

## **TUGAS AKHIR**

# **RANCANG BANGUN FRONT-END APLIKASI MOJARNIK (MODUL PEMBELAJARAN ELEKTRONIK) DENGAN MENGUNAKAN FRAMEWORK FLUTTER**

Diajukan Sebagai Persyaratan untuk Menyelesaikan Program Pendidikan  
Diploma III pada Program Studi Teknik Informatika Jurusan Teknik Elektro  
Politeknik Negeri Pontianak



**OLEH:**

**PATRA PURBAYA**

**3201816094**

**PROGRAM STUDI D-III TEKNIK INFORMATIKA  
JURUSAN TEKNIK ELEKTRO  
POLITEKNIK NEGERI PONTIANAK**

**2021**

## HALAMAN PENGESAHAN

### **RANCANG BANGUN FRONT-END APLIKASI MOJARNIK (MODUL PEMBELAJARAN ELEKTRONIK) DENGAN MENGUNAKAN FRAMEWORK FLUTTER**

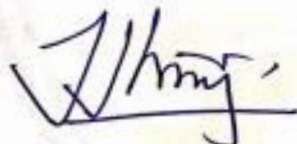
Oleh:

**Patra Purbaya**  
**NIM. 3201816094**

Tugas Akhir ini telah diterima dan disahkan sebagai salah satu syarat untuk menyelesaikan Program Pendidikan Diploma III pada Program Studi Teknik Informatika Jurusan Teknik Elektro Politeknik Negeri Pontianak.

Disahkan oleh:

**Ketua Jurusan  
Teknik Elektro**



**Hasan, S.T., M.T**  
**NIP. 19710820 199903 1 003**

**Ketua Program Studi  
Teknik Informatika**



**Suheri, S.T., M.Cs**  
**NIP. 19830717 200812 1 005**

Mengetahui,

**Direktur Politeknik Negeri Pontianak**



**Dr. Ir. H. Muhammad Toasin Asha, M.Si**  
**NIP. 196112251990111001**

## HALAMAN PERNYATAAN

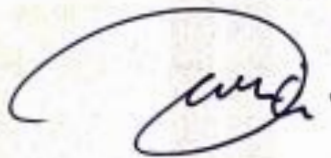
### **RANCANG BANGUN FRONT-END APLIKASI MOJARNIK (MODUL PEMBELAJARAN ELEKTRONIK) DENGAN MENGUNAKAN FRAMEWORK FLUTTER**

Oleh:

**Patra Purbaya**

**NIM. 3201816094**

Dosen Pembimbing:



**Fitri Wibowo, S.ST., MT.**

**NIP. 19851228 201504 1 002**

Telah dipertahankan di depan penguji pada tanggal 3 September 2021 dan dinyatakan memenuhi syarat sebagai Laporan Tugas Akhir.

Dosen Penguji:

Penguji I,

Penguji II,



**Wawan Heryawan, S.T., M.T.**

**NIP. 19701016 199702 1 002**



**Tommi Suryanto, S.Kom., M.Kom.**

**NIP. 19901020 201903 1 013**

## HALAMAN PERNYATAAN ORISINALITAS

Saya yang bertanda tangan dibawah ini:

Nama : Patra Purbaya  
NIM : 3201816094  
Jurusan / Program Studi : Teknik Elektro / Teknik Informatika  
Judul Tugas Akhir : Rancang Bangun Front-End Aplikasi Mojarnik  
(Modul Pembelajaran Elektronik) Dengan  
Menggunakan Framework Flutter

Menyatakan dengan sebenar-benarnya bahwa penulisan Tugas Akhir ini berdasarkan hasil penelitian, pemikiran dan pemaparan asli dari saya sendiri, baik untuk naskah Tugas Akhir maupun kegiatan yang tercantum sebagai bagian dari Tugas Akhir ini. Jika terdapat karya orang lain, saya akan mencantumkan sumber yang jelas.

Demikian pernyataan ini saya buat dengan sesungguhnya dan apabila dikemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya tulis ini dan sanksi lain sesuai dengan peraturan yang berlaku di Politeknik Negeri Pontianak.

Demikian pernyataan ini saya buat dalam keadaan sadar tanpa paksaan dari pihak manapun.

