

MAKALAH

MEMBUAT APLIKASI CRUD MAHASISWA MENGGUNAKAN
FLUTTER DENGAN MEMANFAATKAN REST API



Penyusun :

Satrio Cahyo Agung Wibowo (2214311021)

Dosen Pembimbing :

R DIMAS ADITYOST, MT

JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK.
UNIVERSITAS BHAYANGKARA SURABAYA
2024

KATA PENGANTAR

Dengan penuh rasa syukur, kami memulai makalah ini dengan mengucapkan puji dan syukur kepada Tuhan Yang Maha Esa. Segala puji dan syukur kita haturkan atas rahmat, petunjuk, serta keberkahan-Nya yang senantiasa menyertai langkah-langkah kita dalam mengeksplorasi ilmu dan pengetahuan.

Melalui nikmat dan karunia-Nya, kami diberi kesempatan untuk belajar dan berkontribusi dalam dunia pengetahuan, khususnya dalam pengembangan aplikasi CRUD mahasiswa yang menjadi fokus makalah ini. Kesempatan ini kami jadikan momen untuk bersyukur atas ilmu yang telah diberikan dan untuk berbagi pengetahuan kepada pembaca.

Penyusunan makalah ini tidak terlepas dari dukungan dan bantuan berbagai pihak. Kami mengucapkan terima kasih kepada Bapak Dimas yang telah memberikan pandangan berharga dan saran konstruktif selama proses penulisan makalah ini.

Semoga makalah ini dapat memberikan kontribusi positif dalam pengembangan keterampilan teknis dan pemahaman konsep dasar pengembangan perangkat lunak. Akhir kata, kami berharap pembaca menemukan makalah ini bermanfaat dan dapat menginspirasi untuk eksplorasi lebih lanjut dalam dunia pengembangan perangkat lunak. Terima Kasih.

Satrio Cahyo A.W

DAFTAR ISI

HALAMAN JUDUL	i
KATA PENGANTAR	ii
DAFTAR ISI.....	iii
BAB I PENDAHULUAN	
A. Latar Belakang	1
B. Rumusan Masalah.....	1
C. Tujuan.....	2
D. Manfaat	2
BAB II PEMBAHASAN	
A. Pengertian REST API.....	3
B. Cara Kerja REST API	4
C. Pengertian Aplikasi Mobile	6
D. Penerapan CRUD Aplikasi Mobile Menggunakan REST API	6
BAB III PENUTUP	
A. Kesimpulan.....	10
B. Saran.....	10
DAFTAR PUSTAKA.....	11

BAB I

PENDAHULUAN

A. Latar Belakang

Pengelolaan data mahasiswa merupakan elemen krusial dalam dunia pendidikan tinggi di era digital ini. Dalam upaya meningkatkan efisiensi dan efektivitas, pengembangan aplikasi CRUD (Create, Read, Update, Delete) mahasiswa menjadi solusi yang relevan. Aplikasi semacam ini memungkinkan lembaga pendidikan mengelola informasi mahasiswa dengan lebih terstruktur, membuka peluang peningkatan kualitas layanan akademik.

Dalam konteks tugas kuliah, pemahaman konsep CRUD dan penerapannya dalam pembuatan aplikasi menjadi landasan utama. Tugas ini memberikan kesempatan kepada mahasiswa untuk memahami pemilihan teknologi, desain database, dan langkah-langkah praktis dalam pengembangan aplikasi. Selain itu, tugas ini mendorong kreativitas mahasiswa dalam memecahkan permasalahan pengelolaan data mahasiswa secara inovatif.

Melalui penerapan pengetahuan CRUD, mahasiswa dapat mengembangkan keterampilan yang sesuai dengan tuntutan industri dan memperoleh landasan penting untuk menghadapi tantangan dunia kerja yang semakin kompleks. Tugas kuliah ini bukan hanya sebuah evaluasi akademis, tetapi juga sebuah perjalanan pembelajaran yang dapat memberikan manfaat nyata dalam pemahaman teknologi informasi di bidang pendidikan tinggi.

