

LAPORAN PRAKTIKUM MOBILE PROGRAMMING
(Ujian Tengah Semester)



Nama : Andini Aulia Putri

NIM : 230605110059

Kelas : E

Tanggal : 23 Oktober 2025

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

BAB 1 – Pendahuluan

1.1 Deskripsi Umum

Aplikasi **SujudTime** adalah sebuah aplikasi mobile yang dirancang untuk membantu umat Muslim dalam menjaga kedisiplinan beribadah dengan cara mengingatkan waktu shalat fardhu dan shalat sunnah. Aplikasi ini tidak hanya menampilkan jadwal shalat berdasarkan lokasi pengguna, tetapi juga memberikan pengingat waktu sujud terbaik, serta doa dan motivasi singkat setelah shalat. SujudTime memungkinkan pengguna untuk memilih pengingat sesuai kebutuhan mereka, baik untuk shalat fardhu maupun sunnah seperti Dhuha, Tahajjud, Witir, dan Rawatib.

Selain itu, aplikasi ini juga dilengkapi dengan fitur catatan riwayat ibadah, di mana pengguna dapat mencatat dan memantau konsistensi ibadah harian mereka. Dengan statistik ibadah yang dapat ditampilkan dalam bentuk grafik, aplikasi ini memberikan dorongan bagi pengguna untuk meningkatkan kualitas dan kuantitas ibadah mereka. Aplikasi SujudTime menggabungkan teknologi dan fitur yang memungkinkan pengguna untuk lebih teratur dan khusyuk dalam menjalankan ibadah shalat.

1.2 Alasan Pemilihan Tema

Pemilihan tema aplikasi SujudTime didasari oleh kebutuhan untuk membantu umat Muslim dalam meningkatkan kedisiplinan beribadah. Dalam kehidupan sehari-hari yang penuh dengan kesibukan, sering kali banyak orang yang kesulitan untuk tepat waktu dalam melaksanakan shalat, terutama shalat sunnah yang sering kali dianggap sepele. Aplikasi ini hadir untuk memberikan solusi praktis bagi umat Muslim, dengan menyediakan pengingat waktu shalat fardhu dan sunnah serta fitur-fitur motivasi yang dapat meningkatkan konsistensi ibadah.

Tema ini juga relevan dengan meningkatnya penggunaan teknologi dalam kehidupan sehari-hari. Dengan adanya aplikasi berbasis mobile, umat Muslim dapat dengan mudah mengakses jadwal shalat dan pengingat ibadah di mana saja dan kapan saja. Hal ini tentunya mempermudah mereka dalam melaksanakan ibadah

dengan lebih tepat waktu, serta membantu mereka untuk selalu mengingat pentingnya waktu sujud terbaik.

1.3 Relevansi terhadap Pengembangan Aplikasi Mobile

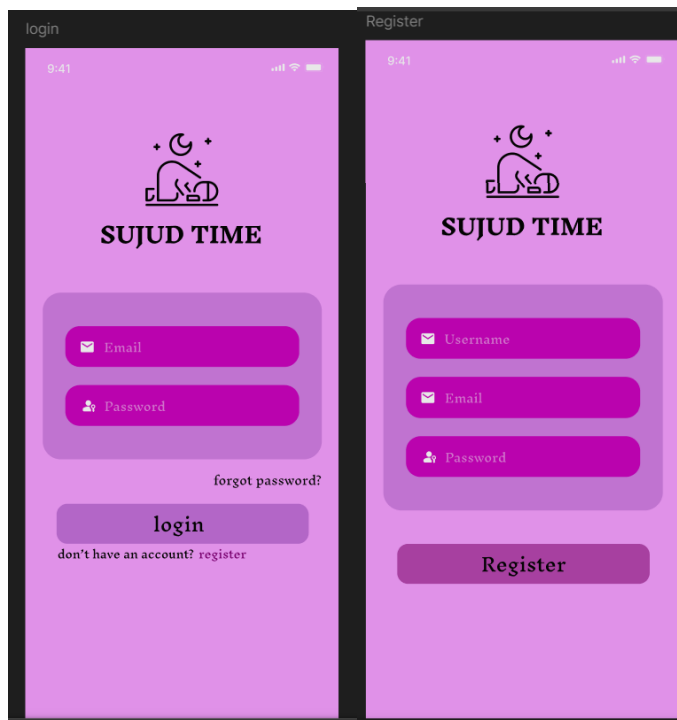
Relevansi tema aplikasi SujudTime dengan perkembangan aplikasi mobile saat ini sangat tinggi. Di era digital yang serba cepat dan modern ini, hampir semua aspek kehidupan dapat dijangkau melalui perangkat mobile, termasuk dalam hal ibadah. Penggunaan aplikasi mobile untuk mendukung aktivitas keagamaan bukan hanya praktis, tetapi juga memberikan nilai tambah berupa kemudahan dalam mengakses informasi dan pengingat yang dibutuhkan.