Pontianak, 01 September 2021

Yang membuat pernyataan,



Patra Purbaya

NIM. 3201816094

## RIWAYAT HIDUP



### **Biodata Mahasiswa:**

Nama Mahasiswa	: Patra Purbaya
NIM	: 3201816094
Tempat / Tanggal Lahir	: Pontianak, 09 Desember 2021
Jenis Kelamin	: Laki-Laki
Agama	: Islam
Alamat	: Jl.Parit Haji Husin 1., Gg. Palasari No.11A
No. Telepon Rumah / Handphone	: 0813-5196-3101
Email	: patra.purbaya277@gmail.com

## ABSTRAK

Pendidikan adalah hal yang sangat penting untuk ditempuh dalam kehidupan semua umat manusia dari dulu hingga sekarang. Untuk mewujudkan pendidikan yang berhasil, tentunya tidak terlepas dari usaha pelaksanaan proses belajar mengajar yang bagaimanapun ilmu yang bermanfaat dapat tersampaikan kepada para penempuh pendidikan khususnya di era sekarang yang sedang dalam kondisi pandemi COVID-19 sehingga pembelajaran harus dilakukan secara *online* (dalam jaringan). Dengan pembelajaran yang dilakukan secara *online* inilah diperlukan adanya suatu teknologi yang dapat mendukung bahkan memudahkan proses belajar mengajar antara pengajar dan pembelajar.

Dengan adanya masalah yang telah diuraikan sebelumnya, penulis memberikan solusi dengan menciptakan suatu aplikasi modul elektronik berbasis *mobile* sebagai bentuk Tugas Akhir sekaligus mengembangkan teknologi yang beredar di masyarakat.

Aplikasi modul elektronik yang bernama MOJARNIK (Modul Ajar Elektronik) ini memiliki fungsi sebagai media baca, penyimpanan, dan media interaksi bagi para pembelajar khususnya di Politeknik Negeri Pontianak. Aplikasi ini memiliki fitur komentar, *bookmark*, dan anotasi untuk memudahkan dalam interaksi antara pengajar dan pembelajar. Manfaat yang ingin dicapai dengan adanya aplikasi ini adalah diharapkan dapat mempermudah pengajar dan peserta didik dalam hal distribusi, pengelolaan, dan mengakses materi atau modul ajar.

**Kata Kunci :** *front-end*, modul elektronik, aplikasi *mobile*, Flutter, *online*

## **ABSTRACT**

*Education was a very important thing e life of all mankind from the past till now. To realize a successful education, of course, it cannot be separated from the effort to implement the teaching and learning process, however useful knowledge can be conveyed to students, especially now which is in a COVID-19 pandemic condition so that learning must be online. With online learning, it is necessary to have a technology that can support and even facilitate the teaching and learning process between teachers and students.*

*With the problems that have been described previously, the author provides a solution by creating a mobile-based electronic module application as a form of Final Project as well as developing technology of the society.*

*This electronic module application called MOJARNIK (Electronic Teaching Module) has a function as a reading, storage, and interaction medium for students, especially at the Pontianak State Polytechnic. This application features comments, bookmarks, and annotations to facilitate interaction between teachers and students. The benefit to be achieved with this application is that it is expected to make it easier for teachers and students in terms of distribution, management, and access to materials or teaching modules.*

**Keywords:** *front-end, electronic module, mobile application, Flutter, online*

## **PRAKATA**

Puji syukur kepada Tuhan Yang Maha Esa atas berkat dan pimpinan-Nya yang telah penulis terima selama melaksanakan Tugas Akhir ini, sehingga pada akhirnya penulis dapat menyelesaikan Tugas Akhir ini dengan baik. Pada kesempatan ini penulis menyampaikan rasa terima kasih yang sebesar besarnya kepada orang-orang yang telah berperan sehingga tugas akhir ini dapat terselesaikan, antara lain :

1. Allah SWT yang telah memberikan kesehatan dan limpahan rahmat-Nya sehingga Praktik Kerja Lapangan ini dapat berjalan baik.
2. Kedua orang tua serta saudara dari penulis yang telah memberikan doa, semangat , masukan, serta saran.
3. Bapak Ir. H. Muhammad Toasin Asha, M.Si, selaku Direktur Politeknik Negeri Pontianak.
4. Bapak Hasan, S.T., M.T., selaku Ketua Jurusan Teknik Elektro.
5. Bapak Suheri, S.T, M.Cs. , selaku Ketua Program Studi Teknik Informatika
6. Bapak Fitri Wibowo, S.ST., MT. selaku Dosen Pembimbing yang telah memberikan bimbingan serta arahan untuk menyelesaikan Tugas Akhir.
7. Bapak Wawan Heryawan, S.T., M.T. selaku Dosen Penguji 1 yang telah meluangkan waktu untuk menguji.
8. Bapak Tommi Suryanto, S.Kom., M.Kom. selaku Dosen Penguji 2 yang telah meluangkan waktu untuk menguji.
9. Seluruh staf pengajar dan administrasi Politeknik Negeri Pontianak, khususnya di Jurusan Teknik Elektro Program Studi Teknik Informatika.
10. Teman-teman Program Studi Teknik Informatika yang selalu memberi semangat, bantuan, dan saran selama proses penyusunan Tugas Akhir ini.
11. Teman-teman seperjuangan yang selama ini mendukung dan menyemangati selama proses menjalani kehidupan.
12. *Partner* saya Sri Purwanti yang sudah membantu saya dalam susah dan senang.



