**TUGAS 09**

**PRAK10 Frontend**



Praktikum Pemrograman Berbasis Web - A

Nama :

Adelia Nurlina Putri 4521210059

**PROGRAM STUDI S1 TEKNIK INFORMATIKA**

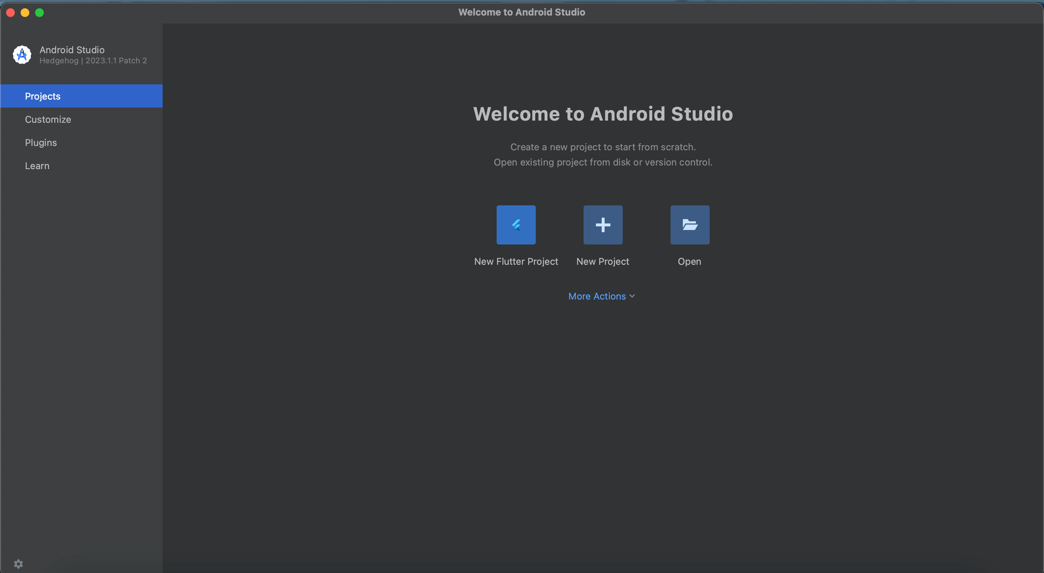
**FAKULTAS TEKNIK**

**UNIVERSITAS PANCASILA**

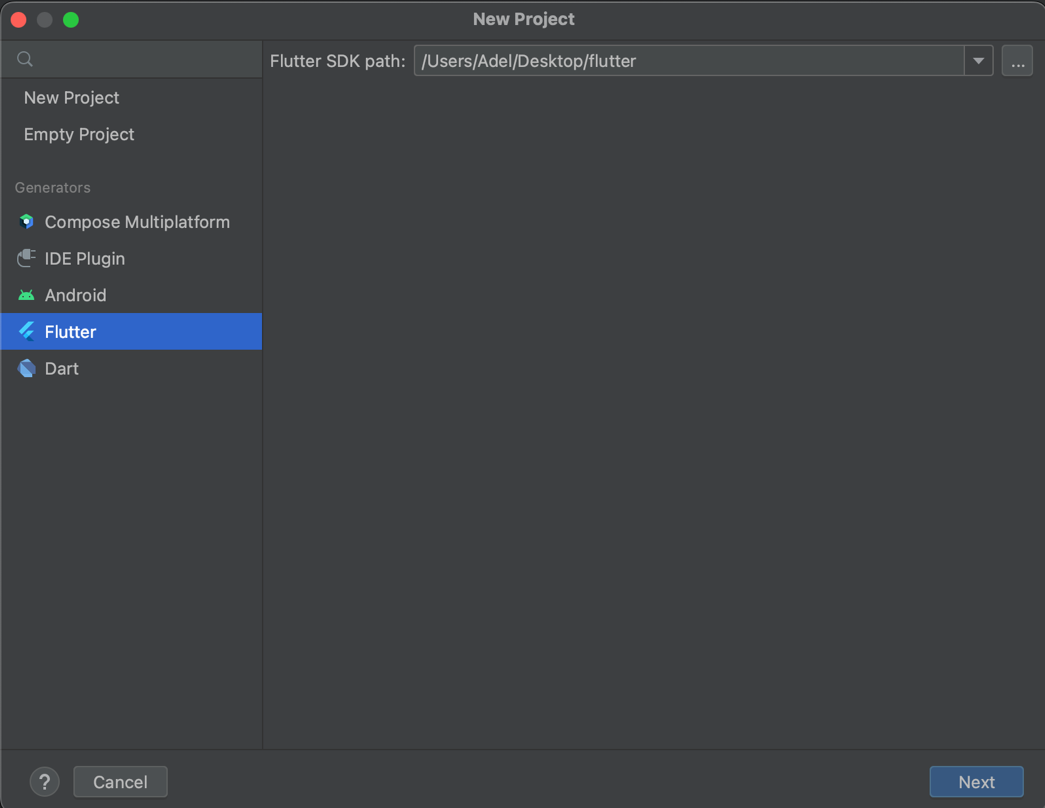
