# PENDIDIKAN AGAMA ISLAM

# “AGAMA DAN KEBAHAGIAAN MANUSIA”



Tugas ini dibuat untuk memenuhi Nilai Tugas Makalah

Matakuliah Pendidikan Agama Islam

Nama Pembuat :

*Siti Bakhiroh Dzakwani (64231245)*

**PROGRAM STUDI MANAJEMEN KAMPUS KRAMAT 98**

**UNIVERSITAS BINA SARANA INFORMATIKA**

**JAKARTA**

**2024**

# KATA PENGANTAR

Puji syukur kehadirat Allah SWT, yang telah melimpahkan rahmat, taufik serta hidayah-Nya, sehingga penulis dapat  menyelesaikan tugas materi Pendidikan Agama Islam dengan baik. Adapun judul tugas Pendidikan Agama Islam  kami adalah ***“ AGAMA DAN KEBAHAGIAAN MANUSIA ”.***

Kami menyadari bahwa masih ada kekurangan baik dari segi susunan kalimat dalam makalah ini. Oleh karena itu kami menerima segala saran dan kritik dari pembaca agar kami dapat memperbaiki makalah ini. Kami berharap semoga makalah ini yaitu tentang Pendidikan Agama Islam ini dapat memberikan manfaat terhadap pembaca.

# Jakarta,09 Januari 2024

Siti Bakhiroh Dzakwani

# DAFTAR ISI

**KATA PENGANTAR**

**DAFTAR ISI**

**DAFTAR SIMBOL**

**BAB I PENDAHULUAN**

* 1. Latar Belakang Masalah
  2. Maksud dan Tujuan
  3. Ruang Lingkup

**BAB II LANDASAN TEORI**

* + - 1. Pengertian Flutter
      2. Kelebihan dan Kekurangan Flutter
      3. Pengertian Restfull API
      4. Pengertian Flowchart

**BAB III PENUTUP**

# DAFTAR SIMBOL

* Terminal

Simbol terminal adalah simbol oval yang menunjukkan awal atau akhir dari suatu proses.

* Input  
  Simbol yang menunjukkan input atau masukan yang diperoleh dari luar sistem atau proses.
* Process

Simbol kotak atau proses, digunakan untuk menandai langkah atau tindakan dalam proses tersebut

* Decision

Simbol berbentuk belah ketupat atau decision, digunakan untuk menandai keputusan yang harus diambil dalam proses tersebut.

* Flow Line

Simbol ini membantu dalam menyajikan alur proses secara visual dan memudahkan pemahaman tentang bagaimana setiap simbol dan proses saling terkait dan berinteraksi satu sama lain.

# PENDAHULUAN

## Latar Belakang Masalah

## Latar belakang masalah untuk pembuatan aplikasi kebun binatang dapat dipahami melalui beberapa aspek yang mencerminkan kebutuhan dan manfaat pengelola kebun binatang,.Aplikasi kebun binatang dapat meningkatkan pengalaman pengelola dengan menyediakan informasi interaktif tentang setiap spesies yang ada di kebun binatang. Pengelola dapat dengan mudah mengakses informasi mengenai habitat dan juga pegawai. Aplikasi kebun binatang memberikan kemudahan akses informasi bagi pengelola yang agar membuat efesien. Pembuatan aplikasi kebun binatang juga mencerminkan penerapan teknologi dan inovasi. pembuatan aplikasi kebun binatang diharapkan dapat memberikan manfaat yang signifikan bagi pengelola kebun binatang.

## Maksud dan Tujuan

Maksud dari Aplikasi tersebut yaitu :

* Pendidikan dan Informasi:

Aplikasi kebun binatang dapat menjadi sarana edukatif yang efektif untuk memberikan informasi tentang berbagai jenis hewan, habitat.

* Kesadaran Lingkungan:

Dengan menyajikan informasi tentang kepunahan dan ancaman terhadap beberapa spesies, aplikasi ini dapat meningkatkan kesadaran pengguna tentang pentingnya konservasi alam dan perlindungan satwa liar.

* Aksesibilitas Global:

Dapat memperluas cakupan pendidikan dan kesadaran lingkungan ke tingkat global, sehingga mendorong partisipasi dan dukungan dari berbagai lapisan masyarakat.