Dalam penulisan laporan ini, penulis yakin bahwa sudah bekerja dengan maksimal serta diarahkan oleh dosen pembimbing. Maka dari itu kritik dan saran sangat dibutuhkan untuk memperbaiki penulisan Tugas Akhir ini menjadi lebih baik dan bermanfaat bagi penulis dan pembaca.

Pontianak, 01 September 2021

Patra Purbaya

NIM. 3201816094

## DAFTAR ISI

<b>HALAMAN PENGESAHAN.....</b>	<b>i</b>
<b>HALAMAN PERNYATAAN ORISINALITAS .....</b>	<b>iii</b>
<b>RIWAYAT HIDUP .....</b>	<b>iv</b>
<b>ABSTRAK .....</b>	<b>v</b>
<b>PRAKATA .....</b>	<b>vii</b>
<b>DAFTAR ISI.....</b>	<b>ix</b>
<b>DAFTAR TABEL .....</b>	<b>xi</b>
<b>DAFTAR GAMBAR.....</b>	<b>xii</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah .....	2
1.3. Batasan Masalah.....	2
1.4. Tujuan Penelitian .....	2
1.5. Manfaat Penelitian .....	2
1.6. Metodologi .....	3
1.6.1. Studi Literatur .....	4
1.6.2. Identifikasi Data .....	4
1.6.3. Pembuatan Tampilan Antarmuka Pengguna suatu Halaman .....	4
1.6.4. Pembuatan Kode Program .....	5
1.6.5. Uji Coba Aplikasi.....	5
1.7. Sistematika Tugas Akhir .....	5
<b>BAB II LANDASAN TEORI .....</b>	<b>6</b>
2.1. Tinjauan Pustaka .....	6
2.2. Dasar Teori.....	6

<b>BAB III PERANCANGAN SISTEM .....</b>	<b>9</b>
3.1. Tampilan Antarmuka .....	9
3.1.1. Halaman <i>Login</i> .....	9
3.1.2. Halaman <i>Home</i> .....	9
3.1.3. Halaman Detail Modul .....	10
3.1.4. Halaman <i>Reading</i> .....	11
3.1.5. Halaman <i>Search</i> .....	11
3.1.6. Halaman <i>Settings</i> .....	12
3.1.7. Halaman <i>About</i> .....	12
3.1.8. <i>Drawer</i> .....	13
3.2. Use Case Diagram .....	13
<b>BAB IV HASIL DAN PEMBAHASAN .....</b>	<b>15</b>
4.1. Implementasi Tampilan Antarmuka .....	15
4.1.1. Halaman <i>Login</i> .....	15
4.1.2. Halaman <i>Home</i> .....	16
4.1.3. Halaman Detail Modul .....	16
4.1.4. Halaman <i>Reading</i> .....	17
4.1.5. Halaman <i>Search</i> .....	17
4.1.6. Halaman <i>Settings</i> .....	18
4.1.7. Halaman <i>About</i> .....	18
4.1.8. <i>Drawer</i> .....	19
4.2. Pengambilan Data API .....	19
4.3. Implementasi Fungsi .....	22
4.4. Uji Coba .....	24
<b>BAB V PENUTUP .....</b>	<b>33</b>
5.1. Kesimpulan .....	33
5.2. Saran .....	33
<b>DAFTAR PUSTAKA .....</b>	<b>34</b>

