

UJIAN AKHIR SEMESTER

Disusun untuk memenuhi ujian akhir semester

Pemrograman Mobile 2

Andri Nugraha Ramdhon, S.Kom., M.Kom.



Oleh

Nama : Fariz Hawari
NPM : 18111051
Kelas : TIF-RP-18 CNS A

**PROGRAM STUDI TEKNIK INFORMATIKA
SEKOLAH TINGGI TEKNOLOGI BANDUNG
2021**

Soal

1. Apa yang dimaksud dengan Stateless Widget dan Statefull Widget? Jelaskan dan berikan contoh penggunaannya. (Nilai 10)
2. Jelaskan dan paparkan apa yang dimaksud aplikasi ber-arsitektur Client – Server. (Nilai 20)
3. Sebutkan 5 buah widget yang kalian sukai pada SDK Flutter paparkan fungsi penggunaannya serta cara penulisannya. (Nilai 20)
4. Deskripsikan latar belakang, gambaran umum sistem, fitur serta hasil yang dicapai dari aplikasi tugas besar yang sudah dibuat. (Nilai 25)
5. Gambarkan dan jelaskan arsitektur pertukaran data Client Server pada Aplikasi Tugas Besar yang sudah dibuat. (Nilai 25)

Catatan :

Silahkan gunakan gambar, diagram maupun screenshot untuk membantu menjawab soal di atas.

Jawab

1. Stateless Widget dan Statefull Widget
 - a. Stateless Widget adalah widget statis dimana seluruh konfigurasi yang dimuat didalamnya telah diinisiasi sejak awal.

Contoh :

```
1  import 'package:flutter/material.dart';
2
3  void main() {
4    runApp(MyApp());
5  }
6
7  class MyApp extends StatelessWidget {
8    @override
9    Widget build(BuildContext context) {
10     return MaterialApp(
11       home: Scaffold(
12         appBar: AppBar(
13           title: Text('BelajarFlutter.com'),
14         ),
15         body: Center(
16           child: Text(
17             'Buku yang dibaca : 1',
18             style: TextStyle( fontSize: 40 ),
19             textAlign: TextAlign.center,
20           ),
21         ),
22       ),
23     );
24   }
25 }
```

Gambar diatas adalah contoh dari Stateless Widget, yaitu tampilannya yang sederhana atau tidak dapat dirubah, jadi hanya terdapat tampilannya saja tanpa fungsional yang lain.

- b. Sedangkan Stateful widget berlaku sebaliknya dimana sifatnya adalah dinamis, sehingga widget ini dapat diperbaharui kapanpun dibutuhkan berdasarkan user actions atau ketika terjadinya perubahan data

Contoh :

```
class MyHomePage extends StatefulWidget {
  MyHomePage({Key key, this.title}) : super(key: key);

  final String title;

  @override
  _MyHomePageState createState() => _MyHomePageState();
}

class _MyHomePageState extends State<MyHomePage> {
  int _counter = 0;

  void _incrementCounter() {
    setState(() {
      _counter++;
    });
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text(widget.title),
      ), // AppBar
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
          children: <Widget>[
            Text(
              'You have pushed the button this many times:',
            ), // Text
            Text(
              '$_counter',
              style: Theme.of(context).textTheme.headline4,
            ), // Text
          ], // <Widget>[]
        ), // Column
      ), // Center
      floatingActionButton: FloatingActionButton(
        onPressed: _incrementCounter,
        tooltip: 'Increment',
        child: Icon(Icons.add),
      ), // FloatingActionButton
    ); // Scaffold
  }
}
```

Sedangkan gambar diatas adalah contoh dari Statefull Widget, yaitu tampilannya terdapat beberapa fungsi yang digunakan.

2. Client-Server adalah arsitektur jaringan yang memisahkan client(biasanya aplikasi yang menggunakan GUI) dengan server. Masing-masing client dapat meminta data atau informasi dari server. Sistem client server didefinisikan sebagai sistem terdistribusi, tetapi ada beberapa perbedaan karakteristik yaitu :
 - a. Servis (layanan)
 - Hubungan antara proses yang berjalan pada mesin yang berbeda.
 - Pemisahan fungsi berdasarkan ide layanannya.
 - Server sebagai provider, client sebagai konsumen.
 - b. Sharing resources (sumber daya), Server bisa melayani beberapa client pada waktu yang sama, dan meregulasi akses bersama untuk share sumber daya dalam menjamin konsistensinya.
 - c. Asymmetrical protocol (protokol yang tidak simetris), Many-to-one relationship antara client dan server.Client selalu menginisiasikan dialog melalui layanan permintaan, dan server menunggu secara pasif request dari client.
 - d. Transparansi lokasi, Proses yang dilakukan server boleh terletak pada mesin yang sama atau pada mesin yang berbeda melalui jaringan.Lokasi server harus mudah diakses dari client.
 - e. Mix-and-Match, Perbedaan server client platforms
 - f. Pesan berbasiskan komunikasi, Interaksi server dan client melalui pengiriman pesan yang menyertakan permintaan dan jawaban.
 - g. Pemisahan interface dan implementasi, Server bisa diupgrade tanpa mempengaruhi client selama interface pesan yang diterbitkan tidak berubah.