Tujuan dari Aplikasi tersebut yaitu :

* Promosi dan Pemasaran:

Aplikasi dapat digunakan sebagai alat promosi untuk menarik lebih banyak pengunjung. Informasi dapat disampaikan melalui aplikasi untuk meningkatkan ketertarikan masyarakat.

* Aksesibilitas dan Keterjangkauan:

Aplikasi dapat membuat informasi tentang kebun binatang lebih mudah diakses oleh para staff pengelola kebun binatang. Ini dapat meningkatkan keterjangkauan efesiensi waktu

## Ruang Lingkup

* Pemberian informasi kepada pengelola.
* Mengembangan strategi pemasaran untuk meningkatkan kunjungan.
* Penggunaan teknologi untuk meningkatkan efisiensi operasional.
* Inisiatif konservasi dan edukasi untuk meningkatkan kesadaran lingkungan.

# Landasan Teori

## Pengertian Flutter

Flutter adalah platform yang digunakan para developer untuk membuat aplikasi multiplatform hanya dengan satu basis coding (codebase). Artinya, aplikasi yang dihasilkan dapat dipakai di berbagai platform, baik mobile Android, iOS, web, maupun desktop. Sebelum resmi diluncurkan pada Desember 2018, Flutter juga merupakan platform yang gratis dan open source. Jika Anda ingin menggunakan Flutter, Anda perlu mempelajari bahasa pemrograman Dart. Berbeda dengan framework front-end pada umumnya, di mana JavaScript adalah bahasa pemrogramannya.Flutter sebenarnya telah dikembangkan oleh Google sejak 2015. Lalu, di tahun 2019, popularitas Flutter mulai meroket dan banyak developer berbondong-bondong menggunakan Flutter.

## Kelebihan dan Kekurangan Flutter

flutter memiliki kelebihan yang menjadi alasan tersebut di atas. Kelebihan flutter adalah:

* Mengembangkan Aplikasi Bisa Lebih Mudah dan Lebih Cepat

Fitur yang ada pada Flutter framework bisa membantu proses pengembangan aplikasi lebih mudah dan tentu saja lebih cepat.Hal ini karena Flutter menawarkan user interface widgets yang telah dikembangkan oleh Google dan memiliki kualitas coding yang mumpuni.

* Biaya yang Dikeluarkan Lebih Hemat

Karena flutter adalah framework untuk pembuatan aplikasi mobile multiplatform, artinya pengembangan aplikasi lebih efisien. Developer tidak perlu membuat aplikasi lagi untuk ditempatkan pada platform lain yang berbeda.Adanya fitur-fitur seperti fitur UI widget, Hot Reload, data backend, aksesibilitas, platform integration, packages dan plugin, memudahkan developer dalam pembuatan aplikasi. Sehingga, mereka bisa lebih menghemat pengeluaran tenaga, waktu, serta biaya.

* Performa Tidak Kalah dari Aplikasi Native

Meskipun sifat dari Flutter adalah apikasi multiplatform, performanya yang dihasilkan seperti aplikasi native. Hal ini karena flutter menggunakan kombimasi kode dari mesin render canggih dan aplikasi native. Sehingga hasilnya bisa tetap baik meskipun desainnya ditampilkan untuk berbagai platform.

kekurangan flutter :

* Library dari Pihak Ketiga Belum Banyak

Flutter termasuk teknologi terobosan baru yang semakin populer dari waktu ke waktu. Karena masih baru, developer yang menggunakannya masih berkembang. Sehingga, library pihak ketiga atau resource code yang ada masih belum banyak.

* Output Aplikasi Berukuran Besar

Kekurangan ini tentu membuat ruang penyimpanan dari perangkat pengguna lebih cepat habis. Performa aplikasi juga bisa terpengaruh apabila ruang penyimpanan perangkat sangat terbatas. Bisa saja kemudian pengguna tidak jadi mendownload aplikasi tersebut.

