**TUGAS 2**

**Implementasi dan Pengujian PL**

****

**Disusun Oleh :**

Ahmad Hazli Yahya – 1203222029

**INFORMATIKA**

**TELKOM UNIVERSITY**

**SURABAYA 2024**

**TUGAS #1**

1. **PSPEC**

Nama Proses : User Authentication

Input : Email dan Password

Proses :

* Verifikasi email dan Password.
* Check akun terdaftar di Firebase Authentication.
* Jika benar, Masuk ke Aplikasi.
* Jika salah, Muncul notifikasi Error

Output :

* Sukses login ke Aplikasi
* Gagal login dan muncul notif error

1. **Stuctured English (Pseudocode)**

A screen shot of a computer

Description automatically generated

1. **DFD**

* DFD Level 1

A diagram of a company

Description automatically generated

* DFD Level 2

A diagram of a company

Description automatically generated

1. Source Code

import 'package:firebase\_auth/firebase\_auth.dart';

import 'package:flutter/gestures.dart';

import 'package:flutter/material.dart';

import 'package:logger/logger.dart';

import 'package:pharmacyapp/common/helper/navigator/app\_navigator.dart';

import 'package:pharmacyapp/common/widgets/appbar/app\_bar.dart';

import 'package:pharmacyapp/common/widgets/button/basic\_app\_button.dart';

import 'package:pharmacyapp/presentation/auth/pages/forgot\_password.dart';

import 'package:pharmacyapp/presentation/auth/pages/signup.dart';

import 'package:pharmacyapp/presentation/home/pages/home.dart';

class SigninPage extends StatefulWidget {

  const SigninPage({super.key});

  @override

  State<SigninPage> createState() => \_SigninPageState();

}

class \_SigninPageState extends State<SigninPage> {

  final TextEditingController \_emailCon = TextEditingController();

  final TextEditingController \_passwordCon = TextEditingController();

  final Logger \_logger = Logger();

  final FirebaseAuth \_auth = FirebaseAuth.instance;

  @override

  void dispose() {

    \_emailCon.dispose();

    \_passwordCon.dispose();

    super.dispose();

  }

  @override

  Widget build(BuildContext context) {

    return Scaffold(

      appBar: const BasicAppbar(hideBack: true),

      body: Padding(

        padding: const EdgeInsets.symmetric(horizontal: 16, vertical: 40),

        child: Column(

          crossAxisAlignment: CrossAxisAlignment.start,

          children: [

            \_signInText(),

            const SizedBox(height: 20),

            \_emailField(),

            const SizedBox(height: 20),

            \_passwordField(),

            const SizedBox(height: 20),

            \_continueButton(),

            const SizedBox(height: 20),

            \_forgotPassword(context),

            const SizedBox(height: 20),

            \_createAccount(),

          ],

        ),

      ),

    );

  }

  Widget \_signInText() {

    return const Text(

      'Sign in',

      style: TextStyle(fontSize: 32, fontWeight: FontWeight.bold),

    );

  }

  Widget \_emailField() {

    return TextField(

      controller: \_emailCon,

      decoration: const InputDecoration(hintText: 'Enter Email'),

      keyboardType: TextInputType.emailAddress,

    );

  }

  Widget \_passwordField() {

    return TextField(

      controller: \_passwordCon,

      decoration: const InputDecoration(hintText: 'Enter Password'),

      obscureText: true,

    );

  }

  Future<bool> \_signInWithEmail(String email, String password) async {

    try {

      await \_auth.signInWithEmailAndPassword(

        email: email,

        password: password,

      );

      return true; // Sign-in successful

    } on FirebaseAuthException catch (e) {

      // Handle sign-in errors

      if (e.code == 'user-not-found') {

        \_showSnackBar('Email not registered');

      } else if (e.code == 'wrong-password') {

        \_showSnackBar('Incorrect password');

      } else {

        \_showSnackBar('An error occurred, please try again');

      }

      return false; // Sign-in failed

    }

  }

  void \_showSnackBar(String message) {

    if (!mounted) return; // Check if mounted

    ScaffoldMessenger.of(context).showSnackBar(

      SnackBar(content: Text(message)),

    );

  }

  Widget \_continueButton() {

    return BasicAppButton(

      onPressed: () async {

        final email = \_emailCon.text;

        final password = \_passwordCon.text;

        if (email.isNotEmpty && password.isNotEmpty) {

          final isEmailValid = RegExp(

            r'^[a-zA-Z0-9.\_%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$',

          ).hasMatch(email);

          if (!isEmailValid) {

            \_showSnackBar('Please enter a valid email');

            return;

          }

          \_logger.i('Attempting to sign in with email $email...');

          // Perform sign-in and check if the widget is still mounted

          final success = await \_signInWithEmail(email, password);

          if (mounted && success) {

            // Ensure the widget is still mounted

            AppNavigator.pushAndRemove(context, const HomePage());

          }

        } else {

          \_showSnackBar('Please enter your email and password');

        }

      },

      title: 'Continue',

    );

  }

  Widget \_createAccount() {

    return RichText(

      text: TextSpan(

        children: [

          const TextSpan(text: "Don't you have an account? "),

          TextSpan(

            text: 'Create one',

            recognizer: TapGestureRecognizer()

              ..onTap = () {

                AppNavigator.push(context, SignupPage());

              },

            style: const TextStyle(fontWeight: FontWeight.bold),

          ),

        ],

      ),

    );

