

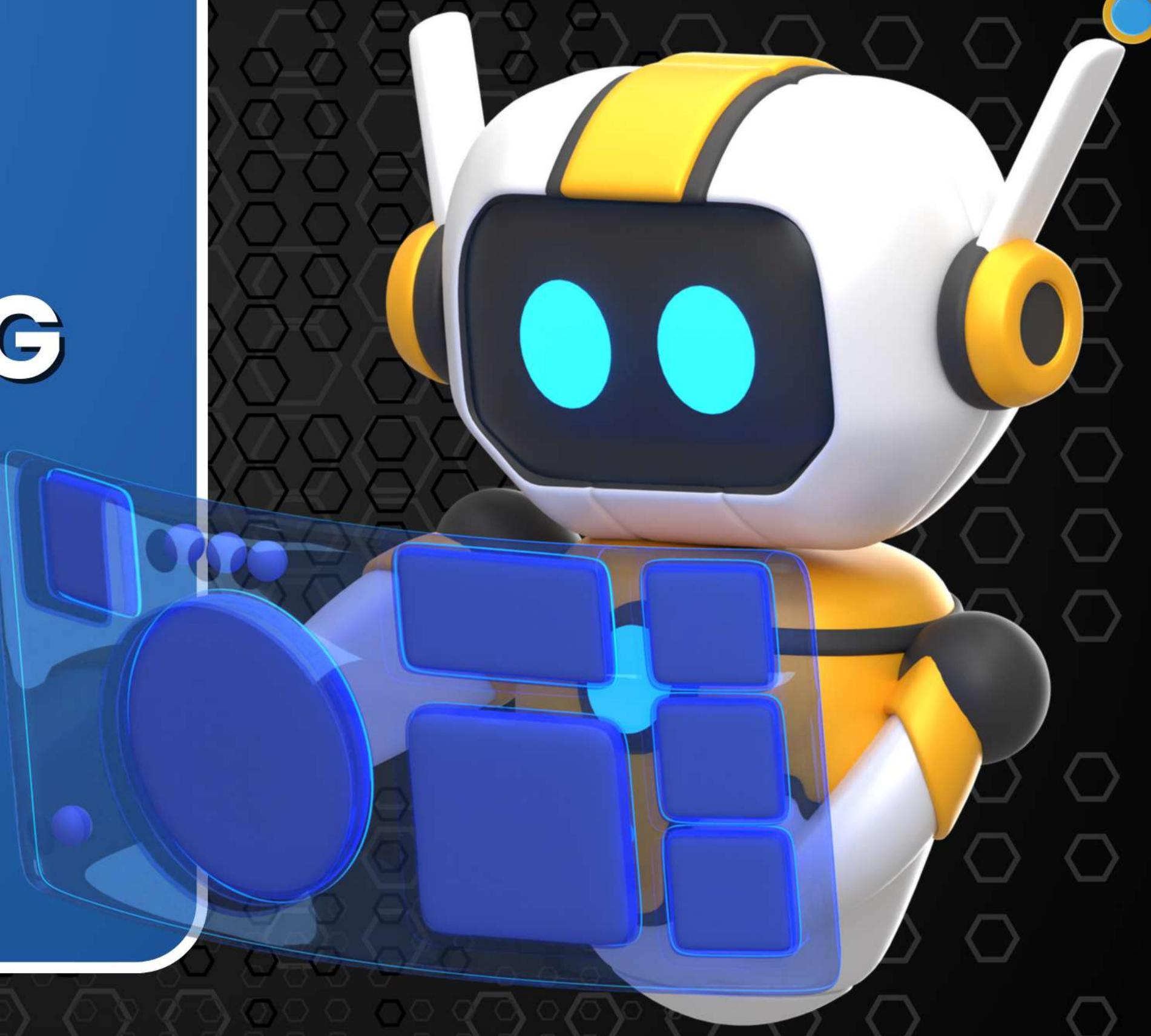


Universitas Pamulang
Sistem Informasi

MOBILE PROGRAMMING

PERTEMUAN 11

Yudisti Prayigo Permana, S.Kom., M.Kom.





TOPIK PEMBAHASAN

- Pengenalan SearchView
- Metode SearchView
- Latihan
- Form Login
- Register Form
- Tugas

01
02
03
04
05
06





Apa itu SearchView?

SearchView adalah komponen UI yang memungkinkan pengguna melakukan pencarian data dalam aplikasi. Di Flutter, SearchView tidak ada sebagai widget bawaan secara eksplisit, tetapi dapat diimplementasikan menggunakan kombinasi beberapa widget seperti TextField, AppBar, dan manajemen state.