B. Rumusan Masalah

1. Apa itu Rest API ?
2. Bagaimana Cara Kerja Rest API ?
3. Apa itu Aplikasi Mobile?
4. Bagaimana penerapan konsep dasar CRUD dalam aplikasi mobile menggunakan REST API?

C. Tujuan

1. Memahami Konsep CRUD: Memberikan pemahaman yang kokoh mengenai konsep dasar CRUD (Create, Read, Update, Delete) dan signifikansinya dalam pengelolaan data mahasiswa.
2. Implementasi Aplikasi CRUD: Menyediakan langkah-langkah praktis dalam membuat aplikasi CRUD mahasiswa mulai dari inisialisasi proyek, pembuatan model, hingga pengujian fungsional.
3. Pemecahan Masalah Nyata: Mendorong mahasiswa untuk mengaplikasikan pengetahuan yang diperoleh dalam memecahkan masalah nyata dalam pengelolaan data mahasiswa.

D. Manfaat

1. Peningkatan Keterampilan Teknis: Mahasiswa akan memperoleh keterampilan teknis dalam pengembangan perangkat lunak, khususnya dalam konteks aplikasi CRUD.
2. Pemahaman yang Mendalam: Memberikan pemahaman yang mendalam tentang pentingnya dan implementasi aplikasi CRUD dalam manajemen data mahasiswa.
3. Relevansi dengan Dunia Kerja: Menyediakan pemahaman dan keterampilan yang relevan dengan tuntutan dunia kerja, khususnya di bidang pengembangan perangkat lunak.
4. Pengembangan Kreativitas: Mendorong mahasiswa untuk mengembangkan kreativitas dalam merancang solusi inovatif terhadap tantangan dalam pengelolaan data mahasiswa.
5. Persiapan Menuju Profesionalisme: Menjadi langkah awal dalam mempersiapkan mahasiswa untuk menghadapi dunia kerja, dengan memahami dan mampu mengimplementasikan konsep-konsep kritis dalam pengembangan perangkat lunak.

BAB II

PEMBAHASAN

A. Pengertian REST API

1. Pengertian API

API singkatan dari Application Programming Interface adalah serangkaian aturan dan alat yang memungkinkan suatu perangkat lunak berkomunikasi dengan perangkat lunak lainnya. API menentukan cara elemen-elemen perangkat lunak harus berinteraksi, memungkinkan integrasi dan pertukaran data antara aplikasi yang berbeda. Dengan kata lain, API menyediakan antarmuka yang memungkinkan pengembang menggunakan fungsionalitas atau layanan dari aplikasi atau platform tertentu tanpa harus memahami rinciannya secara internal. API sangat penting dalam pengembangan perangkat lunak karena memfasilitasi kerjasama dan integrasi antar sistem yang berbeda.

2. Pengertian REST

REST (Representational State Transfer) adalah arsitektur komunikasi yang sering digunakan dalam pengembangan aplikasi web. Ini adalah suatu pendekatan untuk merancang layanan web yang ringan, mudah diimplementasikan, dan dapat diperluas. Prinsip REST dikembangkan oleh Roy Fielding dalam disertasinya pada tahun 2000.

3. Pengertian REST API

REST API (Representational State Transfer Application Programming Interface) adalah antarmuka pemrograman aplikasi berbasis arsitektur REST. Ini adalah cara standar untuk berkomunikasi antara aplikasi dan memungkinkan sistem yang berbeda untuk saling berinteraksi. REST API menggunakan prinsip-prinsip REST untuk merancang layanan web yang ringan, mudah diimplementasikan, dan dapat diakses oleh berbagai klien.

Beberapa karakteristik umum dari REST API melibatkan penggunaan metode HTTP standar seperti GET, POST, PUT, dan DELETE untuk berkomunikasi

dengan sumber daya yang diidentifikasi oleh URI (Uniform Resource Identifier). Data sering dikirim dan diterima dalam format yang umum seperti JSON atau XML.

