

Mata Kuliah : Pemrograman Mobile

Nama : Andaru Adi Wardoyo, Muhammad Zidan Rifardi

NIM : 202110370311252, 202110370311268

Kelas : 5 C

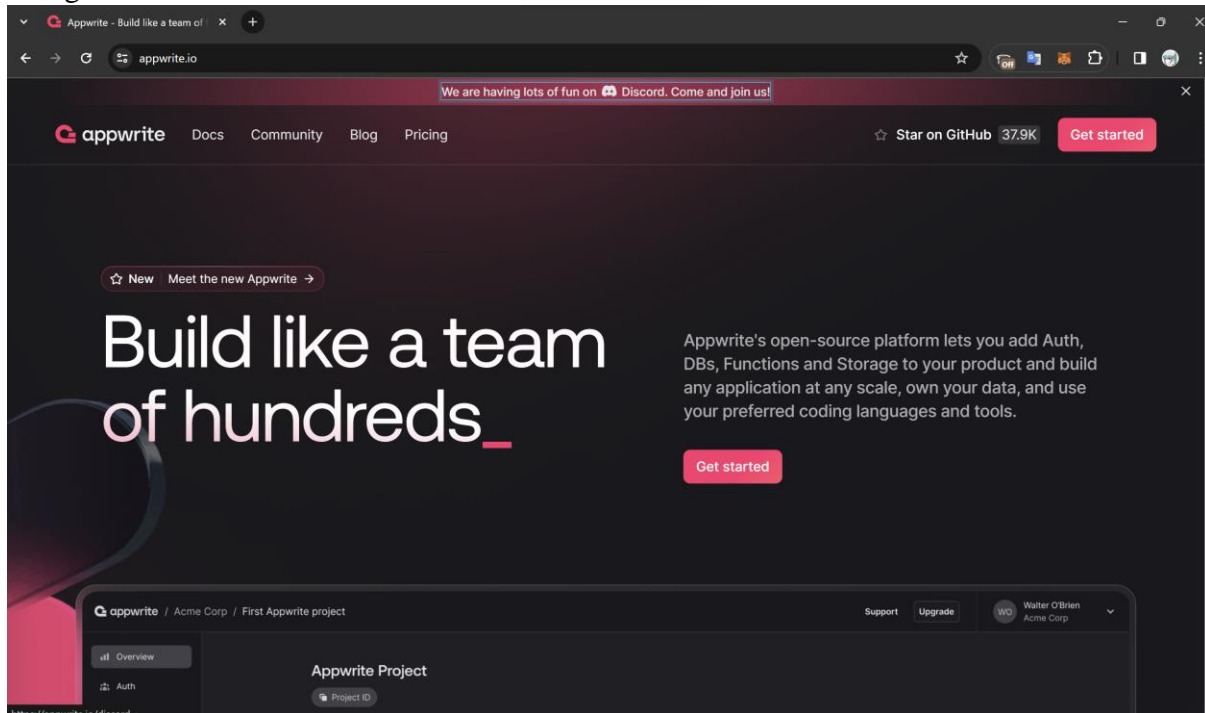
Laporan Latihan Praktikum

A. Setup dan Instalasi Appwrite

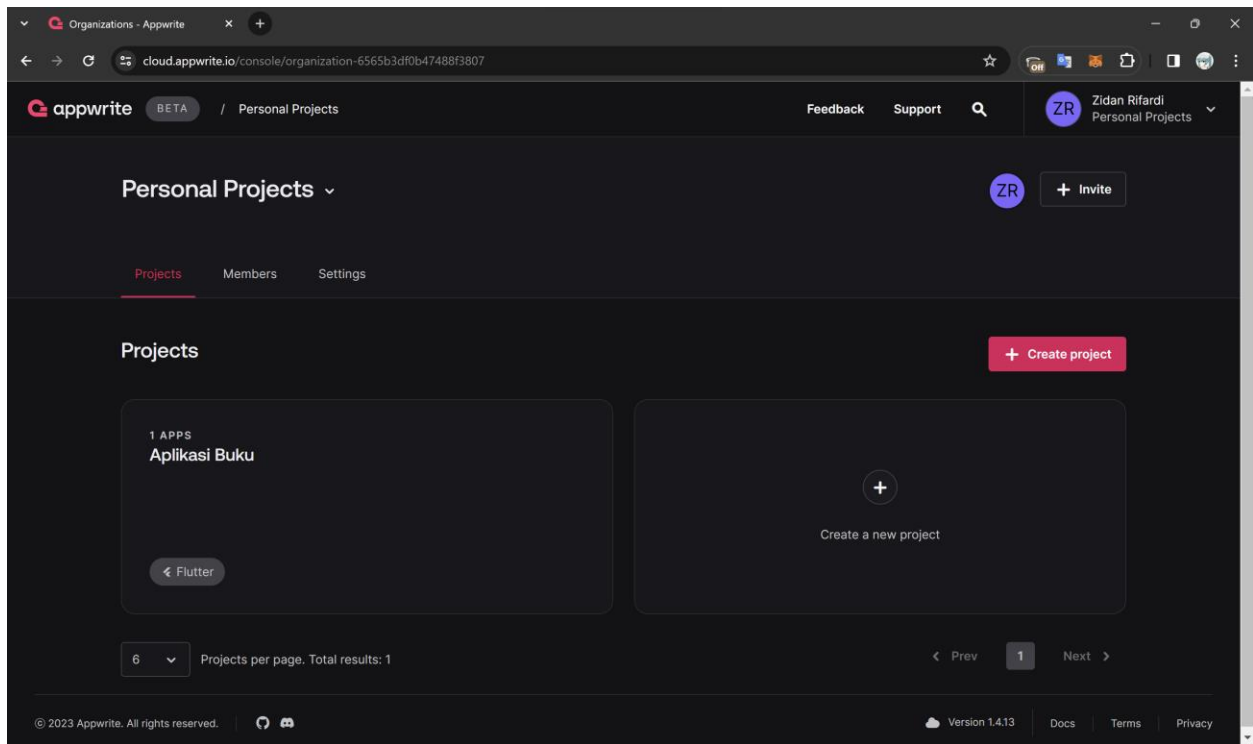