Selain itu, banyak aplikasi serupa yang telah berkembang di pasar, namun SujudTime menawarkan pendekatan yang lebih personal dengan fitur pengingat waktu sujud terbaik, pengingat shalat sunnah, dan riwayat ibadah yang dapat membantu pengguna dalam memantau perkembangan ibadah mereka. Hal ini membuat SujudTime tidak hanya sebagai aplikasi pengingat waktu, tetapi juga sebagai alat untuk meningkatkan kualitas spiritual pengguna.

Dengan semakin banyaknya pengguna smartphone dan meningkatnya ketergantungan terhadap perangkat mobile, aplikasi mobile kini menjadi sarana yang sangat efektif untuk mendukung kegiatan sehari-hari, termasuk dalam bidang ibadah. SujudTime hadir untuk memberikan solusi praktis dan bermanfaat bagi umat Muslim di seluruh dunia, menjadikannya aplikasi yang relevan dan penting dalam konteks perkembangan teknologi mobile saat ini.

BAB 2 - Perancangan Antarmuka (UI Design)

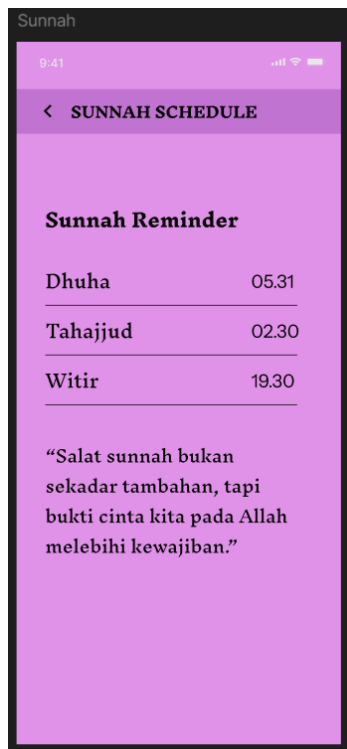
1. Page Login dan Register



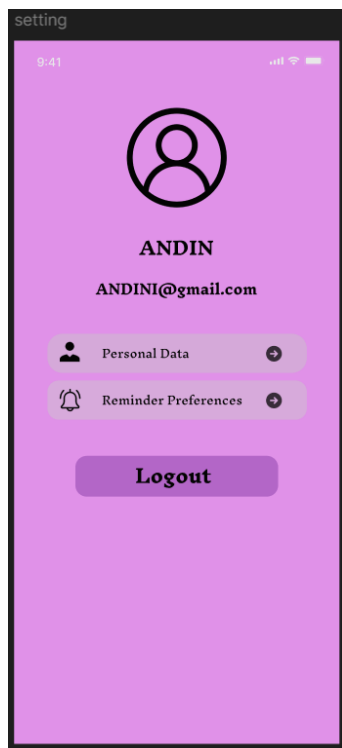
2. Page Home (Schedule Sholat Wajib)