Keuntungan utama dari REST API melibatkan sifat-sifat arsitektur REST, seperti kejelasan, keamanan, dan kemudahan penggunaan. REST API sering digunakan dalam pengembangan aplikasi web dan mobile, serta dalam pembangunan sistem terdistribusi di berbagai industri. Platform dan bahasa pemrograman yang berbeda dapat menggunakan REST API untuk berkomunikasi dan berintegrasi, membuatnya menjadi pendekatan yang populer dalam pengembangan perangkat lunak.

B. Cara Kerja REST API

Cara kerja REST API melibatkan pertukaran data antara klien dan server menggunakan protokol HTTP dan berkomunikasi berdasarkan prinsip-prinsip REST. Berikut adalah langkah-langkah umum yang terlibat dalam cara kerja REST API:

1. Identifikasi Sumber Daya:

- Sumber daya merupakan entitas yang diidentifikasi oleh URI (*Uniform Resource Identifier*). Misalnya, dalam konteks aplikasi CRUD mahasiswa, sumber daya bisa menjadi entitas mahasiswa dengan URI seperti **/mahasiswa**.

2. Pilih Metode HTTP:

- REST API menggunakan metode HTTP seperti **GET**, **POST**, **PUT**, dan **DELETE** untuk berinteraksi dengan sumber daya.
- **GET**: Mengambil data dari server.
- **POST**: Menyimpan data baru di server.
- **PUT** atau **PATCH**: Memperbarui data di server.
- **DELETE**: Menghapus data di server.

3. Buat Permintaan HTTP:

- Klien membuat permintaan HTTP ke server, menggunakan metode yang sesuai dan URI yang mengidentifikasi sumber daya.

- Misalnya, permintaan GET untuk mendapatkan daftar mahasiswa bisa memiliki URI **/mahasiswa** dan metode GET.

4. Proses di Server:

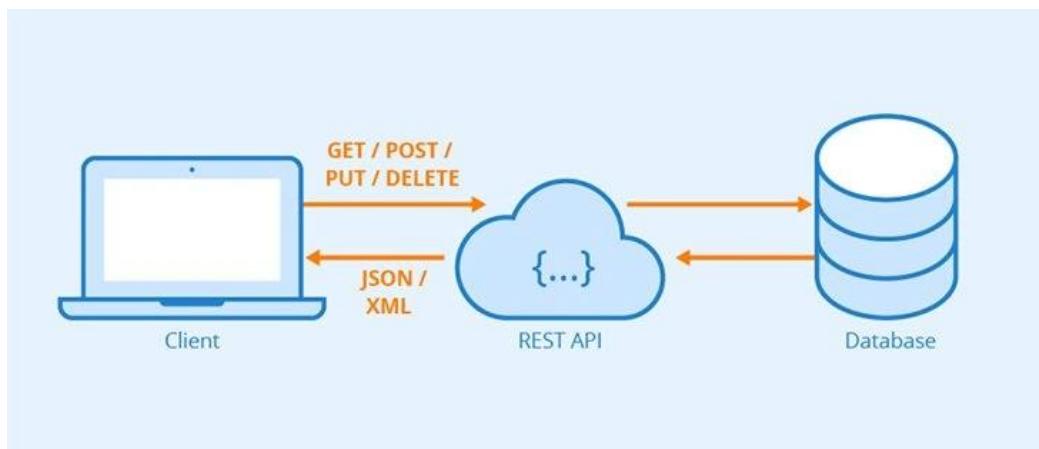
- Server menerima permintaan, memprosesnya, dan memberikan tanggapan sesuai.
- Untuk permintaan GET, server dapat mengirimkan data mahasiswa dalam format yang diminta (JSON, XML, dll.).
- Untuk permintaan POST atau PUT, server dapat menyimpan atau memperbarui data mahasiswa sesuai dengan informasi yang dikirimkan.