1. Lakukan setup dan instalasi Appwrite ke project Anda.
2. Boleh menggunakan Local maupun Cloud

Penjelasan :

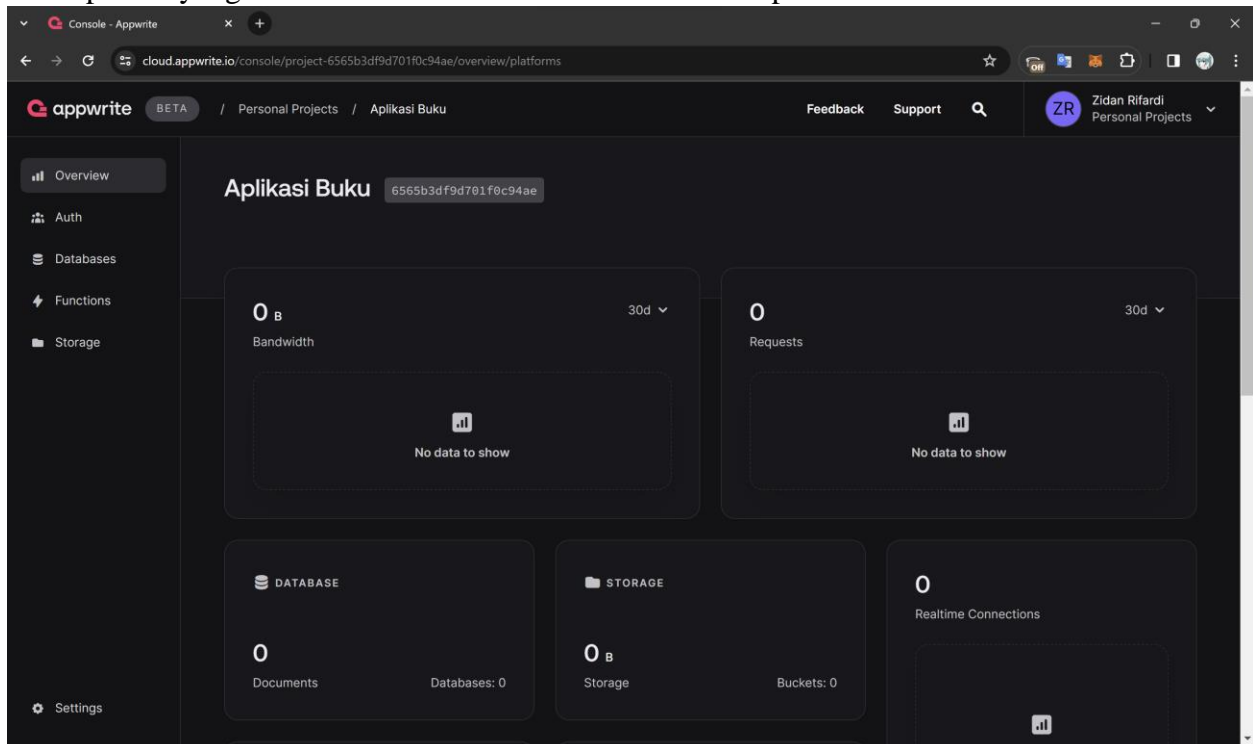
Untuk melakukan Setup Appwrite (Metode Cloud), pertama, kunjungi website appwrite.io dan klik get started.



