrawerApp(),
      body: Center(
        child: Text("Data JSON")
      ),
    );
  }
}

```

Selanjutnya kita membutuhkan beberapa package diantaranya sebagai berikut. Tambahkan http dependencies dalam file pubspecs.yaml

```

dependencies:
  flutter:
    sdk: flutter

  http: ^0.12.0+1

```

Lalu import dalam file .dart

```

import 'dart:convert';
import 'package:http/http.dart' as http;
import 'dart:async';

```

Selanjutnya kita akan menggunakan **Future** untuk menjalankan http.get.

```
List datadariJSON;

Future ambildata() async {
  http.Response hasil = await
  http.get(
    Uri.encodeFull("https://jsonplaceholder.typicode.com/users"),
    headers: {"Accept": "application/json"});

  this.setState(() {
    datadariJSON = json.decode(hasil.body);
  });
}
```

Pada contoh **Future** di atas, sebelumnya kita telah membuat List terlebih dahulu yang bernama **datadariJSON** yang dimaksudkan akan menampung data yang akan diparsing dari url. Jadi sebelum Future **ambildata** dijalankan maka List **datadariJSON** masih bernilai **null**. setState di sini berfungsi untuk merubah state dari **datadariJSON** yang tadinya null menjadi berisi data dari hasil parsing. Lalu untuk menjalankan Future ambil data kita menggunakan **initState**.

```
@override
void initState() {
  this.ambildata();
}
```

Singkatnya Full Code-nya akan menjadi seperti di bawah ini. Di sini saya menampilkan List **datadariJSON** menggunakan **ListView.builder** dimana setiap ListTile nya dapat diubah secara custom sesuai keinginan masingmasing. Untuk melihat contoh Listview custom



```
import 'package:flutter/material.dart';
import 'package:http/http.dart' as
http; import 'dart:async';

void main() { runApp(new
MaterialApp(  title: "My
Apps",  home: new
HalamanJson(),
));
}

class HalamanJson extends StatefulWidget
{ @override
_HalamanJsonState createState() => _HalamanJsonState();
}

class _HalamanJsonState extends
State { List datadariJSON;

Future ambildata() async {
  http.Response hasil = await
  http.get(
    Uri.encodeFull("https://jsonplaceholder.typicode.com/users"),
headers: {"Accept": "application/json"});

  this.setState(() {  datadariJSON
= json.decode(hasil.body);
});
}

@override

import 'dart:convert';
```

```

void initState() {
  this.ambildata();
}

@override
Widget build(BuildContext context) { return
  Scaffold(
    appBar: AppBar(
      title: Text("Data JSON"),
    ),
    body: Container(
      child: ListView.builder(
        itemCount: datadariJSON == null ? 0 : datadariJSON.length,
        itemBuilder: (context, i){
          return ListTile(
            title: Text(datadariJSON[i]['name']),
          );
        }
      ),
    ),
  );
}

```

Saat menampilkan data dari List kita seperti menampilkan array pada umumnya dan disesuaikan dengan struktur dari JSON yang tersedia. Contohnya di sini menampilkan **name** dengan cara **datadariJSON[i]['name']**. Contoh lain jika ingin menampilkan nama jalan maka kita gunakan cara **datadariJSON[i]['address']['street']**.