* Bahasa Pemrograman yang Baru Perlu Dipelajari

Karena Flutter adalah framework baru, Anda sebagai developer juga perlu belajar bahasa pemrograman baru yang bernama Dart. Kode Dart ini belum banyak populer meskipun sudah semakin banyak dimanfaatkan developer dalam pengembangan aplikasi

## Pengertian Restfull API

Apa itu Rest API? Rest API (Representational State Transfer Application Programming Interface) adalah jenis API layanan web yang menggunakan permintaan HTTP untuk melakukan operasi CRUD (Buat, Baca, Perbarui, Hapus) pada data. Ini adalah gaya arsitektur populer yang digunakan untuk membangun sistem terdistribusi di web, memungkinkan pengembang perangkat lunak untuk berinteraksi dengan aplikasi web secara terprogram.

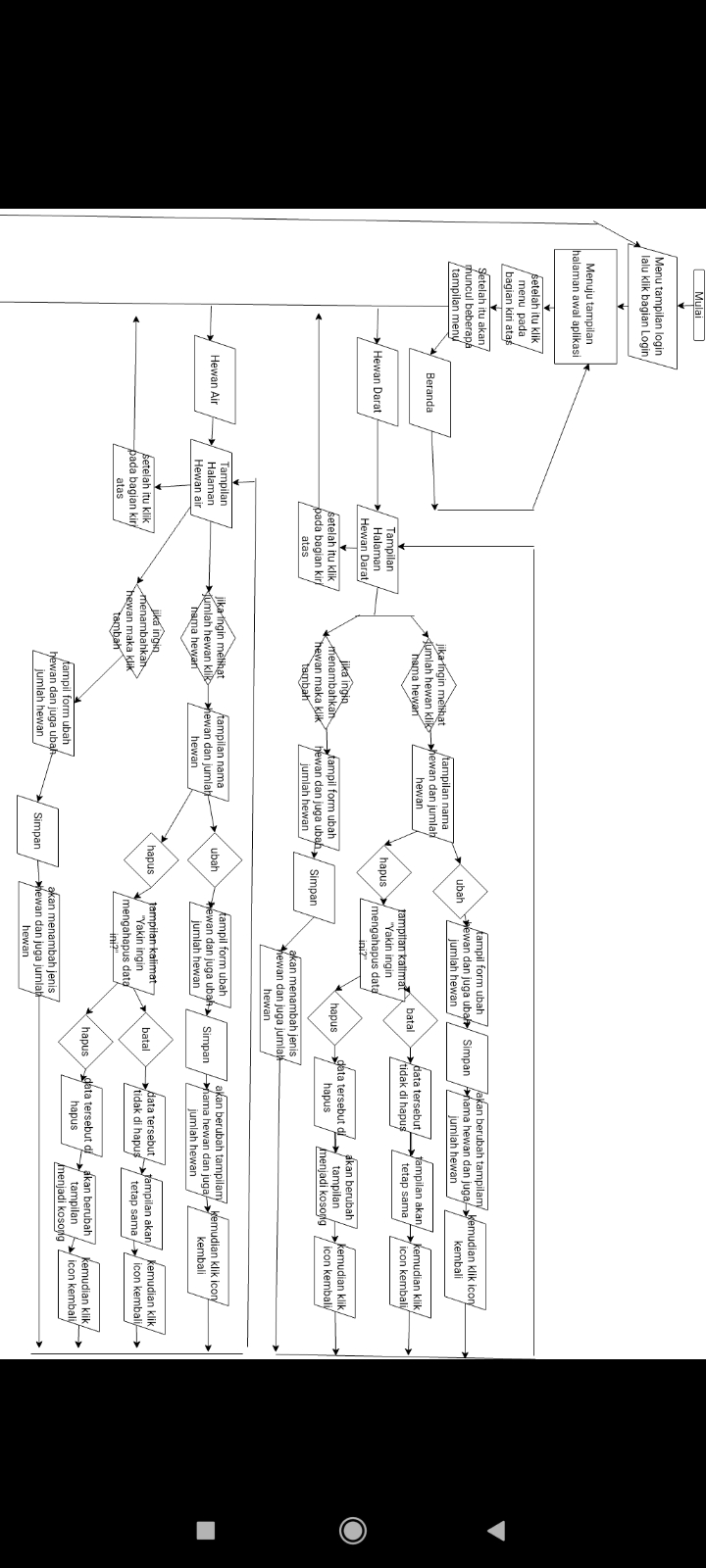
Rest API dirancang agar sederhana dan fleksibel sehingga, memungkinkan pengembang membangun aplikasi yang dapat bekerja di berbagai bahasa dan kerangka kerja pemograman.