5. Kirim Tanggapan ke Klien:

- Server mengirimkan tanggapan ke klien. Tanggapan berisi status HTTP (200 OK, 404 Not Found, dll.) dan mungkin data yang diminta atau hasil dari operasi yang dilakukan.

6. Pemrosesan Tanggapan di Klien:

- Klien menerima tanggapan dan memprosesnya sesuai. Data yang diterima dapat digunakan untuk tampilan atau pemrosesan lebih lanjut di sisi klien.



Gambar 1.0 Cara Kerja REST API

Cara kerja REST API memanfaatkan protokol HTTP yang umum digunakan dan prinsip-prinsip REST, memungkinkan komunikasi yang efisien dan terstandar antara klien dan server. Selain itu, data seringkali disajikan dalam format yang ringkas dan umum seperti JSON, yang memudahkan interpretasi di kedua sisi.

C. Pengertian Aplikasi Mobile

Aplikasi mobile adalah perangkat lunak yang dirancang dan dikembangkan khusus untuk dijalankan pada perangkat mobile seperti smartphone dan tablet. Aplikasi ini dapat diunduh dan diinstal melalui platform distribusi aplikasi, seperti Google Play Store untuk perangkat Android dan Apple App Store untuk perangkat iOS.

D. Penerapan CRUD Aplikasi Mobile Menggunakan REST API *Codeigniter 3*

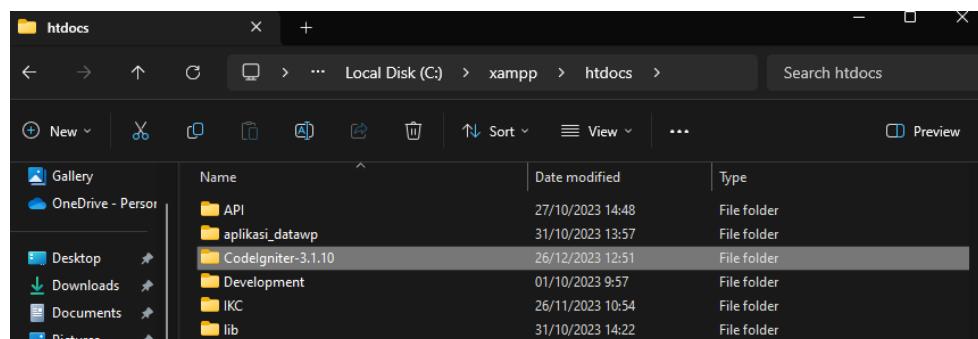
1. Persiapan yang dibutuhkan

- Menyiapkan REST API menggunakan *Codeigniter 3*
- Menyiapkan Flutter Sebagai framework pembuatan aplikasi mobile

2. Menyiapkan REST API menggunakan *Codeigniter 3*

- Installasi *Codeigniter 3* :

- a. Unduh dan ekstrak *Codeigniter 3* dari situs resmi:
<https://codeigniter.com/userguide3/installation/downloads.html>
- b. Pindahkan isi folder Codelgniter ke direktori server web Anda.



- Konfigurasi Database

- a. Buat Database baru di Mariadb beserta tabel yang berisikan kolom sesuai dengan kebutuhan aplikasi
- b. Buka file application/config/database.php.
- c. Atur pengaturan koneksi database sesuai dengan server database Anda (MySQL, PostgreSQL, dll.).

```

76 $db['default'] = array(
77     'dsn' => '',
78     'hostname' => 'localhost',
79     'username' => 'root',
80     'password' => '',
81     'database' => 'mhs',
82     'dbdriver' => 'mysqli',
83     'dbprefix' => '',
84     'pconnect' => FALSE,
85     'db_debug' => (ENVIRONMENT !== 'production'),
86     'cache_on' => FALSE,
87     'cachedir' => '',
88     'char_set' => 'utf8',
89     'dbcollat' => 'utf8_general_ci',
90     'swap_pre' => '',
91     'encrypt' => FALSE,
92     'compress' => FALSE,
93     'stricton' => FALSE,
94     'failover' => array(),
95     'save_queries' => TRUE
96 );

