

**LAPORAN TEORI MOBILE PROGRAMMING**  
**MODUL 7**



Nama : Firman Fadilah Noor  
NIM : 240605110083  
Kelas : B  
Tanggal : 29 September 2025

**JURUSAN TEKNIK INFORMATIKA**  
**FAKULTAS SAINS DAN TEKNOLOGI**  
**UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG**  
**GANJIL 2025/2026**

## I. Tujuan

1. Memahami konsep navigasi antar halaman (routing) pada Flutter.
2. Mengimplementasikan navigasi menggunakan `Navigator.pushNamed()` dan `Navigator.pop()`.
3. Mengimplementasi navigasi menggunakan `Navigator.push(MaterialPageRoute())` dan `Navigator.pop()`.
4. Mendemonstrasikan hasil navigasi antar halaman pada aplikasi Flutter sederhana.

## II. Langkah Kerja

1. Buat proyek baru di Flutter.
2. Buat dua file di folder lib: **Home.dart** dan **tujuan.dart**.
3. Di main.dart, import kedua file dan daftarkan route di MaterialApp:
  - `‘/’` HomePage
  - `‘/tujuan’` TujuanPage
4. Di home.dart, buat tombol dengan `Navigator.pushNamed(context,‘/tujuan’)` untuk menuju halaman tujuan.
5. Di tujuan.dart, buat tombol dengan `Navigator.pop(context)` untuk Kembali ke halaman utama.
6. Jalankan aplikasi dan pastikan navigasi antar halaman berfungsi dengan baik

## III. Screenshot Hasil

### a. Kode Program.

#### Home.dart

```
import 'package:flutter/material.dart';

class HomePage extends StatelessWidget {
  const HomePage({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text(
          "Ini halaman Home",
          style: TextStyle(color: Colors.white),
        ),
        backgroundColor: Colors.blue,
      ),
      backgroundColor: Colors.blue,
      body: SingleChildScrollView(
        child: Padding(
          padding: const EdgeInsets.all(20.0),
          child: Column(
            mainAxisAlignment: MainAxisAlignment.start,
            crossAxisAlignment: CrossAxisAlignment.center,
            children: [
              const Text(
                "Banyak aplikasi memiliki beberapa layar untuk",
                style: TextStyle(
                  fontSize: 16.0,
                  color: Colors.white,
                ),
                textAlign: TextAlign.justify,
              ),
              const SizedBox(height: 20),
            ],
          ),
        ),
      ),
    );
  }
}
```

```
import 'package:flutter/material.dart';
import 'package:image/image.dart' as img;

class TujuanPage extends StatelessWidget {
  const TujuanPage({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text(
          "Ini halaman Tujuan",
          style: TextStyle(color: Colors.white),
        ),
        backgroundColor: Colors.red,
      ),
      backgroundColor: Colors.red,
      body: SingleChildScrollView(
        child: Padding(
          padding: const EdgeInsets.all(20.0),
          child: Column(
            mainAxisAlignment: MainAxisAlignment.start,
            crossAxisAlignment: CrossAxisAlignment.center,
            children: [
              const Text(
                "Ke Halaman Tujuan >",
                style: TextStyle(
                  fontSize: 18.0,
                  fontWeight: FontWeight.bold,
                ),
                textAlign: TextAlign.justify,
              ),
              const SizedBox(height: 20),
              ElevatedButton(
                onPressed: () {
                  Navigator.pushNamed(context, '/');
                },
                child: const Text(
                  "Kembali ke Halaman Utama",
                  style: TextStyle(
                    color: Colors.white,
                  ),
                ),
              ),
            ],
          ),
        ),
      ),
    );
  }
}
```

```

        ),
        child: Column(
          mainAxisAlignment: MainAxisAlignment.start,
          crossAxisAlignment: CrossAxisAlignment.center,
          children: [
            const Text(
              "Ke Halaman Tujuan >",
              style: TextStyle(
                fontSize: 18.0,
                fontWeight: FontWeight.bold,
              ),
              textAlign: TextAlign.justify,
            ),
            const SizedBox(height: 20),
            ElevatedButton(
              onPressed: () {
                Navigator.pushNamed(context, '/');
              },
              child: const Text(
                "Kembali ke Halaman Utama",
                style: TextStyle(
                  color: Colors.white,
                ),
              ),
            ),
          ],
        ),
      ),
    );
  }
}
```

