



Modul Praktikum **Pemrograman Mobile**



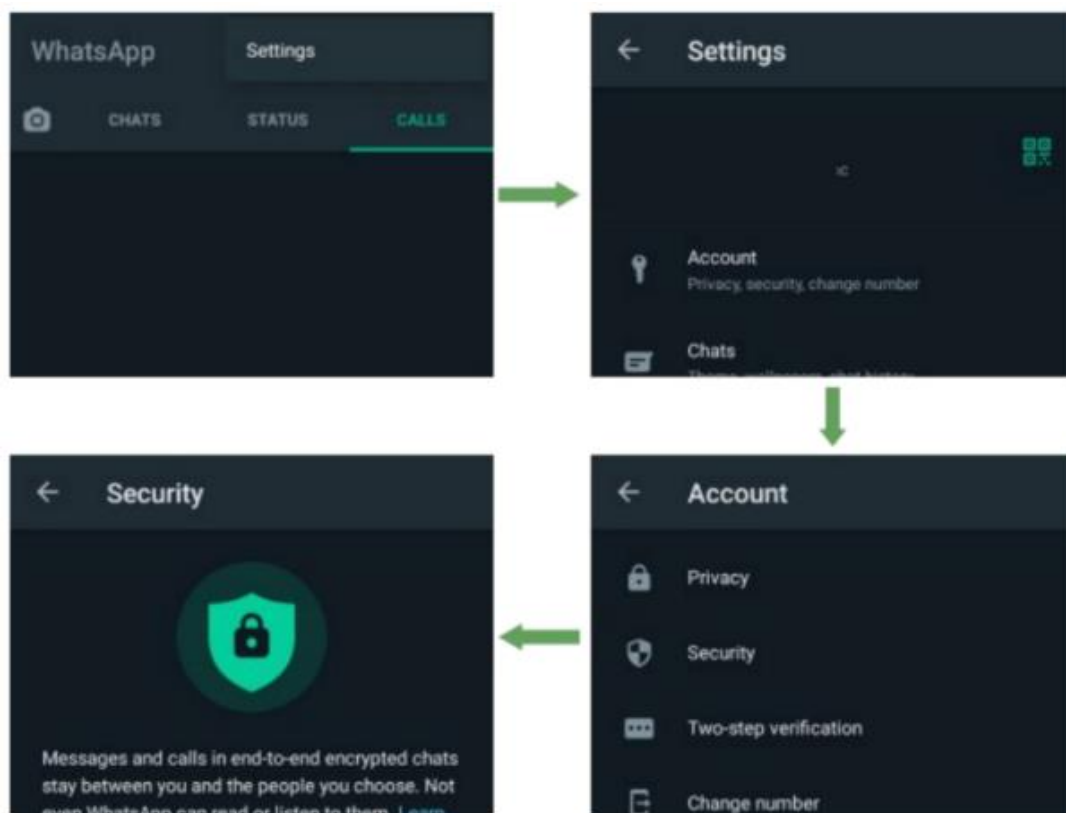
NAVIGATION

TUJUAN PEMBELAJARAN

- A. Mahasiswa memahami penggunaan Navigation
- B. Mahasiswa memahami penggunaan Route
- C. Mahasiswa memahami penggunaan AppBar
- D. Mahasiswa memahami penggunaan Drawer
- E. Mahasiswa memahami penggunaan Bar Navigation

DASAR TEORI

Navigation adalah cara kita masuk ke suatu halaman dalam aplikasi mobile. Di dalam Flutter, Navigation bekerja seperti **Stack**. Jadi halaman yang dituju akan bertumpuk diatas halaman yang dibuka sebelumnya. Contohnya adalah aplikasi Whatsapp. Kalian bisa coba buka **Settings** → **Account** → **Security**.





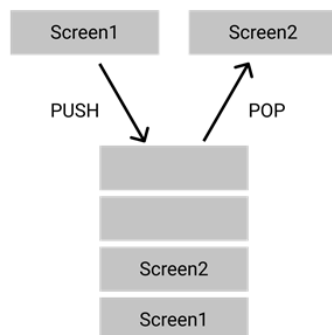
Seperti yang sudah dijelaskan, navigation pada WhatsApp terlihat seperti *stack*(tumpukan).



PRAKTEK

I. Navigation & Route

Kita telah bisa membuat satu tampilan screen (layar/page) pada pembelajaran sebelumnya. Namun, pada saat membangun sebuah aplikasi kita akan membuat banyak sekali screen dan kita akan berpindah dari satu screen ke screen lainnya. Pada Flutter kita akan menggunakan sebuah class bernama **Navigator**. Dengan **Navigator** ini kita akan berpindah dari satu screen ke screen lainnya.