## DAFTAR TABEL

<b>Tabel 1.</b> Data API.....	19
-------------------------------	----

## DAFTAR GAMBAR

<b>Gambar 1</b> Diagram Alir Metode Penelitian.....	3
<b>Gambar 2.</b> Visual Studio Code .....	8
<b>Gambar 3.</b> Postman .....	8
<b>Gambar 4.</b> Halaman Login .....	9
<b>Gambar 5.</b> Halaman Home .....	10
<b>Gambar 6.</b> Halaman Detail Modul .....	10
<b>Gambar 7.</b> Halaman Reading.....	11
<b>Gambar 8.</b> Halaman Search .....	12
<b>Gambar 9.</b> Halaman Settings .....	12
<b>Gambar 10.</b> Halaman About .....	13
<b>Gambar 11.</b> Drawer .....	13
<b>Gambar 12.</b> Use Case Diagram .....	14
<b>Gambar 13.</b> Cek Koneksi .....	15
<b>Gambar 14.</b> Halaman Login Baru.....	16
<b>Gambar 15.</b> Halaman Home Baru .....	16
<b>Gambar 16.</b> Halaman Detail Modul Baru .....	17
<b>Gambar 17.</b> Halaman Reading Baru .....	17
<b>Gambar 18.</b> Halaman Search Baru .....	18
<b>Gambar 19.</b> Halaman Settings Baru .....	18
<b>Gambar 20.</b> Halaman About Baru .....	19
<b>Gambar 21.</b> Drawer Baru.....	19

# **BAB I**

## **PENDAHULUAN**

### **1.1.Latar Belakang**

Seiring berkembangnya teknologi di era digital sekarang ini, segala sesuatu dilakukan diselesaikan menggunakan teknologi yang semakin berkembang dari waktu ke waktu. Dari segala aspek kehidupan, pasti terdapat teknologi yang dapat memudahkan pekerjaan bahkan memberikan jaminan keselamatan bagi penggunanya. Oleh karena itu, penulis proposal tugas akhir ini juga memiliki peran dalam pengembangan teknologi di masyarakat.

Yang melatarbelakangi judul dari proposal tugas akhir ini adalah adanya kondisi dari pandemi COVID-19 yang menyebabkan dampak yang cukup besar terhadap berbagai aspek kehidupan seperti aspek pendidikan, ekonomi, kesehatan, dan pariwisata. Seiring berjalannya waktu masyarakat di berbagai negara termasuk di Indonesia harus beradaptasi dengan kebiasaan baru (New Normal) termasuk dalam aspek pendidikan. Proses Belajar Mengajar (PBM) yang sebelumnya banyak dilakukan di dalam ruang kelas dengan bertatap muka langsung antara pengajar dengan peserta didik saat ini harus beralih ke sistem pembelajaran online atau dalam jaringan (daring) untuk menekan penyebaran virus tersebut. Berdasarkan kondisi tersebut, adanya modul pembelajaran elektronik (E-Modul) akan membantu pengajar dalam hal pendistribusian materi ajar. Oleh karena itu, penulis berinisiatif menetapkan judul dan berencana menciptakan aplikasi modul elektronik berbasis mobile dengan menggunakan Flutter.

Adanya proposal tugas akhir ini diharapkan dapat membantu penulis dalam menyelesaikan Tugas Akhir yang berjudul Rancang Bangun Front-End Aplikasi Mojarnik (Modul Pembelajaran Elektronik) Dengan Menggunakan Framework Flutter Di Politeknik Negeri Pontianak sebagai bentuk usaha untuk memudahkan mahasiswa dalam mengakses modul-modul materi yang akan dipelajari. Untuk studi kasus penerapan aplikasi yang akan dibuat adalah di Politeknik Negeri Pontianak, namun tidak menutup kemungkinan aplikasi ini juga dapat dimanfaatkan di sekolah-sekolah maupun perguruan tinggi lain.

## 1.2.Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan sebelumnya, maka penetapan masalah dalam Proposal Tugas Akhir ini adalah bagaimana merancang dan membangun *Front-end* aplikasi Mojarnik (Modul Pembelajaran Elektronik) Dengan Menggunakan *Framework* Flutter Di Politeknik Negeri Pontianak ?

## 1.3.Batasan Masalah

Batasan masalah dari Proposal Tugas Akhir ini adalah :

1. *Front-end* aplikasi ini merupakan aplikasi modul pembelajaran elektronik berbasis *mobile*.
2. Aplikasi hanya dapat diakses *user* yaitu mahasiswa.
3. Aplikasi menyajikan modul-modul untuk dibaca oleh mahasiswa.
4. Mahasiswa memberikan *bookmark*, anotasi, dan *notes* pada modul.
5. Mahasiswa dapat mengunduh dokumen.
6. Dosen mengelola dan memberikan modul dari web (*Back-end*).
7. Data yang akan digunakan aplikasi diambil dari *API* yang dibuat oleh *Back-end Developer* melalui *URL* yang diberikan.
8. Bahasa Pemrograman yang digunakan adalah Dart.
9. *Tool* yang digunakan adalah Visual Studio Code,dan Postman,.