Konsep Dasar SearchView

- SearchView memiliki karakteristik utama:
- Input Text: Area tempat pengguna mengetik query pencarian
- Real-time Search: Hasil langsung terupdate saat pengguna mengetik
- Filtering: Memfilter data berdasarkan input pengguna
- Clear Button: Opsi untuk menghapus input pencarian



Komponen SearchView

Berikut widget-widget kunci untuk membuat SearchView:

- Widget : Fungsi
- TextField : Input teks untuk query pencarian
- AppBar : Tempat menampilkan SearchView (biasanya di bagian atas layar)
- ListView.builder : Menampilkan hasil pencarian secara efisien
- TextEditingController : Mengontrol dan memantau perubahan input teks
- Provider (opsional) : State management untuk hasil pencarian



Cara Kerja SearchView

1. Pengguna mengetik di TextField
2. TextEditingController mendeteksi perubahan
3. Data difilter menggunakan metode .where()
4. setState() memperbarui tampilan



Jenis Metode Pencarian

- Basic Contains (Pencarian Fleksibel) TextEditingController
Contoh: Query "ap" akan menemukan "Apple", "Grape", "Pineapple"
- StartsWith (Pencarian Awalan)
Contoh: Query "ap" hanya menemukan "Apple"
- Fuzzy Search (Pencarian Tidak Ketat)
Contoh: Query "ale" bisa menemukan "apple" (a-p-p-l-e)



Android vs Flutter

Fitur	Flutter	Android Native
Komponen	TextField + Custom Logic	SearchView Widget
Integrasi AppBar	Manual	Bawaan ActionBar
Animasi	Custom Implementation	Bawaan Material Design
Flexibility	Sangat Fleksibel	Terbatas pada Android API



Metode SearchView

1. Metode Dasar (TextField di AppBar)

Implementasi paling sederhana dengan menggunakan TextField langsung di AppBar.

Kelebihan:

- Implementasi cepat
- Tidak memerlukan package tambahan

Kekurangan:

- Kurang fitur advanced (riwayat, animasi khusus)



Metode SearchView

2. Metode SearchDelegate (Bawaan Flutter)

Menggunakan kelas SearchDelegate yang disediakan Flutter untuk pengalaman pencarian native.

Kelebihan:

- Tampilan konsisten dengan platform
- Mendukung animasi transisi otomatis
- Fitur clear query bawaan

Kekurangan:

- Kurang fleksibel untuk kustomisasi UI



Metode SearchView

3. Metode Package Third-Party

Menggunakan package seperti flutter_search_bar atau searchable_listview.

Kelebihan:

- Implementasi sangat cepat
- Banyak fitur tambahan (highlight, pagination)

Kekurangan:

- Ketergantungan pada package eksternal



Metode SearchView

4. Metode Advanced (Custom dengan Provider/Riverpod)

Implementasi dengan state management untuk pencarian kompleks.

Kelebihan:

- Pemisahan logika bisnis
- Mudah dikembangkan (debounce, API call)
- Reusable di multiple screens

Kekurangan:

- Lebih kompleks untuk pemula



Metode SearchView

5. Metode Pencarian Lanjutan

- Fuzzy Search
- Pencarian dengan Multiple Kriteria
- Pencarian dengan Highlight Hasil

Perbandingan Metode

Metode	Kompleksitas	Fleksibilitas	Cocok Untuk
TextField Dasar	Rendah	Terbatas	Aplikasi sederhana
SearchDelegate	Sedang	Sedang	Aplikasi dengan gaya material
Third-Party	Rendah	Tinggi	Prototyping cepat
Custom Provider	Tinggi	Sangat Tinggi	Aplikasi kompleks



main.dart

```
● ● ●  
  
import 'package:flutter/material.dart';  
  
void main() {  
  runApp(const MyApp());  
}  
  
class MyApp extends StatelessWidget {  
  const MyApp({super.key});  
  
  @override  
  Widget build(BuildContext context) {  
    return const MaterialApp(  
      debugShowCheckedModeBanner: false,  
      home: TabLayoutPage(),  
    );  
  }  
}
```

SOURCE CODE

```
● ● ●  
  
class TabLayoutPage extends StatefulWidget {  
  const TabLayoutPage({super.key});  
  
  @override  
  State<TabLayoutPage> createState() => _TabLayoutState();  
}  
  
class _TabLayoutState extends State<TabLayoutPage> {  
  @override  
  Widget build(BuildContext context) {  
    return const SimpleAppBar();  
  }  
}  
  
class SimpleAppBar extends StatelessWidget {  
  const SimpleAppBar({super.key});  
  
  @override  
  Widget build(BuildContext context) {  
    PageController controller = PageController(  
      initialPage: 0,  
      viewportFraction: 0.8,  
    );  
  }  
}
```