3. Page Schedule Sholat Sunnah



4. Page Setting



5. Page Personal Data

The screenshot shows the 'Personal Data' page of a mobile application. At the top, there is a title bar with the text 'Personal Data' and a back arrow. Below the title bar, the status bar shows the time '9:41' and signal strength. The main content area has a light blue background. It features a circular profile picture placeholder with a white person icon. Below the profile picture, there are three text input fields labeled 'Full Name', 'Email', and 'Phone Number'. At the bottom of the form is a blue 'Save' button.

6. Page Reminder Preferences

The screenshot shows the 'Reminder Preferences' page of a mobile application. At the top, there is a title bar with the text 'Reminder Preferences' and a back arrow. Below the title bar, the status bar shows the time '9:41' and signal strength. The main content area has a light blue background. It features a section titled 'Prayer Reminder' with five toggle switches for 'Shubuh', 'Dhuhur', 'Ashar', 'Maghrib', and 'Isya'. Below this section is another section titled 'Sunnah Reminder' with three toggle switches for 'Dhuha', 'Tahajjud', and 'Witir'. At the bottom of the form is a blue 'Save' button.

BAB 3 – Implementasi Program

1. Login Code

- **Widget Utama (Login Page):**

Widget utama dalam kode ini adalah LoginPage, yang berfungsi sebagai halaman login. Di dalamnya terdapat form input untuk email dan password, tombol untuk login, dan tombol untuk berpindah ke halaman pendaftaran.

```
class LoginPage extends StatefulWidget {  
  const LoginPage({super.key});  
  
  @override  
  State<LoginPage> createState() => _LoginPageState();  
}
```

- **Navigasi Antar Halaman (Halaman Login ke Halaman Utama):**

Setelah berhasil login, pengguna akan diarahkan ke halaman MainPage dengan data pengguna yang berhasil login. Navigasi dilakukan menggunakan.

```
Navigator.pushReplacement(  
  context,  
  
  MaterialPageRoute(  
    builder: (_) => MainPage(user: user), // Pass user ke halaman MainPage  
  ));
```

- **Pengolahan Data (Login dengan API):**

Proses login dilakukan dengan memanggil API menggunakan `apiService.login`. Jika login berhasil, pengguna akan diarahkan ke halaman utama (MainPage), sementara jika gagal, snackbar akan ditampilkan.

```
Future<void> handleLogin() async {

  if (emailController.text.isEmpty || passwordController.text.isEmpty) {

    ScaffoldMessenger.of(context).showSnackBar(

      const SnackBar(content: Text('Email dan password harus diisi')),

    );

    return;
  }

  setState(() {

    isLoading = true;

  });

  try {

    final user = await apiService.login(

      emailController.text,

      passwordController.text,

    );

    if (user != null) {

      // Navigasi ke MainPage jika login sukses

    } else {

      ScaffoldMessenger.of(context).showSnackBar(

        const SnackBar(content: Text('Login gagal! Email atau password salah')),

      );

    }

  }
```



```

    } catch (e) {

        ScaffoldMessenger.of(context).showSnackBar(

            SnackBar(content: Text('Error: $e')),

        );

    } finally {

        setState() {

            isLoading = false;

        });

    }

}

```

2. Register Code

- **Widget Utama (Register Page):**

Widget utama dalam kode ini adalah RegisterPage, yang berfungsi sebagai halaman untuk registrasi pengguna baru. Terdapat form input untuk nama, email, dan password, serta tombol untuk melakukan registrasi.

```

class RegisterPage extends StatefulWidget {

    const RegisterPage({super.key});

    @override

    State<RegisterPage> createState() => _RegisterPageState();

}

```

- **Navigasi Antar Halaman (Kembali ke Halaman Sebelumnya Setelah Registrasi Berhasil):**