## 1.4.Tujuan Penelitian

Menghasilkan aplikasi modul pembelajaran berbasis *mobile* yang membantu dosen dan mahasiswa di Politeknik Negeri Pontianak sebagai media perantara dalam mengakses modul yang diberikan oleh dosen kepada mahasiswa.

## 1.5.Manfaat Penelitian

Adapun manfaat yang akan diperoleh dari penelitian Tugas Akhir ini yaitu :

### 1.5.1 Bagi Mahasiswa

Mempermudah mahasiswa dalam mengakses modul yang diberikan oleh dosen sebagai media pembelajaran dalam proses belajar-mengajar tanpa harus menyimpan dan menggunakan modul yang berbentuk secara fisik (*hardcopy*)

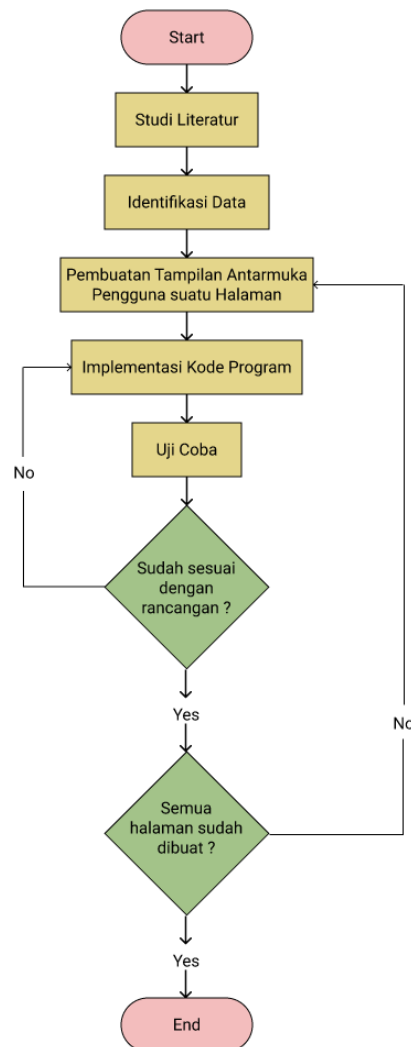
yang pada umumnya dapat menggunakan biaya lebih, memenuhi kapasitas ruang, bahkan memberatkan apabila dibawa.

### 1.5.2 Bagi Politeknik Negeri Pontianak

Aplikasi dapat menjadi media pembelajaran yang mendukung kebijakan kampus yaitu mengutamakan pembelajaran jarak jauh atau daring (dalam jaringan) dalam kondisi pembatasan sosial (*social distancing*) agar terhindar dari pandemi COVID-19.

## 1.6. Metodologi

Diagram alir metodologi penelitian disajikan pada **Gambar 1**.



**Gambar 1.** Diagram Alir Metodologi Penelitian



Metode pengembangan aplikasi E-Moo yang digunakan pada penelitian ini adalah *Rapid Application Development (RAD)*. Metode ini dipilih karena kecepatan, ketepatan, dan lebih sedikit biaya dibandingkan dengan metode konvensional. Adapun dengan adanya keterlibatan *user* saat proses mendesain menjadikan kebutuhan tersebut dapat terpenuhi dengan baik sehingga kepuasan juga akan meningkat sebagai pengguna aplikasi.

Adapun metodologi secara garis besar dibagi menjadi empat (4) tahapan yaitu 1) Studi literatur, 2) Identifikasi data, 3) Perancangan & pembuatan tampilan antarmuka pengguna berbasis *Graphical User Interface (GUI)*, 4) Membuat kode programnya, dan 5) Uji coba[2].

#### **1.6.1. Studi Literatur**

Pada tahapan ini penulis melakukan pencarian referensi yang relevan dengan judul penelitian ini dari berbagai macam sumber yang ditemukan. Dengan adanya studi literatur juga penulis dapat mengidentifikasi dan menganalisa data-data yang diperlukan saat pengembangan aplikasi dilakukan.

#### **1.6.2. Identifikasi Data**

Data-data yang telah diketahui pada tahapan sebelumnya yaitu studi literatur, maka penulis akan melakukan identifikasi data-data yang akan diambil dan ditampilkan pada aplikasi sesuai kebutuhan.