SOURCE CODE

```
● ● ●  
  
List<Color> colors = [Colors.red, Colors.blue, Colors.green];  
  
return DefaultTabController(  
    length: 3,  
    child: Scaffold(  
        appBar: AppBar(  
            leading: PopupMenuButton<String>(  
                icon: const Icon(Icons.menu),  
                onSelected: (value) {  
                    if (value == 'Profil') {  
                        ScaffoldMessenger.of(context).showSnackBar(  
                            const SnackBar(content: Text('Buka Profil')),  
                        );  
                    } else if (value == 'Pengaturan') {  
                        ScaffoldMessenger.of(context).showSnackBar(  
                            const SnackBar(content: Text('Buka Pengaturan')),  
                        );  
                    } else if (value == 'Keluar') {  
                        ScaffoldMessenger.of(context).showSnackBar(  
                            const SnackBar(content: Text('Keluar dari Aplikasi')),  
                        );  
                    }  
                },  
            ),  
        ),  
    ),  
);
```





SOURCE CODE

```
itemBuilder: (BuildContext context) => [
  const PopupMenuItem(
    value: 'Profil',
    child: Text('Profil'),
  ),
  const PopupMenuItem(
    value: 'Pengaturan',
    child: Text('Pengaturan'),
  ),
  const PopupMenuItem(
    value: 'Keluar',
    child: Text('Keluar'),
  ),
],
),
),
title: const Text("Pertemuan 11"),
centerTitle: true,
actions: [
  IconButton(
    icon: const Icon(Icons.search),
    onPressed: () {
      showSearch(
        context: context,
        delegate: ElectronicSearchDelegate(),
      );
    },
  ),
],
```

```
backgroundColor: Colors.blue,
flexibleSpace: Container(
  decoration: const BoxDecoration(
    gradient: LinearGradient(
      colors: [Colors.blue, Colors.lightBlue],
      begin: Alignment.bottomRight,
      end: Alignment.topLeft,
    ),
  ),
),
bottom: const TabBar(
  indicatorColor: Colors.white,
  tabs: [
    Tab(icon: Icon(Icons.apps), text: "Menu"),
    Tab(icon: Icon(Icons.image), text: "Image"),
    Tab(icon: Icon(Icons.info), text: "Info"),
  ],
),
```



SOURCE CODE

```
body: TabBarView(  
    children: [  
        Center(  
            child: PageView.builder(  
                controller: controller,  
                itemCount: colors.length,  
                itemBuilder: (context, index) {  
                    return Container(  
                        color: colors[index],  
                        width: 200,  
                        height: 200,  
                        margin: const  
                            EdgeInsets.all(10));  
                },  
            ),  
            const Center(child: Text("Image")),  
            const Center(child: Text("Info")),  
        ],  
    ),  
);  
}
```

```
class ElectronicSearchDelegate extends SearchDelegate {  
    List<String> electronics = [  
        'Televisi',  
        'Kulkas',  
        'Mesin Cuci',  
        'Laptop',  
        'Smartphone',  
        'Tablet',  
        'Headphone',  
        'Speaker Bluetooth',  
        'Kamera',  
        'Printer',  
        'AC',  
        'Setrika',  
        'Microwave',  
        'Proyektor',  
    ];  
  
    @override  
    List<Widget>? buildActions(BuildContext context) {  
        return [  
            IconButton(  
                icon: const Icon(Icons.clear),  
                onPressed: () {  
                    query = '';  
                },  
            ),  
        ];  
    }
```

```
@override  
Widget? buildLeading(BuildContext context) {  
    return IconButton(  
        icon: const Icon(Icons.arrow_back),  
        onPressed: () => close(context, null),  
    );  
  
    @override  
    Widget buildResults(BuildContext context) {  
        List<String> matchQuery = electronics  
            .where((item) => item.toLowerCase().contains(query.toLowerCase()))  
            .toList();  
  
        return ListView.builder(  
            itemCount: matchQuery.length,  
            itemBuilder: (context, index) {  
                return ListTile(  
                    title: Text(matchQuery[index]),  
                    onTap: () {  
                        close(context, null); // tutup search  
                        Navigator.push(  
                            context,  
                            MaterialPageRoute(  
                                builder: (context) =>  
                                    DetailBarangPage(namaBarang: matchQuery[index]),  
                            ),  
                        );  
                    },  
                );  
            },  
        );  
    }  
}
```