```

- Konfigurasi Controller
 - a. Buat file Controller Mahasiswa di folder ***application/controllers***
 - b. Implementasikan fungsi CRUD dalam kontroler menggunakan *method* Codelgniter (index, show, create, update, delete).
- Konfigurasi Model
 - a. Buat file ***application/models/Mahasiswa_model.php***.
 - b. Definisikan fungsi-fungsi untuk pengolahan data mahasiswa (get, insert, update, delete).
- Konfigurasi Routes
 - a. Buka file ***application/config/routes.php***.
 - b. Tambahkan konfigurasi rute untuk mengarahkan permintaan API ke fungsi yang sesuai di kontroler.

Dengan mengikuti langkah-langkah di atas, Anda dapat menyiapkan REST API sederhana menggunakan Codelgniter 3 untuk operasi CRUD Mahasiswa.

3. Menyiapkan Flutter sebagai Framework pembuatan aplikasi mobile
 - Unduh dan Instal Flutter:
 - a. Ikuti panduan resmi Flutter untuk mengunduh dan menginstal Flutter di sistem Anda: <https://flutter.dev/docs/get-started/install>
 - Buat Proyek Flutter Baru:
 - a. Buka terminal atau command prompt.
 - b. Jalankan perintah berikut untuk membuat proyek baru:



```
Satrio Cahyo ➔ flutter create uascahyo
```