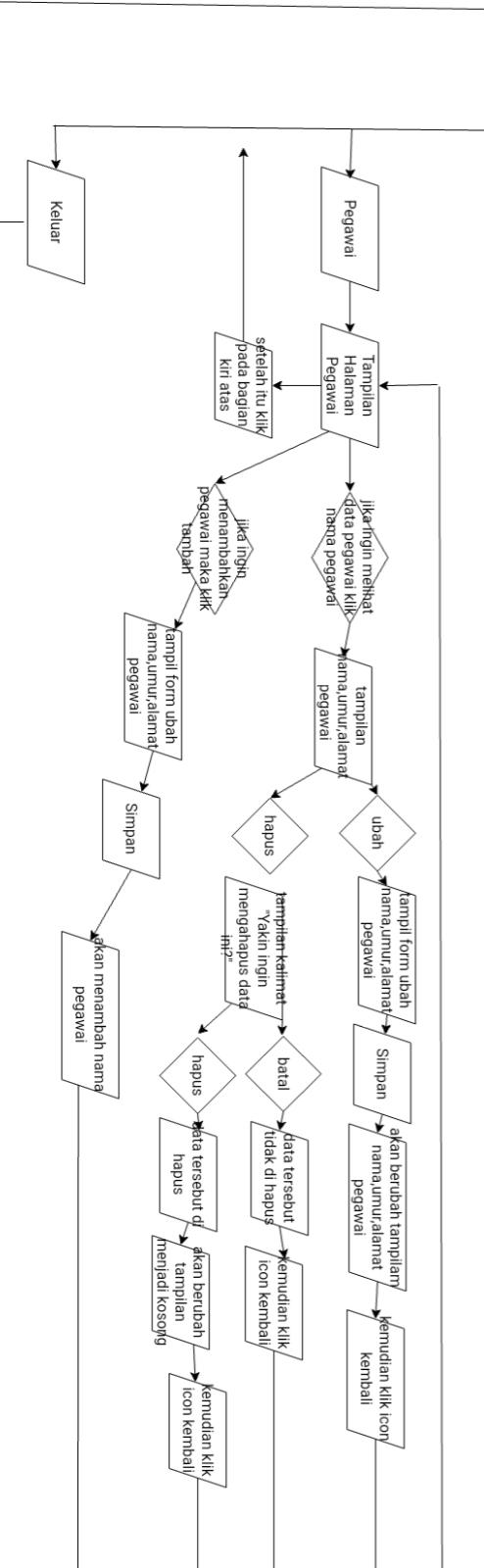
## Pengertian Flowchart

Flowchart atau bagan alur adalah diagram yang menampilkan langkah-langkah dan keputusan untuk melakukan sebuah proses dari suatu program. Setiap langkah digambarkan dalam bentuk diagram dan dihubungkan dengan garis atau arah panah. Flowchart berperan penting dalam memutuskan sebuah langkah atau fungsionalitas dari sebuah proyek pembuatan program yang melibatkan banyak orang sekaligus. Selain itu dengan menggunakan bagan alur proses dari sebuah program akan lebih jelas, ringkas, dan mengurangi kemungkinan untuk salah penafsiran.