Setelah berhasil melakukan registrasi, pengguna akan kembali ke halaman sebelumnya dengan menggunakan Navigator.pop(context).

```

    if (success) {

        ScaffoldMessenger.of(context).showSnackBar(

```

```

const SnackBar(content: Text("Registrasi berhasil!")),

);

Navigator.pop(context); // Kembali ke halaman sebelumnya (halaman
login)

}

```

3. Home code

- **Widget Utama (Main Page):**

MainPage adalah halaman utama yang menampilkan daftar waktu shalat yang diperoleh dari list prayers. Widget ini menggunakan ListTile untuk menampilkan nama dan waktu shalat.

```

body: Padding(

padding: const EdgeInsets.all(16.0),

child: Column(

children: [

...prayers.map((p) => ListTile(

title: Text(p['name']!, style: const TextStyle(fontSize: 18)),

trailing: Text(p['time']!, style: const TextStyle(fontWeight:
FontWeight.bold, fontSize: 16)),

)),

),

```

- **Navigasi Antar Halaman (Ke Halaman Settings dan Sunnah Time):**

Pada halaman ini terdapat dua navigasi:

- **Ke halaman Settings:** Ketika ikon pengaturan ditekan, halaman SettingsPage ditampilkan dengan membawa data user.

```

IconButton(

icon: const Icon(Icons.settings),

onPressed: () {

```

```

Navigator.push(

  context,

  MaterialPageRoute(

    builder: (_) => SettingsPage(user: user), // Pass user ke halaman SettingsPage

  ), ); ;,)

```

- **Ke halaman Sunnah Time:** Ketika tombol "Sunnah Time" ditekan, halaman SunnahPage akan terbuka.

```

ElevatedButton(

  onPressed: () {

    Navigator.push(

      context,

      MaterialPageRoute(builder: (_) => SunnahPage()); // Navigasi ke halaman SunnahPage },

  style: ElevatedButton.styleFrom(

    backgroundColor: Colors.purple.shade400,

    shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(12))),

  child: const Text("Sunnah Time"),)

```

- **Pengolahan Data (Daftar Waktu Shalat):**
Waktu shalat ditampilkan menggunakan ListTile yang dihasilkan dari iterasi list prayers. Setiap item dalam list prayers menampilkan nama shalat dan waktu yang sesuai.

```

...prayers.map((p) => ListTile(

  title: Text(p['name']!, style: const TextStyle(fontSize: 18)),

  trailing: Text(p['time']!, style: const TextStyle(fontWeight: FontWeight.bold,
fontSize: 16)), )),)

```

4. Sunnah Code

- **Widget Utama (Sunnah Page):**

SunnahPage menampilkan daftar salat sunnah dengan waktu yang telah ditentukan dan sebuah kutipan motivasi. Data salat sunnah disimpan dalam list sunnahPrayers, dan tampil menggunakan widget ListTile.

```
class SunnahPage extends StatelessWidget {  
  
  final List<Map<String, String>> sunnahPrayers = [  
  
    {'name': 'Dhuha', 'time': '05:31'},  
  
    {'name': 'Tahajjud', 'time': '02:50'},  
  
    {'name': 'Witir', 'time': '19:50'},  
  
  ];  
  
  final String quote = "Salat sunnah bukan sekadar tambahan, tapi\nbukti cinta  
kita pada Allah melalui\nkejujuran.";
```

- Navigasi Antar Halaman (Kembali ke Halaman Sebelumnya)

```
leading: IconButton(  
  
  icon: const Icon(Icons.arrow_back),  
  
  onPressed: () => Navigator.pop(context), // Kembali ke halaman sebelumnya  
  
),
```