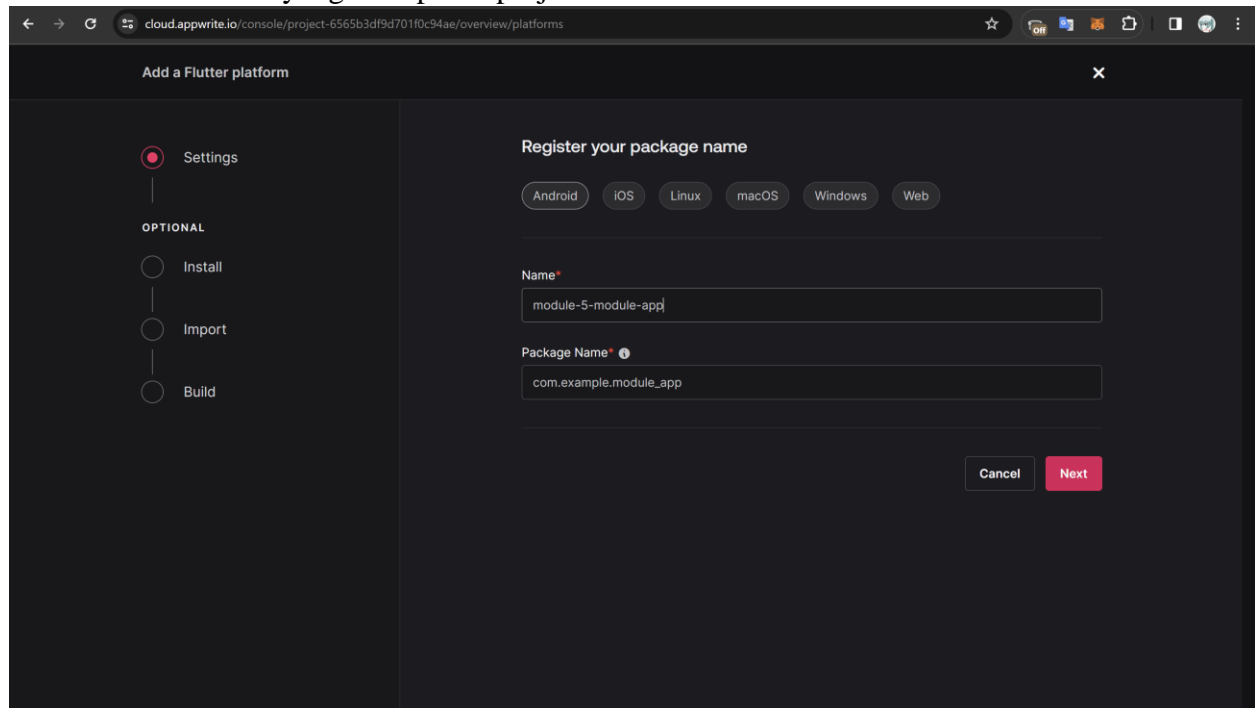
Setelah itu, lakukan login/sign up dan create app pertama kali. Jika sudah, akan masuk seperti dashboard dibawah ini.