# PERANCANGAN DAN IMPLEMENTASI

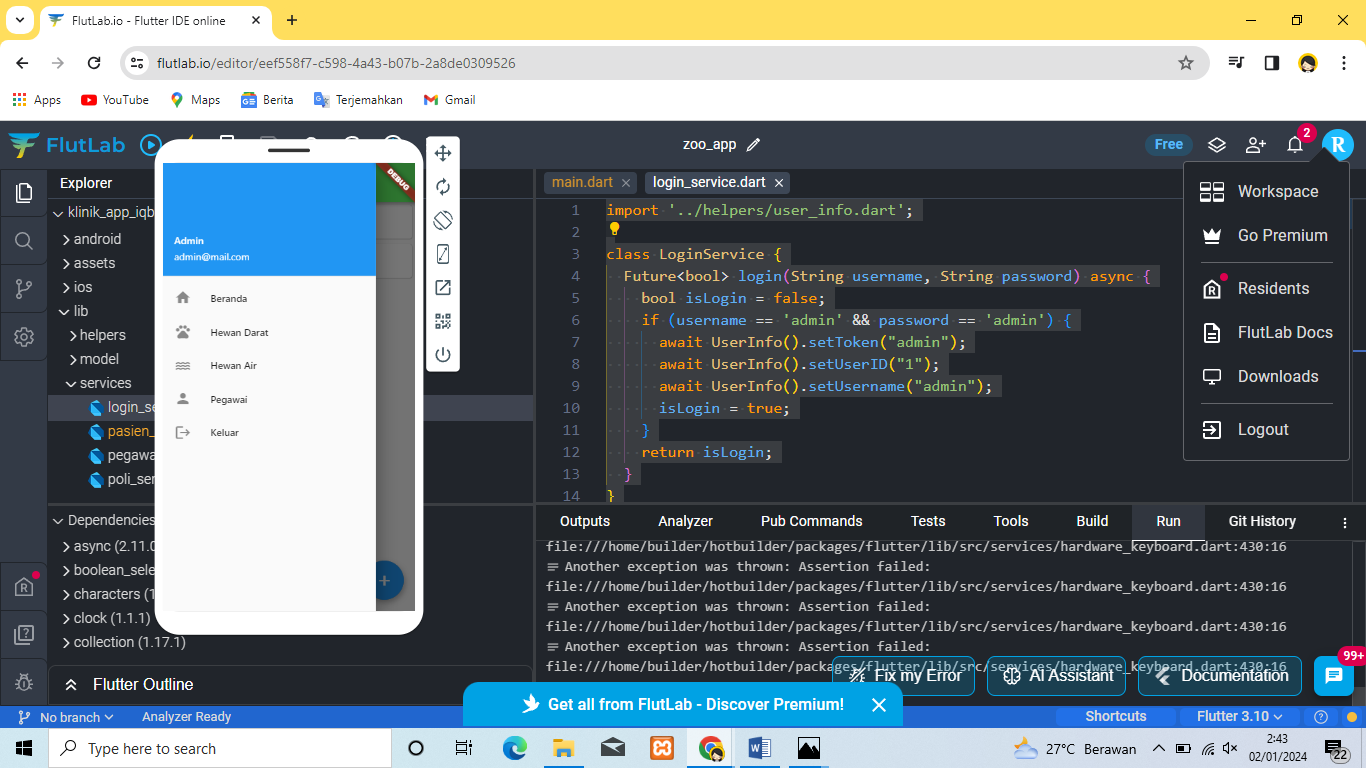
## Rancangan Flowchart Aplikasi



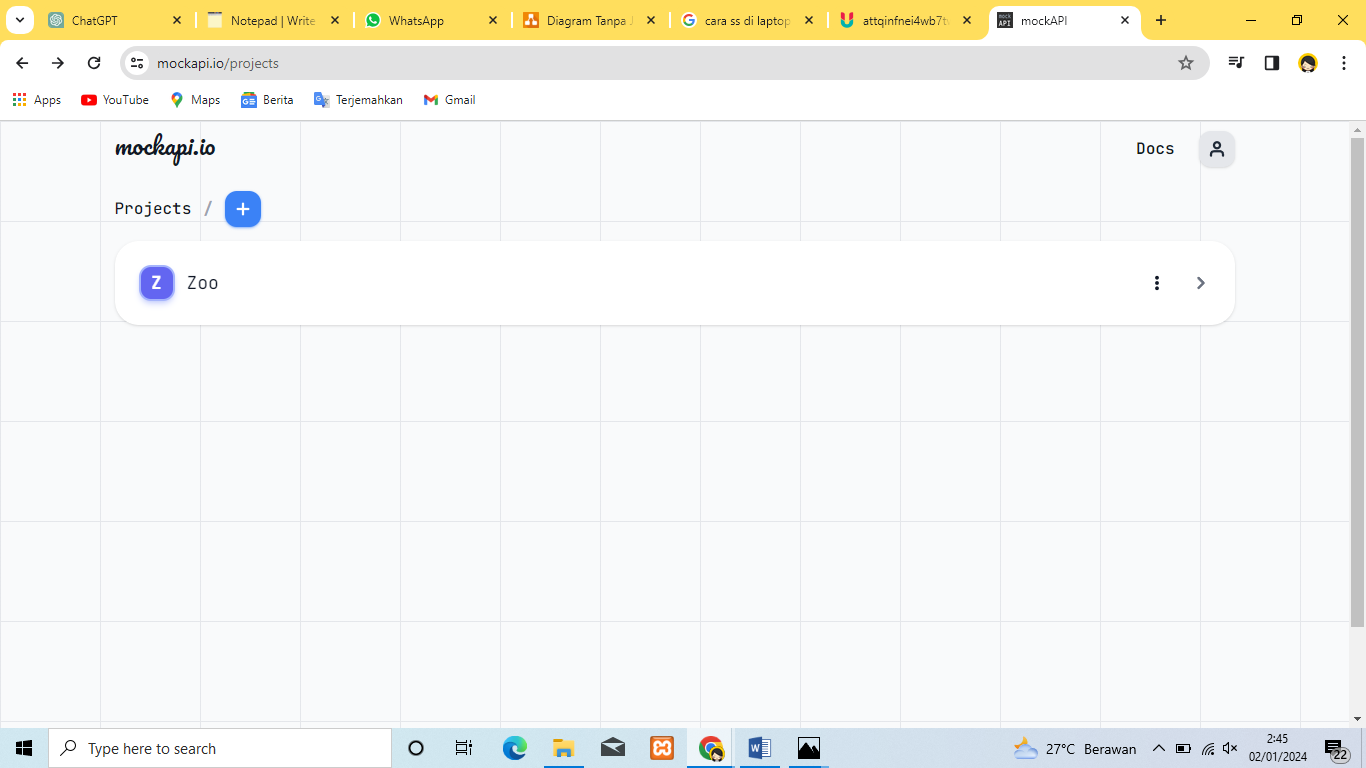


## Pembuatan Back-End Aplikasi

### Spesifikasi API



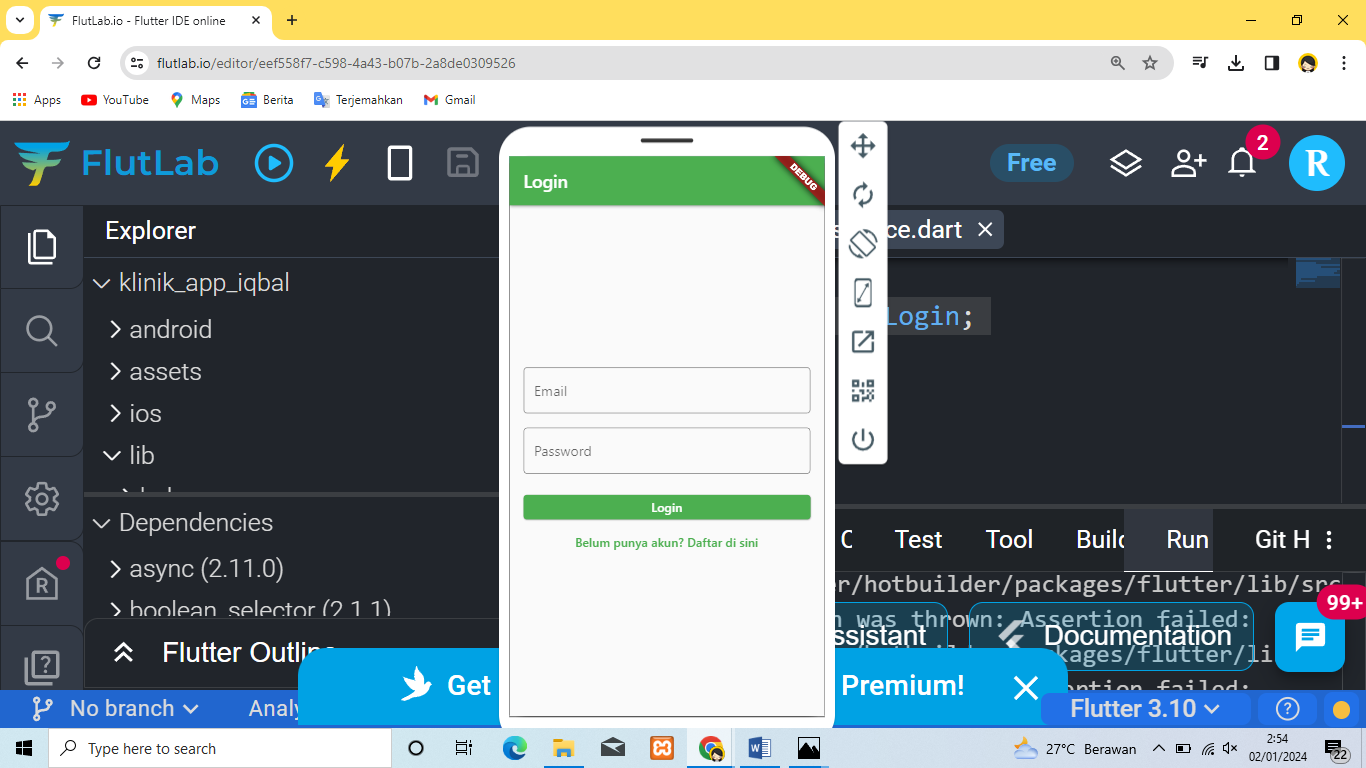
### Spesifikasi Basis Data



Pembuatan Front-End Aplikasi

### Tampilan Aplikasi

* Tampilan Menu Registrasi



Lengkapi dengan alur penggunaan aplikasinya, contoh:  
1. Untuk registrasi, masukkan Email dan Password (bias juga langsung dengan mengklik tombil login saja)

2.klik login

3.masuk kehalaman beranda aplikasi zoo

4.klik tanda strip 3 di atas kiri aplikasi

5.terdapat berbagai macam menu seperti hewan darat,air dan juga tentang pegawai.

6.klik salah satu untuk melihat informasi apa saja yg terdapat di dalam menu tersebut

**7.**Jika sudah klik strip 3 di atas kiri aplikasi dan pilih menu yang ingin di klik

### Source Code

* Source Code main.dart

Source Code main.dart ini digunakan untuk menjalankan keseluruhan codingan

import 'package:flutter/material.dart';

import '/helpers/user\_info.dart';

import '/ui/beranda.dart';

import '/ui/login.dart';

import '/ui/splashscreen.dart';

Future<void> main() async {

  WidgetsFlutterBinding.ensureInitialized();

  var token = await UserInfo().getToken();