**JAKARTA**

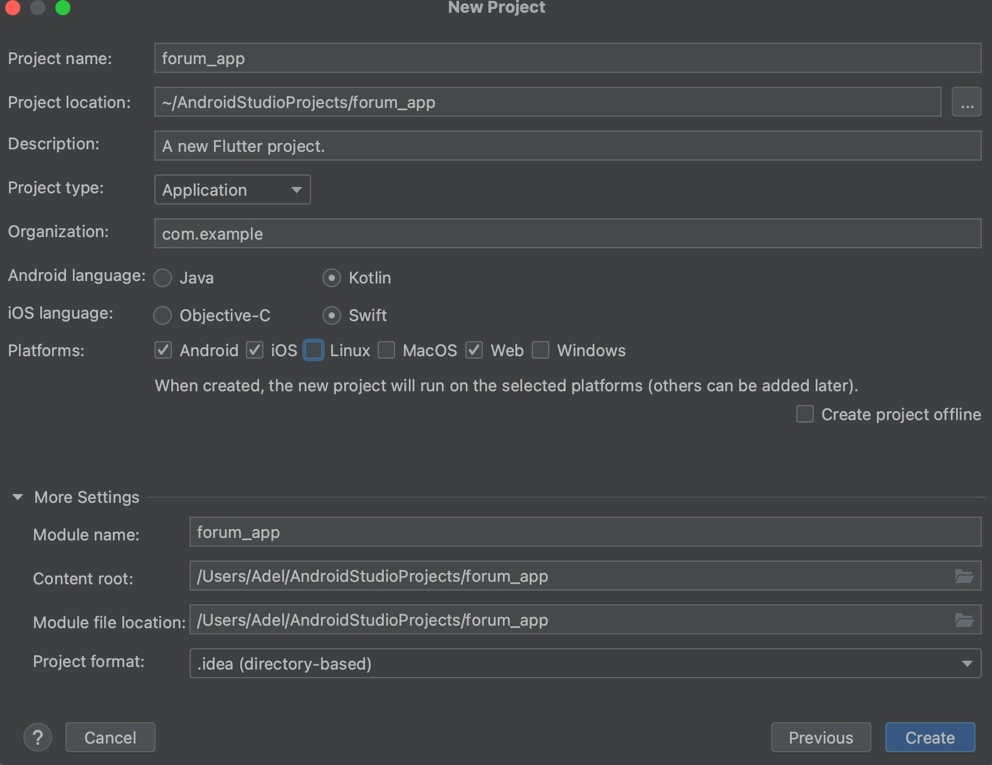
**2024**

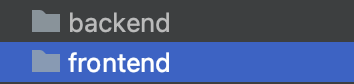
1. **Project prak10**
   1. **First Initialization**
2. Akses web <https://docs.flutter.dev/ui/layout/tutorial>
3. Buka Android Studio