Klik aplikasi yang telah dibuat untuk masuk ke dashboard aplikasi.



Scroll kebawah dan temukan button “add platform” lalu di klik dan pilih flutter app. Berikan nama sesuai dengan aplikasi project sendiri. Isi juga package name sesuai dengan file androidmanifest.xml yang terdapat di project sendiri.



cloud.appwrite.io/console/project-6565b3df9d701f0c94ae/overview/platforms

Add a Flutter platform

Settings

OPTIONAL

Install

Import

Build

Register your package name

Android iOS Linux macOS Windows Web

Name*

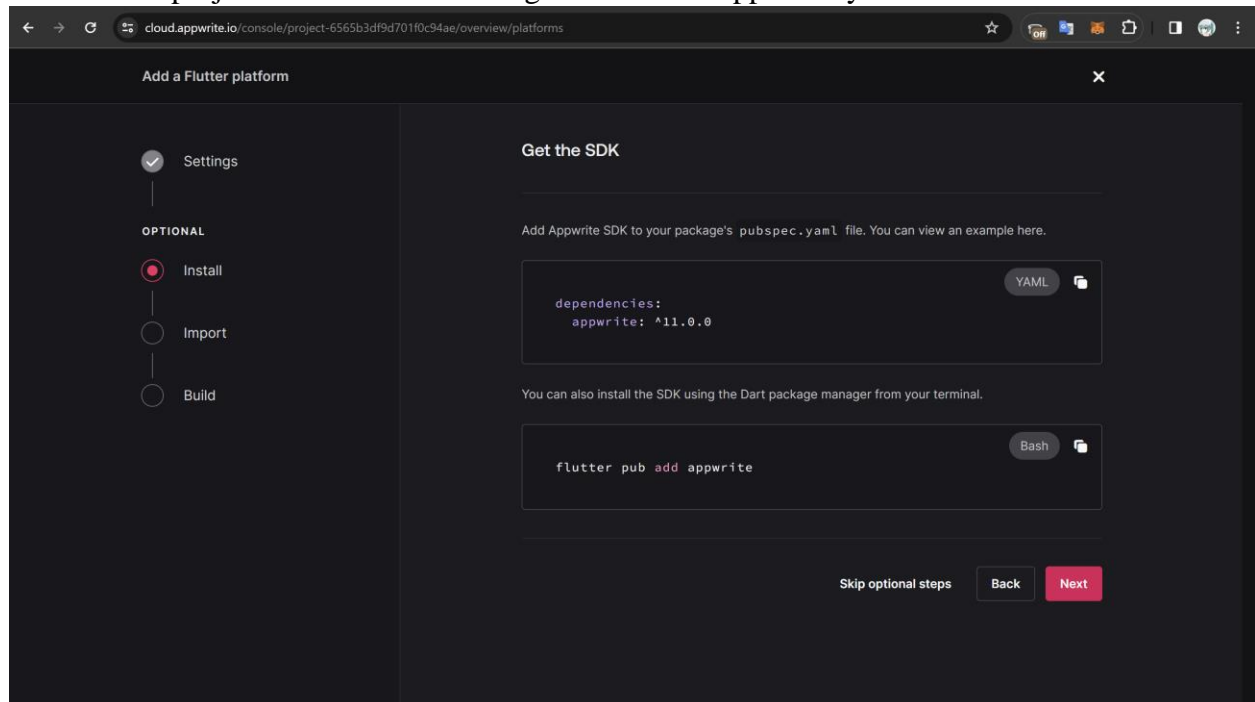
module-5-module-app

Package Name* ⓘ

com.example.module_app

Cancel Next

Masukkan dependencies appwrite kedalam file pubspec.yaml project sendiri. Lalu buat file client_controller.dart, dan masukkan kodingan controller seperti dibawah ini. Jangan lupa untuk memastikan project ID sudah sesuai dengan dashboard appwritenya.