Perlu kita ketahui bahwa konsep *navigasi* pada Flutter mirip sekali dengan pemrograman Android, yakni bahwa ketika berpindah *screen/activity* akan menjadi tumpukan (stack). Jadi ketika berpindah dari satu screen ke screen lain (push), maka screen pertama akan ditumpuk oleh screen kedua. Kemudian apabila kembali dari screen kedua ke pertama, maka screen kedua akan dihapus (pop).

Berikut adalah contoh penerapan navigator pada flutter. Isi kodingan pada halaman pertama sebagai berikut :

```
class FirstScreen extends StatelessWidget {  
  const FirstScreen({Key? key}) : super(key: key);  
  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: const Text('First Screen'),  
      ),  
    ),  
  },  
);
```



```
body: Center(  
  child: ElevatedButton(  
    child: const Text('Pindah Screen'),  
    onPressed: () {  
      Navigator.push(  
        context,  
        MaterialPageRoute(  
          builder: (context) {  
            return const SecondScreen();  
          },  
        ),  
      );  
    },  
  ),  
),  
);  
}
```

Lalu isi kodingan berikut pada page/halaman kedua

```
class SecondScreen extends StatelessWidget {  
  const SecondScreen({Key? key}) : super(key: key);  
  
  @override  
  Widget build(BuildContext context) {  
    return Scaffold(  
      appBar: AppBar(  
        title: const Text('Second Screen'),  
      ),  
      body: Center(  
        child: OutlinedButton(  
          child: const Text('Kembali'),  
          onPressed: () {  
            Navigator.pop(context);  
          },  
        ),  
      ),  
    );  
  }  
}
```

Di dalam Flutter, screen atau halaman juga dikenal dengan istilah routes. Konsep named route ini mirip dengan sebuah website di mana memiliki route atau endpoint untuk



merujuk ke suatu halaman, contohnya seperti /login atau /detail. Salah satu penerapan route pada flutter yaitu menggunakan **named route**, berikut adalah cara penerapannya.

Pertama kita tambahkan penamaan route di MaterialApp

```
routes: {  
  '/halaman1': (context) => const FirstScreen(),  
  '/halaman2': (context) => const SecondScreen(),  
},
```

Setelah itu di widget ElevatedButton tadi bagian onPressed kita ubah navigatornya.

```
onPressed: () {  
  Navigator.pushNamed(context, '/halaman2');  
},
```

2. AppBar

Seharusnya kalian sudah sering melihat AppBar. AppBar yang akan sering kita gunakan adalah Scaffold. Scaffold memungkinkan kita untuk membangun aplikasi dengan UI yang umum pada Android. Berikut beberapa widget pada AppBar yang dapat digunakan

- Title

Title adalah widget yang digunakan untuk memberikan text judul pada page. Berikut adalah contoh nya.

```
appBar: AppBar(  
  title: const Text(  
    'PRAK MOBILE',  
    style: TextStyle(  
      fontSize: 20,  
      fontWeight: FontWeight.bold,  
    ),  
  ),  
)
```

- Actions

Actions adalah list widget yang bisa di masukkan pada posisi kanan setelah Title. Ini adalah contoh kodingan actions.

```
appBar: AppBar(  
  title: const Text(  
    'PRAK MOBILE',  
    style: TextStyle(  
      fontSize: 20,  
      fontWeight: FontWeight.bold,  
    ),  
  ),  
  actions: [  
    IconButton(  
      icon: const Icon(Icons.add),  
      onPressed: () {  
        // Add action  
      },  
    ),  
  ],  
)
```



```
actions: [  
  IconButton(  
    onPressed: () {},  
    icon: const Icon(  
      Icons.person_outline,  
      size: 30,  
    ),  
  ),  
],  
)
```

- **Leading**

Leading adalah Widget yang ada di samping kiri Title. Berikut adalah contoh kodingan.

```
appBar: AppBar(  
  leading: IconButton(  
    icon: const Icon(  
      Icons.menu,  
      size: 30,  
    ),  
    onPressed: () {},  
  ),  
)
```

3. Drawer

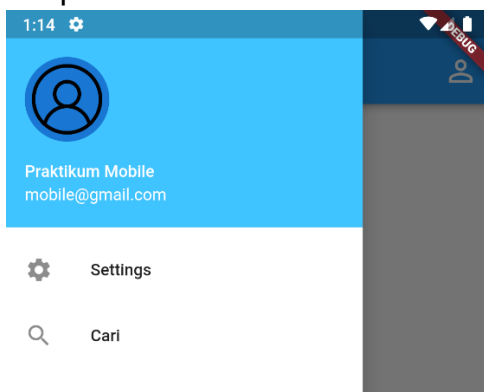
Drawer adalah navigasi yang biasanya ada di samping kiri dan biasanya dapat dibuka dengan cara menekan icon burger bar. Untuk membuat sebuah drawer, kalian bisa membuatnya di dalam Scaffold. Berikut adalah contoh penerapan drawer pada flutter.

```
return Scaffold(  
  drawer: Drawer(  
    backgroundColor: Colors.white,  
    child: Column(  
      children: [  
        const UserAccountsDrawerHeader(  
          currentAccountPicture: CircleAvatar(  
            backgroundImage:  
              AssetImage('assets/profile.png'),  
          ),  
          accountName: Text("Praktikum Mobile"),  
          accountEmail: Text('mobile@gmail.com'),  
        ),  
      ],  
    ),  
  ),  
)
```



```
        decoration: BoxDecoration(color:
Colors.lightBlueAccent),
      ),
      ListTile(
        leading: const Icon(Icons.settings),
        title: const Text('Settings'),
        onTap: () {},
      ),
      ListTile(
        leading: const Icon(Icons.search),
        title: const Text('Cari'),
        onTap: () {},
      ),
    ],
  ),
),
);
```