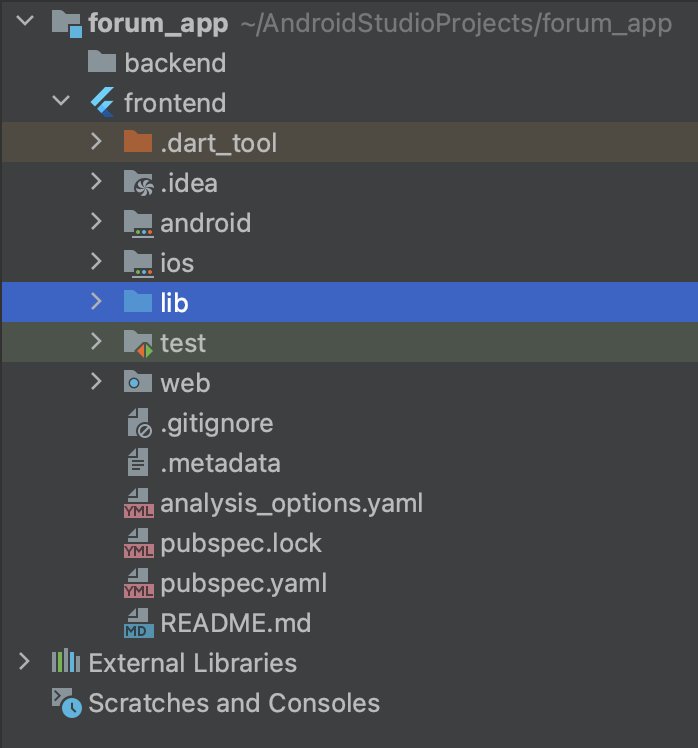


1. Pilih New Flutter Project -> Flutter -> Next

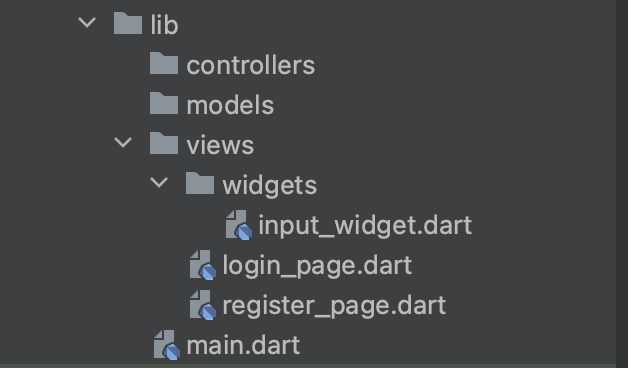


1. Isi form untuk nama project, tempat project, type, bahasa pemrograman, platforms dan lainnya -> Create
2. 
   1. **Membuat App**
3. Buat folder backend & frontend. Dan masukkan seluruh file project ke frontend folder



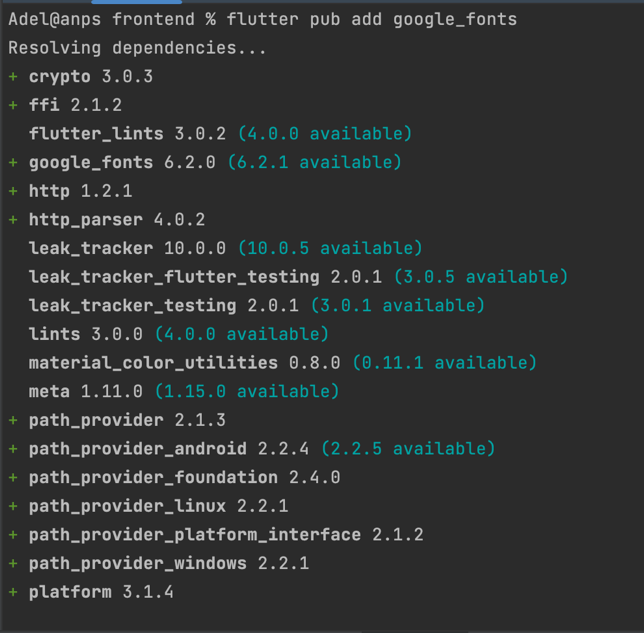


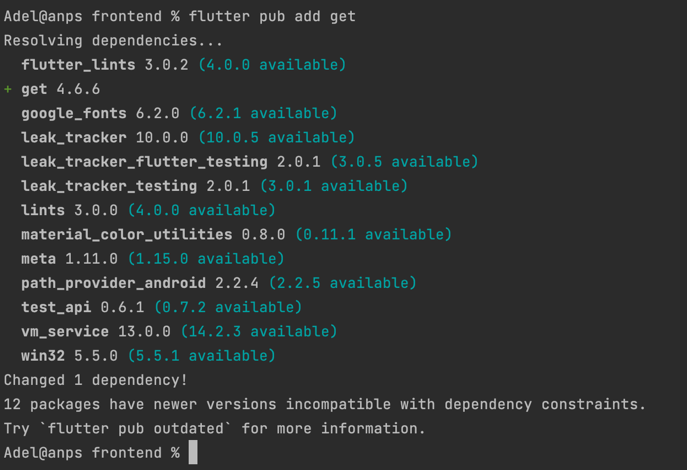
1. Buat folder dan file dengan susunan seperti berikut



1. Add package

|  |
| --- |
| $ flutter pub add google\_fonts  $ flutter pub add get |





1. File main.dart

|  |
| --- |
| import 'package:flutter/material.dart'; import 'views/login\_page.dart';  void main() {  runApp(const MyApp()); }  class MyApp extends StatelessWidget {  const MyApp({super.key});   @override  Widget build(BuildContext context) {  return const MaterialApp(  debugShowCheckedModeBanner: false,  title: 'Forum Application',  home: const LoginPage(),  );  } } |