cloud.appwrite.io/console/project-6565b3df9d701f0c94ae/overview/platforms

Add a Flutter platform

Settings

OPTIONAL

Install

Import

Build

Get the SDK

Add Appwrite SDK to your package's pubspec.yaml file. You can view an example here.

```
dependencies:  
  appwrite: ^11.0.0
```

You can also install the SDK using the Dart package manager from your terminal.

```
flutter pub add appwrite
```

Skip optional steps Back Next

```
dependencies:
  cupertino_icons: ^1.0.2
  get: 4.6.5
  image_picker: ^1.0.4
  flutter:
    sdk: flutter
#http
http: ^1.0.0
#webview
webview_flutter: ^4.4.1
appwrite: ^1.0.0
```

```
// appwrite
const endPoint = "https://cloud.appwrite.io/v1";
const projectID = "6565b3df9d701f0c94ae";
client
  .setEndpoint(endPoint)
  .setProject(projectID)
  .setSelfSigned(status: true);
```

cloud.appwrite.io/console/project-6565b3df9d701f0c94ae/overview/platforms

Add a Flutter platform

- Settings
- OPTIONAL
 - Install
 - Import
 - Build

Initialize SDK

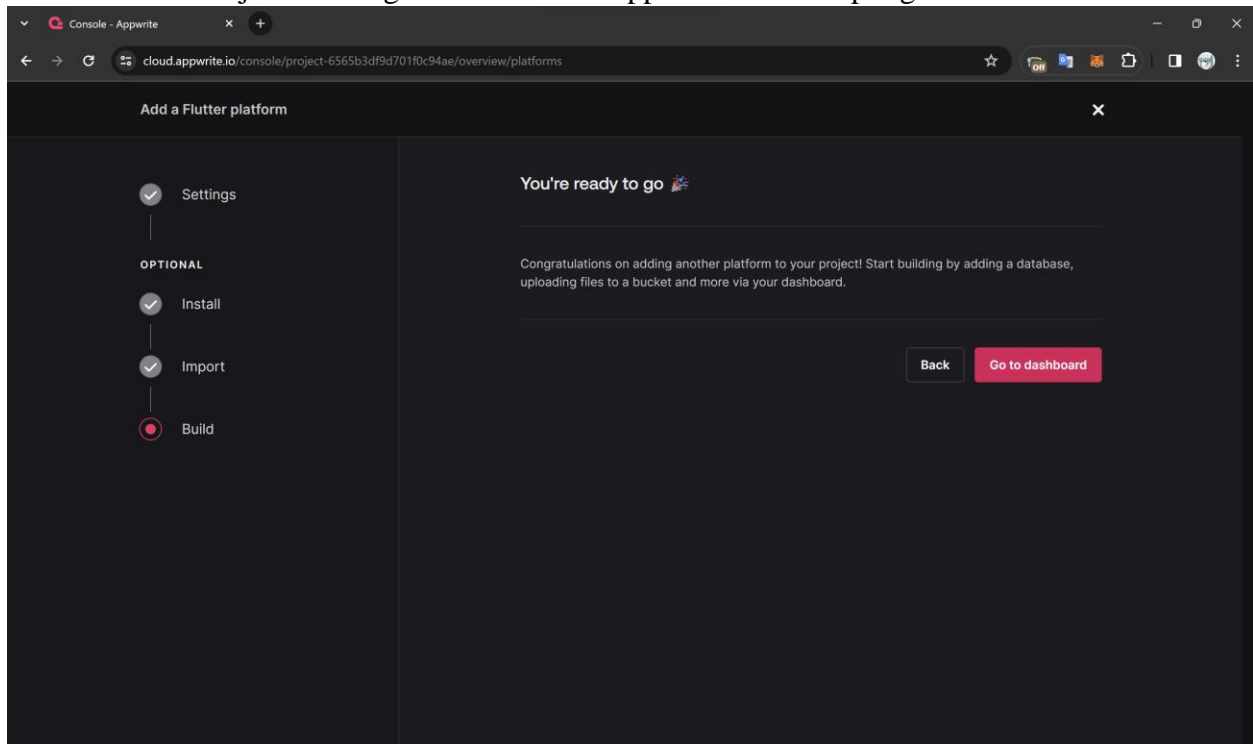
Initialize your SDK by pointing the client to your Appwrite project using your `Project ID`

```
1 import 'package:appwrite/appwrite.dart';
2
3 Client client = Client();
4 client
5   .setEndpoint('https://cloud.appwrite.io/v1')
6   .setProject('6565b3df9d701f0c94ae')
7   .setSelfSigned(status: true); // For self signed certificate
```

Before sending any API calls to your new Appwrite project, make sure your device or emulator has network access to your Appwrite project's hostname or IP address.