- Pengolahan Data (Daftar Salat Sunnah)

```
...sunnahPrayers.map((prayer) => Container(  
  
  margin: const EdgeInsets.only(bottom: 12),  
  
  decoration: BoxDecoration(  
  
    color: Colors.white,  
  
    borderRadius: BorderRadius.circular(12),
```

```

        boxShadow: [
          BoxShadow(
            color: Colors.black.withOpacity(0.05),
            blurRadius: 8,
            offset: const Offset(0, 2),
          ),
        ],
      ),
      child: ListTile(
        contentPadding: const EdgeInsets.symmetric(horizontal: 20, vertical: 8),
        title: Text(
          prayer['name']!,
          style: const TextStyle(fontSize: 18, fontWeight: FontWeight.w500, color:
Colors.black87),
        ),
        trailing: Text(
          prayer['time']!,
          style: TextStyle(fontSize: 20, fontWeight: FontWeight.bold, color:
Colors.purple.shade400),
        ),
      ),
    )),

```

- Bagian Kutipan (Quote Section)

```

Container(
  width: double.infinity,
  padding: const EdgeInsets.all(20),
  decoration: BoxDecoration(
    color: Colors.purple.shade50,
    borderRadius: BorderRadius.circular(16),
    border: Border.all(color: Colors.purple.shade200, width: 1),
  ),
  child: Column(
    children: [
      Icon(Icons.format_quote, size: 32, color: Colors.purple.shade300),
      const SizedBox(height: 8),
      Text(
        quote,
        textAlign: TextAlign.center,
        style: TextStyle(fontSize: 14, fontStyle: FontStyle.italic, color:
Colors.grey[700], height: 1.5),
      ),
    ],
  ),
),

```

5. Page Setting Code

- **Widget Utama (Settings Page)**
SettingsPage adalah halaman pengaturan yang menampilkan informasi pengguna seperti nama dan email, serta menyediakan navigasi ke halaman lain seperti PersonalDataPage dan ReminderPreferencesPage.

```
class SettingsPage extends StatelessWidget {

  final UserModel user;

  const SettingsPage({super.key, required this.user});
```

- **Navigasi Antar Halaman (Ke Halaman Personal Data dan Reminder Preferences)**
 - **Ke halaman Personal Data:** Ketika pengguna menekan ListTile "Personal Data", aplikasi akan melakukan navigasi ke halaman PersonalDataPage dan meneruskan data user.

```
ListTile(

  leading: const Icon(Icons.person),

  title: const Text('Personal Data'),

  trailing: const Icon(Icons.arrow_forward_ios),

  onTap: () {

    Navigator.push(

      context,

      MaterialPageRoute(

        builder: (_) => PersonalDataPage(user: user), // Pass user data

      ),

    );

  },

),
```

- **Ke halaman Reminder Preferences:** Ketika pengguna menekan ListTile "Reminder Preferences", aplikasi akan melakukan navigasi ke halaman ReminderPreferencesPage dan meneruskan data user.

```
ListTile(  
  
  leading: const Icon(Icons.notifications),  
  
  title: const Text('Reminder Preferences'),  
  
  trailing: const Icon(Icons.arrow_forward_ios),  
  
  onTap: () {  
  
    Navigator.push(  
  
      context,  
  
      MaterialPageRoute(  
  
        builder: (_) => ReminderPreferencesPage(user: user), // Pass user data  
  
      ),  
  
    );  
  
  },  
  
),
```

- **Menampilkan Data Pengguna**
Data pengguna yang diteruskan ke SettingsPage (seperti nama dan email) ditampilkan di bagian atas halaman.

```
CircleAvatar(  
  
  radius: 50,  
  
  backgroundColor: Colors.grey.shade300,  
  
  child: const Icon(Icons.person, size: 50, color: Colors.black),  
  
),  
  
const SizedBox(height: 10),
```



```

Text(
  user.name, // Nama pengguna
  style: const TextStyle(fontSize: 24, fontWeight: FontWeight.bold),
),
const SizedBox(height: 5),
Text(
  user.email, // Email pengguna
  style: const TextStyle(fontSize: 18, color: Colors.black45),
),