## Tujuan.dart

```
import 'package:flutter/material.dart';

class TujuanPage extends StatelessWidget {
  const TujuanPage({super.key});

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: const Text(
          "Ini Halaman Tujuan",
          style: TextStyle(color: Colors.white),
        ), // Text Text
        backgroundColor: Colors.deepOrange,
      ), // AppBar AppBar
      backgroundColor: Colors.deepOrange,
      body: SingleChildScrollView(
        child: Padding(
          padding: const EdgeInsets.all(20.0),
          child: Column(
            mainAxisAlignment: MainAxisAlignment.start,
            crossAxisAlignment: CrossAxisAlignment.center,
            children: [
              const Text(
                "Untuk berpindah ke halaman baru, gunakan metode W",
                style: TextStyle(
                  fontSize: 16.0,
                  color: Colors.white,
                ), // TextStyle TextStyle
                textAlign: TextAlign.justify,
              ), // Text Text
              const SizedBox(height: 20),
              Image.asset(
                'assets/image/island.png',
                width: 250,
                height: 250,
                fit: BoxFit.contain,
              ), // Image.asset Image.asset
              const SizedBox(height: 20),
              const Text(
                "Untuk menutup halaman ke dua kembali ke halaman pertama, B",
                style: TextStyle(
                  fontSize: 16.0,
                  color: Colors.white,
                ), // TextStyle TextStyle
                textAlign: TextAlign.justify,
              ), // Text Text
              const SizedBox(height: 30),
              ElevatedButton(
                style: ElevatedButton.styleFrom(
                  backgroundColor: Colors.blue,
                  foregroundColor: Colors.white,
                  padding: const EdgeInsets.symmetric(
                    horizontal: 40, vertical: 15), // EdgeInsets.symmetric
                  shape: RoundedRectangleBorder(
                    borderRadius: BorderRadius.circular(10),
                  ), // RoundedRectangleBorder RoundedRectangleBorder
                ),
                onPressed: () {
                  Navigator.pop(context);
                },
                child: const Text(
                  "< Kembali ke home",
                  style: TextStyle(
                    fontSize: 18.0,
                    fontWeight: FontWeight.bold,
                  ), // TextStyle TextStyle
                ), // ElevatedButton ElevatedButton
            ],
          ), // Column Column
        ), // Padding Padding
      ), // SingleChildScrollView SingleChildScrollView
    ); // Scaffold Scaffold
  }
}
```

## Main.dart

```
1 import 'package:flutter/material.dart';
2 import 'home.dart';
3 import 'tujuan.dart';
4
5 void main() {
6   runApp(const MyApp());
7 }
8
9 class MyApp extends StatelessWidget {
10   const MyApp({super.key});
11
12   @override
13   Widget build(BuildContext context) {
14     return MaterialApp(
15       debugShowCheckedModeBanner: false,
16       initialRoute: '/',
17       routes: {
18         '/': (context) => const HomePage(),
19         '/tujuan': (context) => const TujuanPage(),
20       },
21     ); // MaterialApp MaterialApp
22   }
23 }
24
```

### b. Penjelasan program

Pada percobaan pertama, program menggunakan library **LiquidCrystal.h** agar Arduino dapat berkomunikasi dengan modul LCD 16x2. Di bagian awal kode, pin-pin yang digunakan untuk mengirim data ke LCD diinisialisasi — mulai dari pin RS, E, hingga empat pin data utama (D4–D7). Setelah itu, dibuat objek bernama `Lcd` yang menjadi “jembatan” antara

Arduino dan layar LCD. Di dalam fungsi `setup()`, perintah `lcd.begin(16, 2)` digunakan untuk mengatur ukuran layar menjadi 16 kolom dan 2 baris. Kemudian, di dalam `loop()`, LCD akan menampilkan dua kalimat berbeda secara bergantian: *"Praktikum Siskom"* dan *"T. Informatika UIN Malang"*, masing-masing dengan jeda lima detik. Proses ini menunjukkan bahwa komunikasi antara Arduino dan LCD berjalan dengan baik — seolah Arduino sedang belajar "berbicara" lewat teks di layar.