[Skip optional steps](#) [Back](#) [Next](#)

Setelah itu next saja dan klik go to dashboard. Appwrite sudah siap digunakan :D



B. Latihan Melengkapi Tutorial Diatas

1. Lengkapi code-code dari penjelasan materi diatas.
2. Buatlah implementasi sesuai case aplikasi masing-masing.
3. Buatlah screen page untuk satu atau lebih penggunaan service.

Membuat controller untuk Appwrite create account

```
import 'package:appwrite/appwrite.dart';
import 'package:flutter/material.dart';
import 'package:flutter/widgets.dart';
import 'package:get/get.dart';

import '../pages/login/userlogin_page.dart';
import 'client_controller.dart';

class AccountController extends ClientController {
  RxBool isLoading = false.obs;
  Account? account;
  @override
  void onInit() {
    super.onInit();
    // appwrite
    account = Account(client);
  }

  Future createAccount(String email, String password, String name) async {
    try {
      isLoading.value = true;
      final result = await account!.create(
        userId: ID.unique(),
        email: email,
        password: password,
        name: name,
      );
      Get.snackbar('Success', 'Registration successful',
        backgroundColor: Colors.green);
      Get.off(userLoginPage()); //Navigate ke Login Page
    } catch (error) {
      Get.snackbar('Error', 'Registration failed: $error',
        backgroundColor: Colors.red);
    } finally {
      isLoading.value = false;
    }
  }
}
```