Output:



4. Navigation Bar

Navigation bar hampir sama juga artinya dengan navigator yaitu berpindah dari halaman satu ke halaman lainnya, tetapi pada navigation bar page hanya berpindah dengan cara di geser atau navigasi yang digunakan masih berada di satu halaman. Ada beberapa Navigation bar pada flutter yang dapat di gunakan di antaranya yaitu **TabBar** dan **BottomNavigationBar**

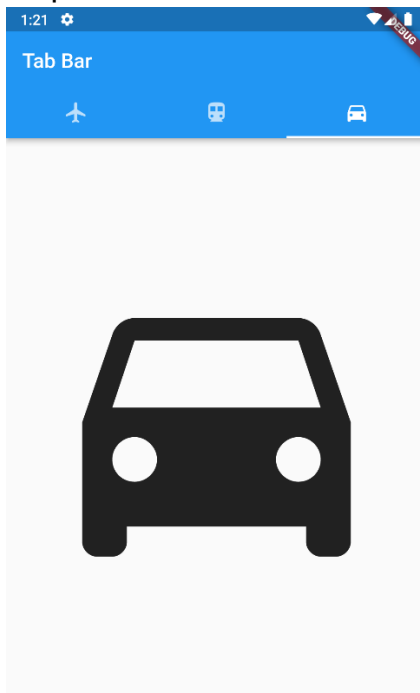
- Tab Bar

Navigation Tab Bar ini secara tampilan terletak di atas. Berikut adalah contoh penerapannya.



```
return DefaultTabController(  
  length: 3,  
  child: Scaffold(  
    appBar: AppBar(  
      title: const Text('Tab Bar'),  
      bottom: const TabBar(  
        tabs: [  
          Tab(icon: Icon(Icons.flight)),  
          Tab(icon: Icon(Icons.directions_transit)),  
          Tab(icon: Icon(Icons.directions_car)),  
        ],  
      ),  
    ),  
    body: const TabBarView(  
      children: [  
        Icon(Icons.flight, size: 350),  
        Icon(Icons.directions_transit, size: 350),  
        Icon(Icons.directions_car, size: 350),  
      ],  
    ),  
  ),  
);
```

Output:





- **BottomNavigationBar**

Seperti dengan namanya, **BottomNavigationBar** ini secara tampilan terletak dibawah. Berikut adalah cara penerapannya. Pertama kita menggunakan `statefulwidget` setelah itu buatlah widget **bottomNavigationBar** didalam scaffold

```
bottomNavigationBar: BottomNavigationBar(  
  type: BottomNavigationBarType.fixed,  
  backgroundColor: Colors.blueAccent,  
  unselectedItemColor: Colors.white,  
  selectedItemColor: Colors.black,  
  currentIndex: _index,  
  onTap: _onItemTap,  
  items: const [  
    BottomNavigationBarItem(  
      icon: Icon(Icons.home),  
      label: "Home",  
    ),  
    BottomNavigationBarItem(  
      icon: Icon(Icons.chat_bubble),  
      label: "Chat",  
    ),  
    BottomNavigationBarItem(  
      icon: Icon(Icons.call),  
      label: "Telpon",  
    ),  
    BottomNavigationBarItem(  
      icon: Icon(Icons.camera_alt_outlined),  
      label: "Camera",  
    ),  
  ],  
)
```

Setelah itu kita buat variable untuk index bottomNavigationBar dan list page tiap bar

```
int _index = 0;  
  
void _onItemTap(int index) {  
  setState(() {  
    _index = index;  
  });  
}  
  
static List<Widget> _pages = [  
  Icon(  

```



```
Icons.home,  
size: 100,  
),  
Icon(  
Icons.chat_bubble,  
size: 100,  
),  
Icon(  
Icons.call,  
size: 100,  
),  
Icon(  
Icons.camera,  
size: 100,  
),  
];
```

Lalu di dalam scaffold pada bagian body kita tambahkan list page yang kita buat tadi.

```
body: Center(  
  child: _pages.elementAt(_index),  
),
```

Output :