  await UserInfo().logout();

  print(token);

  runApp(MaterialApp(

    title: "Klinik APP",

    debugShowCheckedModeBanner: false,

    home: token == null ? SplashScreen() : Login(),

  ));

}

* Source Code login.dart

Source Code main.dart ini digunakan untuk menjalankan tampilan login

* import 'package:flutter/material.dart';
* import 'package:klinik\_app\_iqbal/ui/beranda.dart';
* void main() {
* runApp(Login());
* }
* class User {
* final String email;
* final String password;
* User(this.email, this.password);
* }
* class Login extends StatelessWidget {
* @override
* Widget build(BuildContext context) {
* return MaterialApp(
* title: 'Login & Register',
* theme: ThemeData(
* primarySwatch: Colors.green,
* ),
* home: LoginPage(),
* );
* }
* }
* @override
* Widget build(BuildContext context) {
* return Scaffold(
* appBar: AppBar(
* title: Text('Login'),
* ),
* body: Padding(
* padding: EdgeInsets.all(16.0),
* child: Column(
* mainAxisAlignment: MainAxisAlignment.center,
* crossAxisAlignment: CrossAxisAlignment.stretch,
* children: <Widget>[
* TextFormField(
* controller: emailController,
* decoration: InputDecoration(
* labelText: 'Email',
* border: OutlineInputBorder(),
* ),
* ),
* SizedBox(height: 16.0),
* TextFormField(