```

- **Logout**
Tombol "Logout" akan membawa pengguna kembali ke halaman login dengan menggunakan `Navigator.popUntil(context, ModalRoute.withName('/login'))`.

```

ElevatedButton(
  onPressed: () {
    Navigator.popUntil(context, ModalRoute.withName('/login'));
  },
  style: ElevatedButton.styleFrom(
    backgroundColor: Colors.purple.shade400,
    padding: const EdgeInsets.symmetric(horizontal: 80, vertical: 14),
    shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(12)),
  ),
  child: const Text('Logout', style: TextStyle(color: Colors.white)),
),

```

6. Personal Data Code

- **Widget Utama (Personal Data Page):**
 PersonalDataPage adalah halaman untuk menampilkan dan mengedit data pribadi pengguna. Pengguna dapat mengubah nama, email, dan nomor telepon mereka. Data ini diambil melalui API dan dimuat menggunakan TextEditingController.

```
class PersonalDataPage extends StatefulWidget {

  final UserModel user;

  const PersonalDataPage({super.key, required this.user});
```

- **Mengambil dan Menampilkan Data Pengguna**
 Data pengguna dimuat menggunakan `_loadUserData` yang memanggil API untuk mengambil profil pengguna. Setelah data dimuat, nilai-nilai seperti nama, email, dan nomor telepon akan ditampilkan di form.

```
Future<void> _loadUserData() async {

  final profile = await _apiService.getUserProfile(widget.user.id);

  if (profile != null) {

    setState(() {

      _nameController.text = profile['name'] ?? widget.user.name;

      _emailController.text = profile['email'] ?? widget.user.email;

      _phoneController.text = profile['phone'] ?? "";

    });

  }

}
```

- **Form untuk Mengedit Data Pribadi**
 Halaman ini menggunakan TextFormField untuk mengedit nama, email, dan nomor telepon. Form ini juga memvalidasi input pengguna agar tidak kosong.

```
TextFormField(

  controller: _nameController,
```

```
decoration: InputDecoration(labelText: 'Full Name'),  
  
validator: (value) => value!.isEmpty ? 'Please enter your full name' : null,  
  
),
```

- Navigasi Kembali ke Halaman Sebelumnya:
Ketika pengguna menekan ikon panah pada AppBar, halaman ini akan menutup dan mengarahkan kembali pengguna ke halaman sebelumnya.

```
leading: IconButton(  
  
  icon: const Icon(Icons.arrow_back),  
  
  onPressed: () => Navigator.pop(context), // Kembali ke halaman sebelumnya  
  
),
```

7. Reminder Preferences Code

- Widget Utama (Reminder Preferences Page)
ReminderPreferencesPage adalah halaman yang memungkinkan pengguna untuk mengatur preferensi pengingat salat. Pengguna dapat mengaktifkan atau menonaktifkan pengingat untuk waktu salat fardhu dan sunnah.

```
class ReminderPreferencesPage extends StatefulWidget {  
  
  final UserModel user;  
  
  const ReminderPreferencesPage({super.key, required this.user});  
  
  @override  
  
  State<ReminderPreferencesPage> createState() =>  
    _ReminderPreferencesPageState();  
}
```

- Memuat Data Preferensi Pengguna
Mengambil preferensi pengingat dari SharedPreferences untuk mengatur status switch sesuai dengan data yang disimpan.

```
Future<void> _loadPreferences() async {  
  
  setState(() => _isLoading = true);  
  
}
```

```

final prefs = await SharedPreferences.getInstance();

setState(() {
  shubuhReminder = prefs.getBool('shubuhReminder') ?? true;
  dhuhurReminder = prefs.getBool('dhuhurReminder') ?? true;
  // Load the other reminders...
  _isLoading = false;
});
}