Selanjutnya, pada percobaan kedua, program berfokus pada cara menampilkan angka menggunakan **seven segment display**. Di awal kode, terdapat array dua dimensi `seven_seg_digits` yang berisi pola logika HIGH dan LOW untuk membentuk angka 0 hingga 9. Masing-masing angka memiliki kombinasi segmen yang menyala dan mati, seperti layaknya lampu-lampu kecil yang berpadu membentuk angka digital. Di dalam `setup()`, semua pin yang terhubung ke segmen diatur sebagai output. Fungsi `sevenSegWrite()` berperan untuk menghidupkan segmen sesuai angka yang ingin ditampilkan. Lalu, dalam `loop()`, Arduino menampilkan angka dari 0 sampai 9, lalu mundur lagi dari 9 ke 0, dengan jeda satu detik di tiap angka. Program ini memperlihatkan bagaimana logika digital bisa dikombinasikan dengan pengaturan waktu untuk menghasilkan tampilan angka yang teratur dan dinamis.

Pada percobaan terakhir, digunakan **library Keypad.h** untuk membaca masukan dari **keypad 4x4**. Keypad ini memiliki empat baris dan empat kolom, masing-masing berisi karakter angka dan huruf seperti '1', '2', '3', 'A', hingga '#'. Pin baris dan kolom dihubungkan ke Arduino, lalu dipetakan dalam array `hexaKeys`. Dengan membuat objek `customKeypad`, Arduino bisa mengenali tombol mana yang ditekan berdasarkan kombinasi baris dan kolom tersebut. Dalam `setup()`, perintah `Serial.begin(9600)` digunakan untuk menampilkan hasil pembacaan tombol ke *Serial Monitor*. Saat program berjalan, ketika pengguna menekan salah satu tombol, karakter dari tombol tersebut langsung muncul di layar monitor. Dari sinilah kita bisa melihat bagaimana Arduino menerima dan memproses input pengguna secara real-time. Secara keseluruhan, ketiga program ini menggambarkan bagaimana Arduino dapat menjadi sistem interaktif yang sederhana tapi kuat. LCD berperan sebagai "layar suara", seven segment menjadi "angka bergerak", dan keypad menjadi "jari-jari pengguna" yang berinteraksi langsung dengan sistem. Ketiganya menunjukkan betapa fleksibelnya Arduino dalam menggabungkan logika pemrograman dengan perangkat keras untuk menciptakan antarmuka manusia-mesin yang komunikatif dan menarik.

Output:



#### IV. Kesimpulan.

Dari rangkaian percobaan yang telah dilakukan, dapat disimpulkan bahwa Arduino merupakan sistem yang sangat fleksibel dalam berinteraksi dengan berbagai perangkat input dan output. Pada percobaan pertama, LCD 16x2 mampu menampilkan teks sesuai dengan perintah yang diberikan, menandakan bahwa proses komunikasi antara Arduino dan layar berjalan dengan baik. Percobaan kedua menunjukkan bagaimana seven segment dapat menampilkan angka secara berurutan dari 0 hingga 9 dan kembali lagi, membuktikan kemampuan Arduino dalam mengatur logika digital dan waktu dengan presisi. Sedangkan pada percobaan ketiga, keypad 4x4 berhasil digunakan sebagai alat input yang dapat membaca setiap tombol yang ditekan dan menampilkannya secara langsung melalui Serial Monitor.

Secara keseluruhan, ketiga percobaan ini menunjukkan bahwa Arduino bukan hanya alat pembelajaran elektronik sederhana, tetapi juga fondasi penting dalam memahami konsep antarmuka manusia dan mesin. LCD berperan sebagai sarana tampilan visual, seven

segment sebagai penunjuk angka, dan keypad sebagai media komunikasi pengguna. Melalui kombinasi ketiganya, kita belajar bagaimana sistem mikrokontroler dapat dirancang agar mampu “mendengar”, “berpikir”, dan “berbicara” — menciptakan interaksi yang cerdas, efisien, dan inspiratif antara manusia dan teknologi.