* controller: passwordController,
* obscureText: true,
* decoration: InputDecoration(
* labelText: 'Password',
* border: OutlineInputBorder(),
* ),
* ),
* SizedBox(height: 24.0),
* ElevatedButton(
* onPressed: () {
* \_login(context);
* },
* child: Text('Login'),
* ),
* SizedBox(height: 12.0),
* TextButton(
* onPressed: () {
* Navigator.push(
* context,
* MaterialPageRoute(builder: (context) => RegisterPage()),
* );
* },
* child: Text('Belum punya akun? Daftar di sini'),
* ),
* ],
* ),
* ),
* );
* }
* }
* class RegisterPage extends StatelessWidget {
* final TextEditingController emailController = TextEditingController();
* final TextEditingController passwordController = TextEditingController();
* void \_register(BuildContext context) {
* final enteredEmail = emailController.text;
* final enteredPassword = passwordController.text;
* User newUser = User(enteredEmail, enteredPassword);
* showDialog(
* context: context,
* builder: (context) => AlertDialog(
* title: Text('Registrasi Berhasil'),
* content: Text('Registrasi berhasil! Silakan login.'),
* actions: [
* TextButton(
* onPressed: () {
* Navigator.push(
* context,
* MaterialPageRoute(builder: (context) => Login()),
* );
* },
* child: Text('OK'),
* ),
* ],
* ),
* );
* }
* Source Code sidebar.dart