```

- Tampilan Switch untuk Peningat Salat:

```

Widget _buildSwitchTile(
  String title,
  bool value,
  Function(bool) onChanged,
) {
  return SwitchListTile(
    title: Text(title),
    value: value,
    onChanged: onChanged,
    activeColor: Colors.purple.shade400,
  );
}

```

BAB 4 - Hasil dan Pembahasan

4.1 Hasil Implementasi Aplikasi

Aplikasi **Sujud Time** berhasil mengimplementasikan desain awal dengan fungsionalitas yang sesuai. Halaman login dan registrasi memungkinkan pengguna untuk masuk atau membuat akun baru, sementara halaman utama menampilkan waktu shalat fardhu dan tombol untuk mengakses halaman sunnah. Pengguna juga dapat mengatur preferensi pengingat waktu salat, baik fardhu maupun sunnah, melalui halaman pengaturan dan data pribadi.

Interaksi pengguna berjalan lancar dengan pengaturan yang mudah diakses, seperti memperbarui data pribadi melalui halaman **Personal Data Page** dan menyesuaikan pengingat salat di **Reminder Preferences Page**. Setiap perubahan yang dilakukan oleh pengguna memberikan umpan balik yang jelas, baik melalui indikator loading atau snackbar yang menampilkan status penyimpanan.

Secara keseluruhan, aplikasi ini berhasil memenuhi tujuannya untuk membantu pengguna mengingat waktu salat fardhu dan sunnah, serta memperbarui data pribadi dengan antarmuka yang sederhana namun efektif. Warna ungu yang konsisten dalam desain memperkuat pengalaman visual yang menyenangkan bagi pengguna.

4.2 Kesesuaian Desain dan Implementasi

Halaman	Desain Awal	Implementasi Akhir	Kesesuaian
Login Page	Menampilkan input untuk Email dan Password, tombol Login, dan link untuk registrasi. Latar belakang berwarna ungu muda.	Implementasi halaman login sesuai dengan desain awal, menampilkan input email dan password, serta tombol login dengan link untuk	Sangat Sesuai: Fungsionalitas dan desain halaman login sudah diterapkan dengan benar, sesuai dengan desain yang direncanakan.

Halaman	Desain Awal	Implementasi Akhir	Kesesuaian
		registrasi. Latar belakang berwarna ungu.	
Register Page	Menampilkan input untuk Username, Email, dan Password dengan tombol Register berwarna ungu.	Halaman registrasi juga mengikuti desain awal dengan input untuk Username, Email, dan Password serta tombol registrasi berwarna ungu.	Sangat Sesuai: Desain dan interaksi pada halaman registrasi sudah sesuai dengan rencana.
Main Page	Menampilkan daftar waktu shalat dengan Shubuh, Dhuhur, Ashar, Maghrib, dan Isya, dan tombol untuk menuju Sunnah Time.	Implementasi akhir menampilkan waktu salat fardhu yang sama, serta tombol untuk membuka halaman Sunnah Time dengan desain yang konsisten.	Sangat Sesuai: Tampilan dan fungsionalitas halaman utama sudah sesuai dengan desain yang diinginkan.
Sunnah Time Page	Menampilkan waktu-waktu shalat sunnah (Dhuha, Tahajjud, Witir) dan kutipan motivasi.	Implementasi akhir mencakup waktu-waktu shalat sunnah (Dhuha, Tahajjud, Witir) dengan kutipan yang memotivasi pengguna.	Sangat Sesuai: Pengingat sunnah dan kutipan motivasi telah diimplementasikan sesuai dengan desain awal.
Settings Page	Menampilkan informasi pengguna dan menyediakan opsi	Halaman pengaturan sesuai dengan desain awal dengan informasi pengguna dan opsi	Sangat Sesuai: Fungsionalitas pengaturan dan navigasi berjalan