3. Macam – macam widget pada flutter :

a. Scaffold

Scaffold adalah widget utama untuk membuat sebuah halaman pada flutter.

```
body: Row(  
  mainAxisAlignment: MainAxisAlignment.spaceBetween,  
  children: <Widget>[  
    Container(  
      child: Text("Halo 1 !!!"),  
      color: Colors.lime,  
      padding: EdgeInsets.all(16.0),  
    ),  
  ],  
)
```

b. Row

Row akan menampilkan widget-widget secara horizontal atau sebaris dari kiri ke kanan.

```
body: Row(  
  mainAxisAlignment: MainAxisAlignment.spaceBetween,  
  children: <Widget>[  
    Container(  
      child: Text("Halo 1 !!!"),  
      color: Colors.lime,  
      padding: EdgeInsets.all(16.0),  
    ),  
  ],  
)
```

c. Column

Column berlaku sebaliknya widget akan mengarah secara vertikal atau dari atas ke bawah.

```
body: Column(  
  children: <Widget>[  
    Container(  
      child: Text("Halo 1 !!!"),  
      color: Colors.lime,  
      padding: EdgeInsets.all(16.0),  
    ),  
  ],  
)
```

d. Container

Container merupakan widget yang fungsinya untuk membungkus widget lain sehingga dapat diberikan nilai seperti margin, padding, warna background, atau dekorasi.

```
body: Container(  
  padding: EdgeInsets.all(32.0),  
  margin: EdgeInsets.fromLTRB(20.0, 10.0, 20.0, 0),  
  decoration: BoxDecoration(  
    borderRadius: BorderRadius.circular(20.0),  
    color: Colors.purple,  
    // color: Colors.purple,  
    child: Text('Haiii', style: TextStyle(color: Colors.white,  
      fontSize: 20.0),)  
  )  
)
```

e. ListView

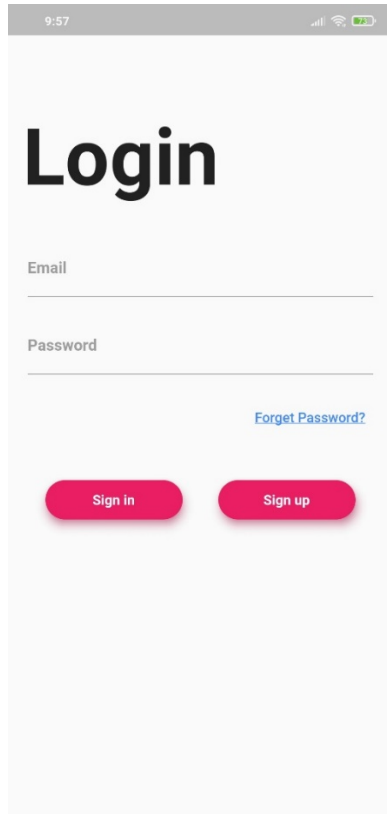
ListView digunakan untuk menampilkan daftar item dalam jumlah yang banyak.

```
body: ListView(  
  children: <Widget>[  
    Padding(  
      padding: EdgeInsets.all(8.0),  
      child: ListTile(  
        leading: Icon(Icons.access_alarm),  
        title: Text('Alarm'),  
      ),  
    ),  
  ],  
)
```

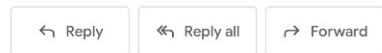
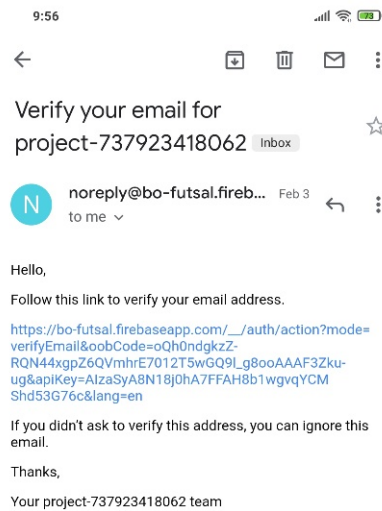
4. Latar belakang pembuatan aplikasi fullen app yaitu masih banyaknya tempat futsal yang masih menginputkan datanya secara manual, disini kami memiliki ide untuk mempermudah masalah penginputan data yang asalnya secara manual, menjadi digital, serta mempermudah dalam urusan *booking* atau memesan lapangan agar para pemain tidak harus datang dulu ke tempat untuk memastikan apakah tersedia atau tidak lapangan yang ingin di sewa. Untuk gambaran aplikasinya, aplikasi yang kami buat baru sampai pada fungsi login dan register, namun fungsi pada pembookingan atau pemesanannya belum terjadi. Berikut lampiran dari fitur – fitur dari aplikasi yang kami buat :

- a. Splash Screen
- b. Login dan Register email (termasuk kedalam autentikasi)
- c. Dashboard
- d. Penentuan jadwal booking lapangan

5. Untuk pertukaran data Client Server pada aplikasi yang kami buat, baru hanya sebatas email, yaitu fungsi sign in, sign up, dan reset password, berikut saya lampirkan gambaran dari proses pertukaran data Client Servernya :



Pada tampilan login ini, terdapat fungsi sign up, dimana kita mendaftarkan email untuk dapat masuk ke dashboard aplikasi, apabila sudah melakukan sign up, maka akan ada email autentikasi untuk memastikan bahwa email yang didaftarkan aktif, contohnya sebagai berikut :



Kita tinggal klik saja link yang ada di email masuk dari database aplikasi, lalu email sudah bisa digunakan untuk login ke aplikasi. Apabila kita lupa password, kita dapat mereset password dari email yang kita daftarkan, kita tinggal memasukkan email yang sudah didaftarkan, maka akan ada email verifikasi seperti berikut :