1. File input\_widget.dart

|  |
| --- |
| import 'package:flutter/material.dart'; import 'package:google\_fonts/google\_fonts.dart';  class InputWidget extends StatelessWidget {  const InputWidget({  super.key,  required this.hintText,  required this.controller,  required this.obscureText,  });   final String hintText;  final TextEditingController controller;  final bool obscureText;   @override  Widget build(BuildContext context) {  return Container(  width: double.infinity,  decoration: BoxDecoration(  color: Colors.grey[200],  borderRadius: BorderRadius.circular(10),  ),  child: TextField(  obscureText: obscureText,  controller: controller,  decoration: InputDecoration(  border: InputBorder.none,  hintText: hintText,  hintStyle: GoogleFonts.poppins(),  contentPadding: const EdgeInsets.symmetric(horizontal: 20),  ),  ),  );  } } |

1. File login\_page.dart

|  |
| --- |
| import 'package:flutter/material.dart'; import 'package:google\_fonts/google\_fonts.dart';  class LoginPage extends StatefulWidget {  const LoginPage({super.key});   @override  State<LoginPage> createState() => \_LoginPageState(); }  class \_LoginPageState extends State<LoginPage> {  final TextEditingController \_emailController = TextEditingController();  final TextEditingController \_passwordController = TextEditingController();   @override  Widget build(BuildContext context) {  var size = MediaQuery.*of*(context).size.width;  return Scaffold(  body: Center(  child: Padding(  padding: const EdgeInsets.symmetric(horizontal: 20.0),  child: Column(  mainAxisAlignment: MainAxisAlignment.center,  children: [  Text(  'Login Page',  style: GoogleFonts.*poppins*(  fontSize: size \* 0.080,  ),  ),  const SizedBox(  height: 30,  ),  InputWidget(  hintText: 'Email',  obscureText: false,  controller: \_emailController,  ),  const SizedBox(  height: 20,  ),  InputWidget(  hintText: 'Password',  obscureText: true,  controller: \_passwordController,  ),  const SizedBox(  height: 30,  ),  ElevatedButton(  style: ElevatedButton.*styleFrom*(  backgroundColor: Colors.*black*,  elevation: 0,  padding: const EdgeInsets.symmetric(horizontal: 50, vertical: 15),  ),  onPressed: () {  // Add your login logic here  },  child: Text(  'Login',  style: GoogleFonts.*poppins*(  fontSize: size \* 0.040,  ),  ),  ),  ],  ),  ),  ),  );  } }  class InputWidget extends StatelessWidget {  final String hintText;  final bool obscureText;  final TextEditingController controller;   const InputWidget({  required this.hintText,  required this.obscureText,  required this.controller,  });   @override  Widget build(BuildContext context) {  return TextField(  controller: controller,  obscureText: obscureText,  decoration: InputDecoration(  hintText: hintText,  border: OutlineInputBorder(),  ),  );  } } |