Halaman	Desain Awal	Implementasi Akhir	Kesesuaian
	untuk mengakses Personal Data dan Reminder Preferences.	pengaturan untuk data pribadi dan preferensi pengingat.	sesuai dengan yang diharapkan.
Personal Data Page	Menampilkan form untuk mengedit Full Name, Email, dan Phone Number dengan tombol Save.	Halaman untuk mengedit data pribadi sesuai dengan desain awal, menampilkan form untuk Full Name, Email, dan Phone Number dengan tombol Save.	Sangat Sesuai: Implementasi halaman ini sudah sesuai dengan desain dan fungsionalitas yang direncanakan.
Reminder Preferences Page	Menampilkan pengaturan preferensi pengingat salat fardhu dan sunnah dengan opsi untuk mengaktifkan atau menonaktifkan pengingat.	Halaman preferensi pengingat salat diimplementasikan dengan fitur untuk mengaktifkan atau menonaktifkan pengingat salat fardhu dan sunnah, serta tombol Save untuk menyimpan preferensi.	Sangat Sesuai: Pengaturan pengingat bekerja dengan baik sesuai dengan desain yang telah ditetapkan.

BAB 5 – Kesimpulan

Aplikasi **Sujud Time** telah berhasil dikembangkan dengan mengikuti tahapan perancangan, implementasi, dan pengujian yang matang. Proses dimulai dengan perancangan antarmuka yang sederhana namun efektif, dengan fokus pada kemudahan penggunaan dan estetika yang konsisten. Desain antarmuka aplikasi

dirancang dengan dominasi warna ungu yang memberikan kesan tenang dan spiritual, serta penggunaan elemen-elemen fungsional seperti input form dan tombol yang jelas. Seluruh desain mengikuti prinsip keterbacaan dan kemudahan akses, memastikan pengguna dapat dengan mudah mengakses fitur aplikasi tanpa kebingungannya.

Pada tahap implementasi, setiap halaman dalam aplikasi, mulai dari **Login Page**, **Register Page**, **Main Page**, **Sunnah Time Page**, hingga halaman pengaturan seperti **Settings Page**, **Personal Data Page**, dan **Reminder Preferences Page**, diimplementasikan dengan baik sesuai dengan desain awal. Fungsionalitas aplikasi berjalan dengan lancar, memungkinkan pengguna untuk melakukan login, registrasi, melihat pengingat waktu shalat fardhu dan sunnah, serta mengelola preferensi pengingat mereka dengan mudah. Selain itu, aplikasi juga telah dilengkapi dengan mekanisme pengambilan dan penyimpanan data melalui API dan **SharedPreferences**, memastikan data pengguna tetap aman dan dapat diperbarui dengan mudah.

Setelah implementasi, pengujian dilakukan untuk memastikan semua fitur berfungsi dengan baik. Pengujian berhasil mengonfirmasi bahwa aplikasi dapat memenuhi fungsinya sesuai dengan ekspektasi, seperti mengingatkan pengguna tentang waktu salat dan memungkinkan mereka untuk memperbarui data pribadi serta preferensi pengingat dengan lancar. Salah satu pencapaian utama dari aplikasi ini adalah kemampuannya untuk menyediakan pengingat salat yang komprehensif, baik untuk salat fardhu maupun sunnah, serta memberikan pengelolaan data yang mudah melalui antarmuka yang sederhana.

Refleksi terhadap hasil yang dicapai menunjukkan bahwa aplikasi **Sujud Time** telah memenuhi tujuan yang ditetapkan, dengan pengalaman pengguna yang menyenangkan dan bermanfaat. Namun, masih terdapat potensi pengembangan lanjutan, seperti integrasi dengan fitur notifikasi untuk mengingatkan pengguna saat waktu salat tiba, serta penambahan fitur komunitas untuk berbagi pengingat dengan pengguna lain. Pengembangan lanjutan ini dapat meningkatkan pengalaman pengguna lebih jauh, membuat aplikasi ini lebih interaktif dan sesuai dengan

kebutuhan pengguna modern yang membutuhkan pengingat waktu salat yang fleksibel dan efektif.