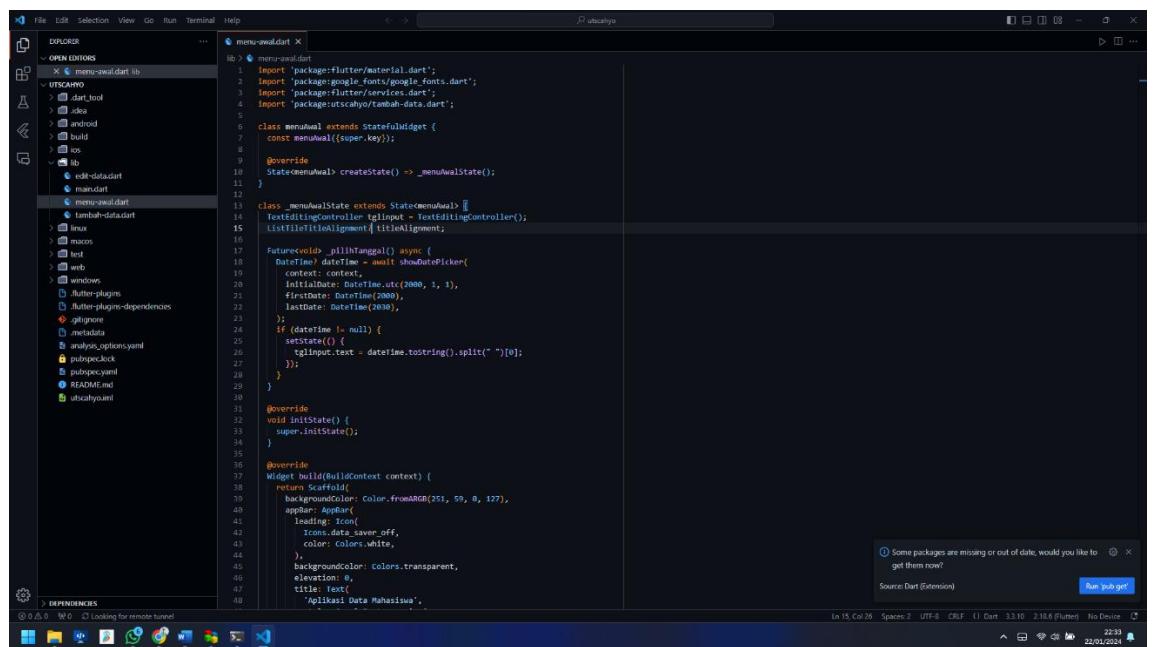
- c. Maka proyek baru akan terbuat secara otomatis, dan tunggu hingga selesai.
- Instal Dependensi Flutter
 - a. Tambahkan paket http ke file pubspec.yaml di bagian dependencies:

```

31   dependencies:
32     flutter:
33       sdk: flutter
34
35     http: ^0.13.4
36     # The following adds the Cupertino Icons font to your application.
37     # For more details, see https://flutter.dev/docs/development/ui/assets-and-images#using-cupertino-icons

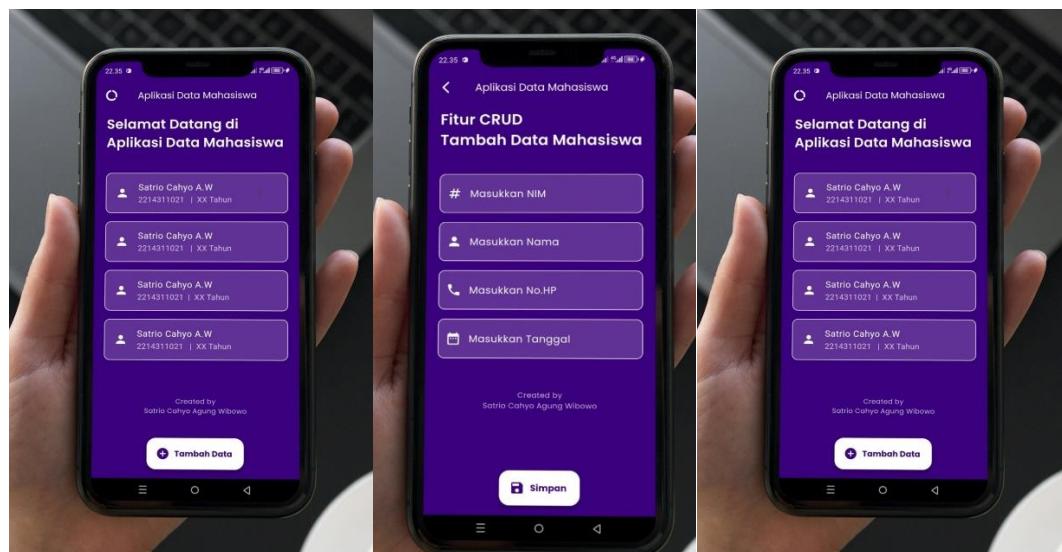
```

- b. Jalankan perintah **flutter pub get** untuk menginstal paket atau **save pubspec.yaml** di vs code
- Buat User Interface Aplikasi Mobile CRUD Mahasiswa



The screenshot shows the Visual Studio Code interface with the following details:

- Project Structure:** The left sidebar shows a tree view of the project files. It includes:
 - OPEN EDITORS: menu-waitdate.dart
 - UTSCAHYO: start, tool, idea, android, build, lib, lib/utschayohu, lib/main.dart, lib/menu-waitdate, lib/tambah-data.dart, lib/linux, lib/macos, lib/win, lib/windows, lib/flutter-plugins, lib/flutter-plugin-dependencies, .gitignore, .metadata, analysis_options.yaml, pubspec.lock, pubspec.yaml, README.md, and utscayho.yaml.
- Code Editor:** The main area displays the content of the 'menu-waitdate.dart' file. The code is a Dart file defining a StatefulWidget named 'menuWala'. It includes imports for flutter/material.dart, package:flutter/material.dart, package:google_fonts/google_fonts.dart, package:flutter/services.dart, and package:utschayohu/tambah-data.dart. It also includes logic for handling date pickers and state management.
- Status Bar:** At the bottom right, there is a status bar showing 'Some packages are missing or out of date, would you like to get them now?' and a 'Run pub get' button.
- Bottom Taskbar:** The taskbar at the bottom shows various icons for different applications like browser, file explorer, and terminal.