Source Code sidebar.dart ini digunakan untuk menyatukan halaman Manajemen (CRUD)

import 'package:flutter/material.dart';

import 'package:klinik\_app\_iqbal/ui/beranda.dart';

import 'package:klinik\_app\_iqbal/ui/login.dart';

import 'package:klinik\_app\_iqbal/ui/pasien/pasien\_page.dart';

import 'package:klinik\_app\_iqbal/ui/pegawai/pegawai\_page.dart';

import 'package:klinik\_app\_iqbal/ui/poli/poli\_page.dart';

class Sidebar extends StatelessWidget {

  const Sidebar({Key? key}) : super(key: key);

  @override

  Widget build(BuildContext context) {

    return Drawer(

      child: ListView(

        padding: EdgeInsets.zero,

        children: [

          UserAccountsDrawerHeader(

            accountName: Text("Admin"),

            accountEmail: Text("admin@mail.com"),

          ),

          ListTile(

            leading: Icon(Icons.home),

            title: Text("Beranda"),

            onTap: () {

              Navigator.pushReplacement(

                  context,

                  MaterialPageRoute(

                    builder: (context) => Beranda(),

                  ));

            },

          ),

          ListTile(

            leading: Icon(Icons.pets),

            title: Text("Hewan Darat"),

            onTap: () {

              Navigator.pushReplacement(

                  context,

                  MaterialPageRoute(

                    builder: (context) => PoliPage(),

                  ));

            },

          ),

          ListTile(

            leading: Icon(Icons.water),

            title: Text("Hewan Air"),

            onTap: () {

              Navigator.pushReplacement(

                  context,

                  MaterialPageRoute(

                    builder: (context) => PasienPage(),

                  ));

            },

          ),

          ListTile(

            leading: Icon(Icons.person),

            title: Text("Pegawai"),

            onTap: () {

              Navigator.pushReplacement(

                  context,

                  MaterialPageRoute(

                    builder: (context) => PegawaiPage(),

                  ));

            },

          ),

          ListTile(

            leading: Icon(Icons.logout),

            title: Text("Keluar"),

            onTap: () {

              Navigator.pushReplacement(

                  context,

                  MaterialPageRoute(

                    builder: (context) => Login(),

                  ));

            },

          )

        ],

      ),

    );

  }

}

# PENUTUP

## Kesimpulan

Aplikasi kebun binatang merupakan solusi inovatif yang membawa manfaat besar dalam pengelolaan dan pengalaman pengunjung kebun binatang. Dengan adanya aplikasi ini, pengelola kebun binatang dapat meningkatkan efisiensi operasional.

## Saran

Semoga kedepannya kami dapat mengembangkan aplikasi ini menjadi lebih baik dan juga dapat berguna untuk pengelola kebun binatang .Serta kami terus menemukan solusi untuk aplikasi ini agar berkembang lebih baik lagi kedepannya.

# DAFTAR PUSTAKA

* <https://www.niagahoster.co.id/blog/pengertian-flutter/#Apa_itu_Flutter>
* [https://codingstudio.id/blog/flutter-adalah-kelebihan-dan kekurangan/#Kelebihan\_Framework\_Futter](https://codingstudio.id/blog/flutter-adalah-kelebihan-dan%20kekurangan/#Kelebihan_Framework_Futter)
* <https://www.domainesia.com/berita/rest-api-adalah/#Mengenal_Rest_API>
* <https://www.dicoding.com/blog/flowchart-adalah/>