SOURCE CODE

```
Widget buildSuggestions(BuildContext context) {
    List<String> matchQuery = electronics
        .where((item) => item.toLowerCase().contains(query.toLowerCase()))
        .toList();

    return ListView.builder(
        itemCount: matchQuery.length,
        itemBuilder: (context, index) {
            return ListTile(
                title: Text(matchQuery[index]),
                onTap: () {
                    close(context, null);
                    Navigator.push(
                        context,
                        MaterialPageRoute(
                            builder: (context) =>
                                DetailBarangPage(namaBarang: matchQuery[index]),
                        ),
                    );
                },
            );
        },
    );
}
```

```
class DetailBarangPage extends StatelessWidget {
    final String namaBarang;

    const DetailBarangPage({super.key, required this.namaBarang});

    @override
    Widget build(BuildContext context) {
        return Scaffold(
            appBar: AppBar(
                title: Text('Detail $namaBarang'),
                backgroundColor: Colors.blue,
            ),
            body: Center(
                child: Card(
                    elevation: 4,
                    margin: const EdgeInsets.all(16),
                    shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(12)),
                    child: Padding(
                        padding: const EdgeInsets.all(24),
                        child: Column(
                            mainAxisAlignment: MainAxisAlignment.min,
                            children: [
                                Icon(Icons.electrical_services, size: 60, color: Colors.blue),
                                const SizedBox(height: 20),
                                Text(
                                    namaBarang,
                                    style: const TextStyle(
                                        fontSize: 24,
                                        fontWeight: FontWeight.bold,
                                    ),
                                ),
                                const SizedBox(height: 12),
                                const Text(
                                    "Ini adalah informasi detail dari barang elektronik yang dipilih.",
                                    textAlign: TextAlign.center,
                                ),
                            ],
                        ),
                    ),
                ),
            );
    }
}
```



HASIL

The figure displays three screenshots of a web application interface, likely a search or information retrieval system, showing results for the query "Televisi".

- Screenshot 1:** A main page titled "Pertemuan 11" with a large red placeholder area. It includes navigation icons for "Menu", "Image", and "Info".
- Screenshot 2:** A search results page titled "Search" with a list of items. The results include "Televisi" at the top, followed by Kulkas, Mesin Cuci, Laptop, Smartphone, Tablet, Headphone, Speaker Bluetooth, Kamera, Printer, AC, Setrika, Microwave, and Dronektor.
- Screenshot 3:** A detailed view page titled "Detail: Televisi". It features a large blue television set icon. Below it, there is a section with a blue video camera icon and the text "Televisi". A subtitle states "Ini adalah informasi detail dari barang elektronik yang dipilih."





TUGAS

Buatlah Form Login dan Register Login

Batas pengumpulan 6 Juni 2025





Installation

Ketikan pada Terminal :

```
$ flutter pub add salomon_bottom_bar
```

```
$ flutter pub add icons_plus
```

```
flutter pub outdated
```

Pubspec.yaml

```
dependencies:  
  salomon_bottom_bar: ^3.3.2  
  icons_plus: ^5.0.0
```

Pada main.dart tambahkan :

```
import 'package:salomon_bottom_bar/salomon_bottom_bar.dart';
```

```
import 'package:icons_plus/icons_plus.dart';
```



Installation

Buat susunan folder dan dua file seperti gambar disamping. Dengan cara :

- Klik kanan folder lib > New Folder > beri nama foldernya dengan ‘pages’, ‘style’, dan ‘widget’
- Klik kanan folder pages > New File > beri nama filenya dengan ‘profile_page.dart’ , ‘welcome_page.dart’ , ‘login_page.dart’ , ‘register_page.dart’ dan ‘forgot_password.dart’.

```
lib
└── pages
    ├── forgotpassword_page.dart
    ├── login_page.dart
    ├── profile_page.dart
    ├── register_page.dart
    └── welcome_page.dart
└── style
    └── color.dart
└── widget
    └── custom_scaffold.dart
└── main.dart
```



Source Code

main.dart

```
import 'package:flutter/material.dart';
import 'package:flutter_pert11_login/pages/profile_page.dart';
import 'package:flutter_pert11_login/pages/welcome_page.dart';
import 'package:salomon_bottom_bar/salomon_bottom_bar.dart';

void main(){
    runApp(MyApp());
}

class MyApp extends StatefulWidget{
    @override
    _MyAppState createState() => _MyAppState();
}

class _MyAppState extends State<MyApp>{
    final List<Widget> _page = [ProfilePage(), WelcomePage()];

    var currentPage = 0;
}
```

```
@override
Widget build(BuildContext context){
    return MaterialApp(
        debugShowCheckedModeBanner: false,
        theme: ThemeData(
            primarySwatch: Colors.blue,
            visualDensity: VisualDensity.adaptivePlatformDensity,
        ),
        home: Scaffold(
            body: _page[currentPage],
            bottomNavigationBar: SalomonBottomBar(
                currentIndex: currentPage,
                onTap: (i) => setState(() => currentPage = i),
                items: [
                    //Profile
                    SalomonBottomBarItem(
                        icon: Icon(Icons.person),
                        title: Text("Profile"),
                        selectedColor: Colors.blue,
                    ),
                    //Login dan Register
                    SalomonBottomBarItem(
                        icon: Icon(Icons.lock),
                        title: Text("Login dan Register"),
                        selectedColor: Colors.blue,
                    ),
                ],
            );
        }
}
```



Source Code





Thank You!