- Buat Fungsionalitas CRUD dengan menggunakan **endpoint** API dari Codeigniter 3 yang sudah kita buat di awal

```

import 'dart:convert';
import 'package:flutter/material.dart';
import 'package:http/http.dart' as http;

Run | Debug | Profile
void main() {
  runApp(MyApp());
}

class MyApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'CRUD Mahasiswa',
      home: MahasiswaList(),
    );
  }
}

class MahasiswaList extends StatefulWidget {
  @override
  MahasiswaListState createState() => _MahasiswaListState();
}

class _MahasiswaListState extends State<MahasiswaList> {
  List mahasiswaList = [];

  @override
  void initState() {
    super.initState();
    fetchData();
  }

  Future<void> fetchData() async {
    final response = await http.get(Uri.parse("http://localhost/mahasiswa/api/"));

    if (response.statusCode == 200) {
      final List<dynamic> data = json.decode(response.body);
      setState(() {
    }
  }
}

```

- Jika sudah, jadikan kodingan flutter kita menjadi aplikasi dengan cara ketik pada terminal vs code **flutter build apk –split per abi** , maka flutter akan mengcompile kode kita menjadi sebuah aplikasi yang bisa kita install di hp android.
- Ketika proses compile sudah selesai maka lakukan instalasi aplikasi di hp anda, dan selamat aplikasi crud mahasiswa menggunakan rest api berhasil dibuat.

BAB III

PENUTUP

A. Kesimpulan

Dalam perjalanan pembuatan aplikasi CRUD mahasiswa dengan menggunakan konsep REST API, kita dapat menyimpulkan bahwa:

1. Konsep CRUD (Create, Read, Update, Delete) adalah dasar penting dalam pengelolaan data mahasiswa, memungkinkan manipulasi data dengan efisien dan efektif.
2. Panduan praktis dan langkah-demi-langkah yang diberikan dalam makalah ini dapat menjadi sumber daya berharga bagi mahasiswa dan pengembang aplikasi, membantu mereka memahami konsep CRUD secara mendalam.
3. Penggunaan REST API memberikan manfaat interoperabilitas dan integrasi yang tinggi, memungkinkan aplikasi untuk berkomunikasi dengan sistem lain dengan mudah.
4. Proses pengembangan aplikasi tidak hanya merupakan tugas akademis, tetapi juga memberikan manfaat pendidikan praktis yang relevan dengan tuntutan dunia kerja yang terus berkembang.

B. Saran

Kami sangat menghargai adanya masukan dan saran yang dapat meningkatkan kualitas pengembangan aplikasi CRUD mahasiswa ini.

DAFTAR PUSTAKA

Apa itu API RESTful? - Penjelasan tentang API RESTful - AWS. (n.d.). Amazon Web Services, Inc. <https://aws.amazon.com/id/what-is/restful-api/>

S. (2023, May 17). Format Makalah Lengkap yang Benar - Penerbit Deepublish. Penerbit Deepublish. https://penerbitdeepublish.com/format-makalah/#2_Kata_Pengantar

Welcome to CodeIgniter — CodeIgniter 3.1.13 documentation. (n.d.). <https://codeigniter.com/userguide3/general/welcome.html>

Flutter SDK archive. (n.d.). Flutter. <https://docs.flutter.dev/release/archive?tab=windows>

Nusantara, P. M. T. (2021, March 17). Mobile Apps : Pengertian, Manfaat & Keunggulan Aplikasi Mobile. PT. Majapahit Teknologi Nusantara. <https://majapahit.id/blog/2021/03/17/apa-itu-mobile-apps/>

Dart documentation. (n.d.). Dart. <https://dart.dev/guides>

REST API Server Sederhana dengan Codeigniter 3. (n.d.). Codepolitan. <https://www.codepolitan.com/rest-api-server-sederhana-dengan-codeigniter-58901f324a29f/>

What is a REST API? (n.d.). <https://www.redhat.com/en/topics/api/what-is-a-rest-api>