Membuat view tampilan pada aplikasi dan tambahkan textfield untuk menerima inputan dengan code berikut

```
Container(
  width: 360,
  height: 60,
  padding: EdgeInsets.fromLTRB(24, 6, 20, 0),
  decoration: BoxDecoration(
    borderRadius: BorderRadius.circular(30),
    color: const Color(0xFFD4D4D4), // BoxDecoration
  ),
  child: TextField(
    controller: _emailController,
    decoration: InputDecoration(
      hintText: "Email", border: InputBorder.none), // InputDecoration
  ), // TextField
), // Container

SizedBox(
  height: 20,
), // SizedBox

Container(
  width: 360,
  height: 60,
  padding: EdgeInsets.fromLTRB(24, 6, 20, 0),
  decoration: BoxDecoration(
    borderRadius: BorderRadius.circular(30),
    color: const Color(0xFFD4D4D4), // BoxDecoration
  ),
  child: TextField(
    controller: _passwordController,
    obscureText: true,
    decoration: InputDecoration(
      hintText: "Password", border: InputBorder.none), // InputDecoration
  ), // TextField
), // Container

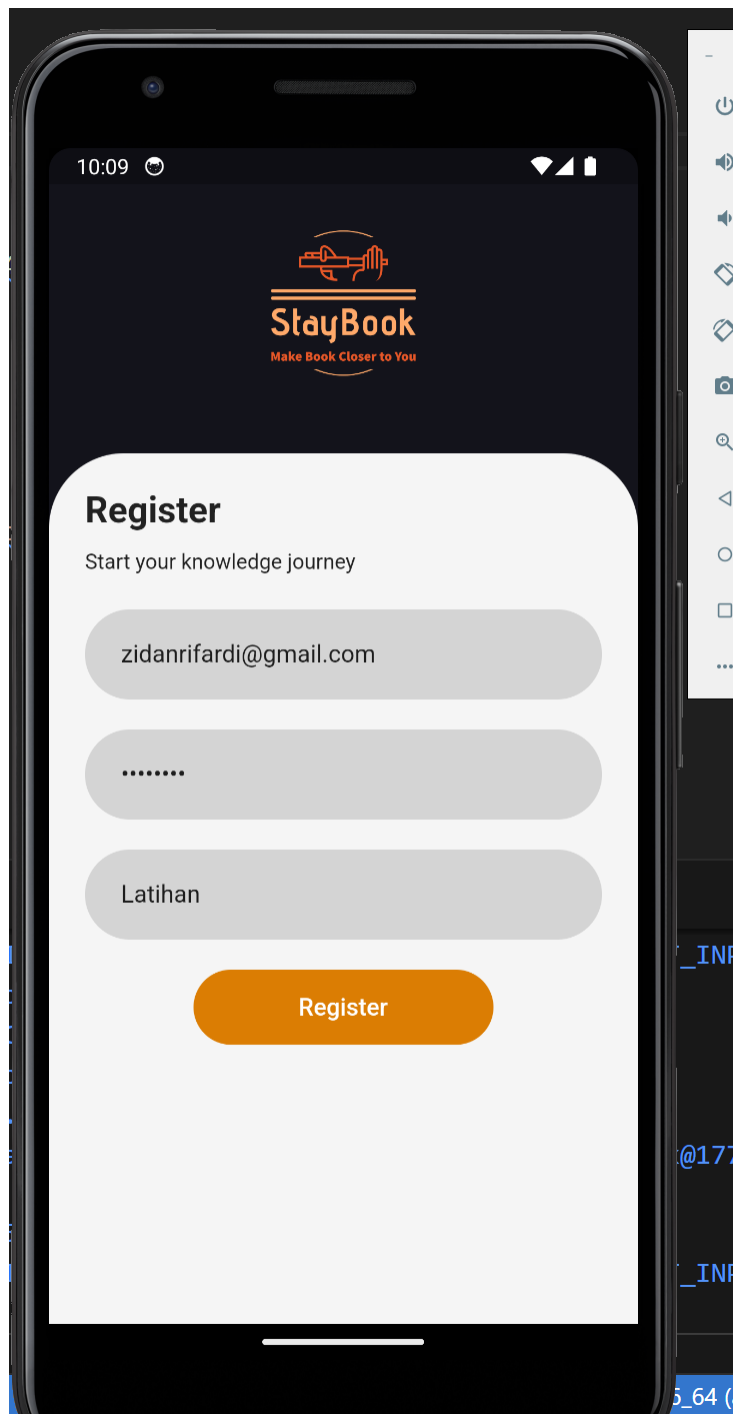
SizedBox(
  height: 20,
), // SizedBox

Container(
  width: 360,
  height: 60,
  padding: EdgeInsets.fromLTRB(24, 6, 20, 0),
  decoration: BoxDecoration(
    borderRadius: BorderRadius.circular(30),
    color: const Color(0xFFD4D4D4), // BoxDecoration
  ),
  child: TextField(
    controller: _nameController,
    decoration: InputDecoration(
      hintText: "Name", border: InputBorder.none), // InputDecoration
  ), // TextField
), // Container
```