  }

  Widget \_forgotPassword(BuildContext context) {

    return RichText(

      text: TextSpan(children: [

        const TextSpan(text: "Forgot password? "),

        TextSpan(

          text: 'Reset',

          recognizer: TapGestureRecognizer()

            ..onTap = () {

              AppNavigator.push(context, ForgotPasswordPage());

            },

          style: const TextStyle(fontWeight: FontWeight.bold),

        )

      ]),

    );

  }

}

1. Hasil Source Code

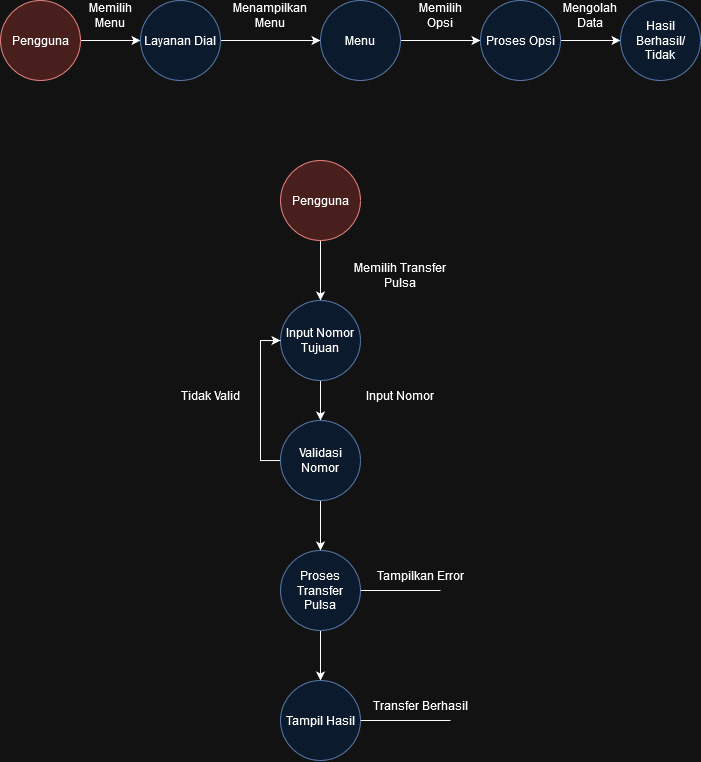
A screenshot of a login form

Description automatically generated

Tugas #2

1. DFD

* DFD Level 0



* DFD Level 1

A diagram of a process flow

Description automatically generated

1. PSPEC

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

1. Source Code

Index.html

<!DOCTYPE html>

<html lang="id">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Layanan Dial \*888#</title>

    <link rel="stylesheet" href="/css/style.css">

</head>

<body>

    <div class="container">

        <h1>Layanan Dial \*888#</h1>

        <div id="menu">

            <h2>Menu Awal</h2>

            <ol>

                <li><button onclick="showTransfer()">Transfer Pulsa</button></li>

                <li><button onclick="alert('Fitur ini belum tersedia.')">Minta Pulsa</button></li>

                <li><button onclick="alert('Fitur ini belum tersedia.')">Auto TP</button></li>

                <li><button onclick="alert('Fitur ini belum tersedia.')">Delete Auto TP</button></li>

                <li><button onclick="alert('Fitur ini belum tersedia.')">List Auto TP</button></li>

                <li><button onclick="alert('Fitur ini belum tersedia.')">Cek Kupon Undian TP</button></li>

            </ol>

        </div>

        <div id="transferSection" class="hidden">

            <h2>Transfer Pulsa</h2>

            <label for="phoneNumber">Nomor Tujuan:</label>

            <input type="text" id="phoneNumber" placeholder="Masukkan nomor tujuan" />

            <button onclick="executeTransfer()">Kirim</button>

            <div id="transferResult" class="hidden"></div>

        </div>

    </div>

    <script src="/js/script.js"></script>

</body>

</html>

Style.css

body {

    font-family: Arial, sans-serif;

    background-color: #f4f4f4;

    margin: 0;

    padding: 20px;

}

.container {

    max-width: 600px;

    margin: auto;

    background: white;

    padding: 20px;

    border-radius: 5px;

    box-shadow: 0 0 10px rgba(0, 0, 0, 0.1);

}

h1 {

    text-align: center;

    color: #333;

}

button {

    padding: 10px 15px;

    margin: 5px 0;

    background-color: #007bff;

    color: white;

    border: none;

    border-radius: 5px;

    cursor: pointer;

}

button:hover {

    background-color: #0056b3;

}

.hidden {

    display: none;

}

input[type="text"] {

    width: calc(100% - 30px);

    padding: 10px;

    margin: 10px 0;

    border: 1px solid #ccc;

    border-radius: 5px;

}

#transferResult {

    margin-top: 10px;

    font-size: 12px;

    color: black;

}

Script.js

function showTransfer() {

    document.getElementById('menu').classList.add('hidden');

    document.getElementById('transferSection').classList.remove('hidden');

}

function executeTransfer() {

    const phoneNumber = document.getElementById('phoneNumber').value;

    const result = document.getElementById('transferResult');

    if (phoneNumber) {

        result.textContent = `Transfer pulsa ke ${phoneNumber} berhasil!`;

        result.classList.remove('hidden');

    } else {

        alert('Silakan masukkan nomor tujuan!');

    }

}

Tampilan:

A screenshot of a chat

Description automatically generated

A screenshot of a phone number

Description automatically generated

Link github :