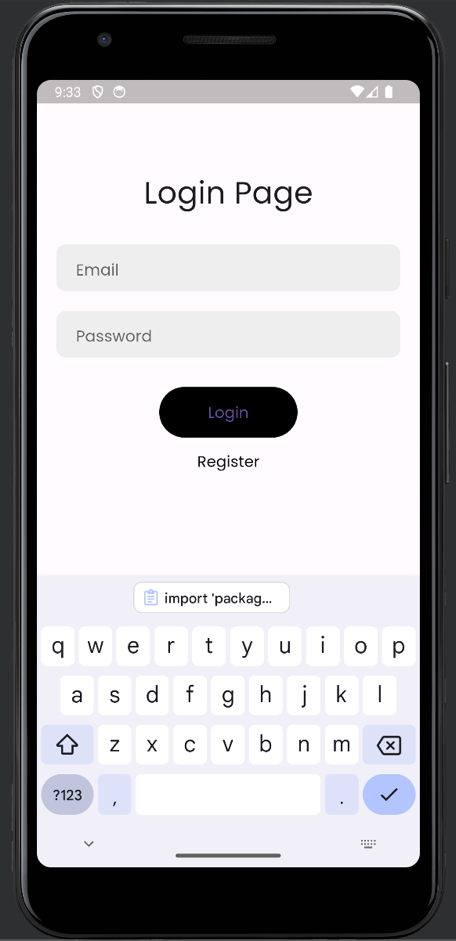
1. File add\_screen.dart

|  |
| --- |
| import 'package:flutter/material.dart'; import 'package:google\_fonts/google\_fonts.dart'; import 'package:get/get.dart'; import 'register\_page.dart'; import 'widgets/input\_widget.dart';  class LoginPage extends StatefulWidget {  const LoginPage({super.key});   @override  State<LoginPage> createState() => \_LoginPageState(); }  class \_LoginPageState extends State<LoginPage> {  final TextEditingController \_emailController = TextEditingController();  final TextEditingController \_passwordController = TextEditingController();   @override  Widget build(BuildContext context) {  var size = MediaQuery.of(context).size.width;  return Scaffold(  body: Center(  child: Padding(  padding: const EdgeInsets.symmetric(horizontal: 20.0),  child: Column(  mainAxisAlignment: MainAxisAlignment.center,  children: [  Text(  'Login Page',  style: GoogleFonts.poppins(  fontSize: size \* 0.080,  ),  ),  const SizedBox(  height: 30,  ),  InputWidget(  hintText: 'Email',  obscureText: false,  controller: \_emailController,  ),  const SizedBox(  height: 20,  ),  InputWidget(  hintText: 'Password',  obscureText: true,  controller: \_passwordController,  ),  const SizedBox(  height: 30,  ),  ElevatedButton(  style: ElevatedButton.styleFrom(  backgroundColor: Colors.black,  elevation: 0,  padding: const EdgeInsets.symmetric(horizontal: 50, vertical: 15),  ),  onPressed: () {  // Add your login logic here  },  child: Text(  'Login',  style: GoogleFonts.poppins(  fontSize: size \* 0.040,  ),  ),  ),  TextButton(  onPressed: () {  Get.to(() => const RegisterPage());  },  child: Text(  'Register',  style: GoogleFonts.poppins(  fontSize: size \* 0.040,  color: Colors.black,  ),  ),  ),  ],  ),  ),  ),  );  } } |

1. File register\_page.dart

|  |
| --- |
| import 'package:flutter/material.dart'; import 'login\_page.dart'; import 'package:get/get.dart'; import 'package:google\_fonts/google\_fonts.dart'; import 'widgets/input\_widget.dart';  class RegisterPage extends StatefulWidget {  const RegisterPage({super.key});   @override  State<RegisterPage> createState() => \_RegisterPageState(); }  class \_RegisterPageState extends State<RegisterPage> {  final TextEditingController \_emailController = TextEditingController();  final TextEditingController \_passwordController = TextEditingController();  final TextEditingController \_nameController = TextEditingController();  final TextEditingController \_usernameController = TextEditingController();   @override  Widget build(BuildContext context) {  var size = MediaQuery.of(context).size.width;  return Scaffold(  body: Center(  child: Padding(  padding: const EdgeInsets.symmetric(horizontal: 20.0),  child: Column(  mainAxisAlignment: MainAxisAlignment.center,  children: [  Text(  'Register Page',  style: GoogleFonts.poppins(  fontSize: size \* 0.080,  ),  ),  const SizedBox(  height: 30,  ),  InputWidget(  hintText: 'Name',  obscureText: false,  controller: \_nameController,  ),  const SizedBox(  height: 20,  ),  InputWidget(  hintText: 'Username',  obscureText: false,  controller: \_usernameController,  ),  const SizedBox(  height: 30,  ),  InputWidget(  hintText: 'Email',  obscureText: false,  controller: \_emailController,  ),  const SizedBox(  height: 20,  ),  InputWidget(  hintText: 'Password',  obscureText: true,  controller: \_passwordController,  ),  const SizedBox(  height: 30,  ),  ElevatedButton(  style: ElevatedButton.styleFrom(  backgroundColor: Colors.black,  elevation: 0,  padding: const EdgeInsets.symmetric(horizontal: 50, vertical: 15),  ),  onPressed: () {  // Add your register logic here  },  child: Text(  'Register',  style: GoogleFonts.poppins(  fontSize: size \* 0.040,  ),  ),  ),  TextButton(  onPressed: () {  Get.to(() => const LoginPage());  },  child: Text(  'Login',  style: GoogleFonts.poppins(  fontSize: size \* 0.040,  color: Colors.black,  ),  ),  ),  const SizedBox(  height: 20,  ),  ],  ),  ),  ),  );  } } |

Kode di atas akan membuat tampilan seperti berikut



* 1. **Push Code to GitHub** <https://github.com/adelianurlinap/Prak_PBM.git>
     1. Clone Repository yang telah dibuat :

|  |
| --- |
| git clone https://github.com/adelianurlinap/Prak\_PBM.git |

* + 1. Add dan commit

|  |
| --- |
| git add .  git commit -m "Tugas9" |

* + 1. Push code

|  |
| --- |
| git push |

1. **Kesimpulan**

Pada tugas 9 ini, telah dilakukan praktek untuk membuat frontend register dan login

1. **Referensi**

modul prak10