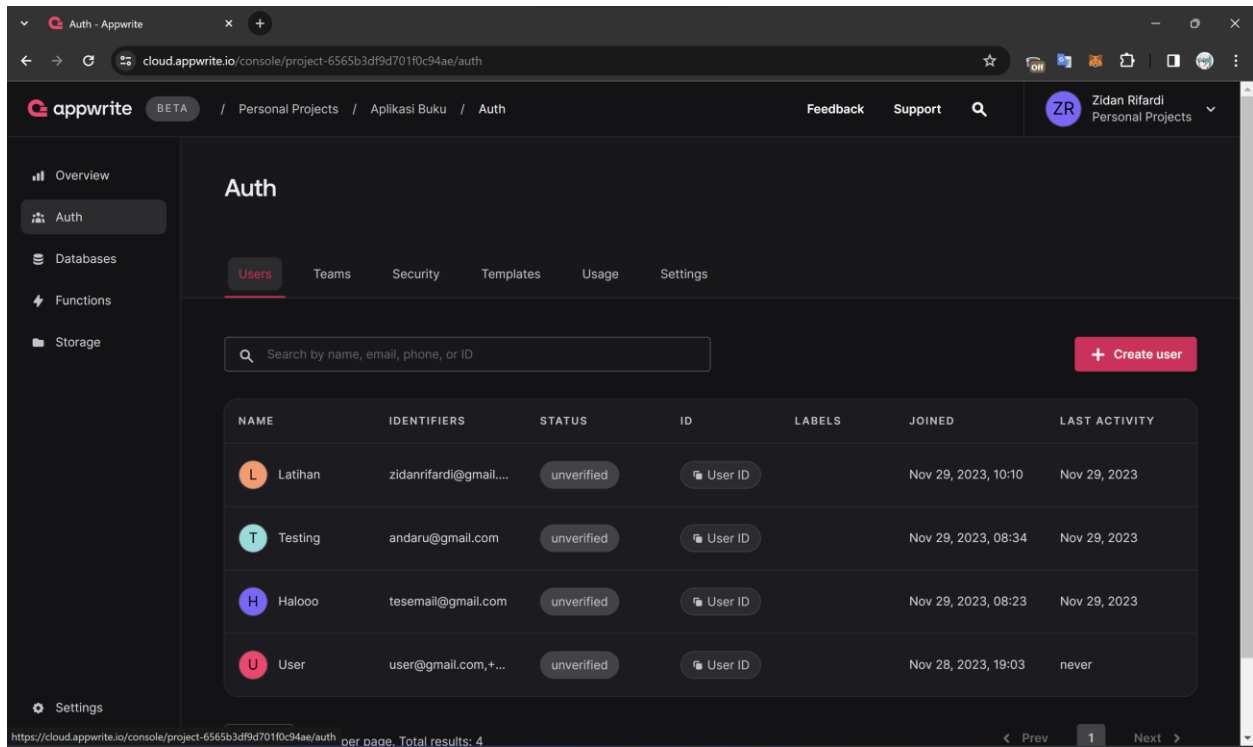
Gunakan text button untuk memanggil controller create account yang telah dibuat sebelumnya

```
), // SizedBox
    SizedBox(
      width: 200,
      height: 50,
      child: TextButton(
        style: TextButton.styleFrom(
          backgroundColor: Color.fromARGB(255, 219, 125, 3),
          shape: RoundedRectangleBorder(
            borderRadius: BorderRadius.circular(30),
          ), // RoundedRectangleBorder
        ),
        onPressed: _appwriterController.isLoading.value
          ? null
          : () {
              _appwriterController.createAccount(
                _emailController.text,
                _passwordController.text,
                _nameController.text
              );
            },
        child: const Text(
          'Register',
          style: TextStyle(
            fontSize: 16,
            color: Colors.white,
          ), // TextStyle
        ), // Text
      ), // TextButton
    ), // SizedBox
```


Aplikasi siap untuk digunakan dan akan berjalan secara normal.



Akun yang telah dibuat dapat dilihat didalam dashboard appwrite pada bagian auth.



The screenshot shows the Appwrite Auth dashboard. The left sidebar contains navigation links: Overview, Auth (selected), Databases, Functions, Storage, and Settings. The main content area is titled 'Auth' and has tabs for Users, Teams, Security, Templates, Usage, and Settings. The 'Users' tab is active, displaying a search bar and a '+ Create user' button. Below is a table of users with columns: NAME, IDENTIFIERS, STATUS, ID, LABELS, JOINED, and LAST ACTIVITY. The table lists four users: Latihan, Testing, Halooo, and User, all with 'unverified' status. The footer shows the URL 'https://cloud.appwrite.io/console/project-6565b3df9d7010c94ae/auth' and pagination information: 'per page. Total results: 4'.

NAME	IDENTIFIERS	STATUS	ID	LABELS	JOINED	LAST ACTIVITY
L Latihan	zidanrifardi@gmail....	unverified	User ID		Nov 29, 2023, 10:10	Nov 29, 2023
T Testing	andaru@gmail.com	unverified	User ID		Nov 29, 2023, 08:34	Nov 29, 2023
H Halooo	tesemail@gmail.com	unverified	User ID		Nov 29, 2023, 08:23	Nov 29, 2023
U User	user@gmail.com, +...	unverified	User ID		Nov 28, 2023, 19:03	never