#### **1.6.3. Pembuatan Tampilan Antarmuka Pengguna suatu Halaman**

Pembuatan *user interface (UI)* mempunyai peran penting dalam efektivitas suatu sistem informasi dimana hal itu bertujuan untuk menjadikan teknologi informasi tersebut mudah digunakan oleh pengguna atau disebut dengan istilah *user friendly* [3]. Pembuatan tampilan antarmuka pengguna ini bertujuan agar pengguna (*user*) mendapat kemudahan saat mengoperasikan aplikasi. Namun tidak hanya itu, tampilan antarmuka dari sisi estetika juga perlu diperhatikan agar *software* dapat menjadi daya tarik dan dilihat dengan nyaman oleh *user*.

Dari penjelasan tersebut dapat disimpulkan bahwa tampilan antarmuka pengguna penting dilakukan secara berulang-ulang tergantung banyaknya halaman yang akan dibuat.

#### **1.6.4. Pembuatan Kode Program**

Setelah tampilan antarmuka dari suatu halaman sudah dibuat, maka kode program akan diimplementasikan sesuai masing-masing fitur yang terdapat di dalam suatu halaman.

#### **1.6.5. Uji Coba Aplikasi**

Uji coba dilakukan dengan metode *black-box testing* yaitu dengan memastikan apakah hasil implementasi antarmuka dengan kode program dari suatu fitur telah berjalan sesuai rancangan, jika sudah maka kembali lagi kepada pembuatan tampilan antarmuka dan implementasi kode program dari fitur yang selanjutnya. Target dari pengujian akan dilakukan kepada beberapa mahasiswa dan dosen untuk memastikan bahwa aplikasi sudah sesuai rancangan dan siap digunakan.

### **1.7. Sistematika Tugas Akhir**

Adapun sistematika penulisan yang terdapat pada Tugas Akhir ini terdiri atas:

#### **BAB 1: PENDAHULUAN**

Pendahuluan menjelaskan secara singkat tentang latar belakang, rumusan masalah, batasan masalah, tujuan, manfaat dari penelitian yang dilakukan.

#### **BAB 2: DASAR TEORI**

Dasar teori menguraikan teori-teori yang berhubungan dengan permasalahan yang dibahas.

#### **BAB 3: PERANCANGAN SISTEM**

Perancangan sistem memuat penjelasan tentang analisis serta perancangan Aplikasi Modul Ajar Elektronik (MOJARNIK)

#### **BAB 4: HASIL DAN PEMBAHASAN**

Berisi tentang pembahasan mengenai rancangan aplikasi, penerapan API *end-point* dalam penggunaan fitur, dan hasil yang ditampilkan oleh aplikasi.

#### **BAB 5: PENUTUP**

Penutup berisi tentang uraian kesimpulan dan saran bagi pembaca agar dapat dijadikan masukan dalam penelitian ini.

## **BAB II**

### **LANDASAN TEORI**

#### **2.1. Tinjauan Pustaka**

Terdapat penelitian sejenis yang pernah dilakukan sebelum penelitian ini dibuat yang dapat dijadikan acuan. Salahsatunya adalah penelitian[4] yang dilakukan pada tahun 2018 yaitu pengembangan aplikasi modul pembelajaran matematika berbasis mobile. Penelitian tersebut berfokus pada pembahasan materi terkait matriks. Tujuan dari penelitian tersebut adalah untuk membantu guru dalam mengajar peserta didik yang kurang memiliki motivasi maupun antusias dalam aktivitas kegiatan belajar karena peserta didik merasa bosan dan masih merasa malas untuk mempelajari matematika karena memiliki rumus-rumus dan perhitungan. Penelitian tersebut mentransformasikan modul pembelajaran yang berbasis konvensional menjadi berbasis digital. Adapun IDE yang digunakan dalam pengembangan aplikasi ini adalah Android Studio sebagai basis tampilan atau layout sekaligus mencakup sistem di dalamnya.

Berdasarkan perbandingan teknologi, tujuan serta metode dari penelitian sejenis sebelumnya, maka pada penelitian ini penulis mencoba untuk membuat sebuah aplikasi modul pembelajaran yang dapat menampilkan modul-modul yang diberikan dari dosen ke mahasiswa sesuai program studi dan semesternya masing-masing. Sedangkan media yang digunakan dalam mengakses modul adalah perangkat mobile seperti Android.

#### **2.2. Dasar Teori**

##### **2.2.1. Flutter**

Flutter adalah SDK untuk pengembangan aplikasi mobile yang dikembangkan oleh Google. Sama seperti React Native, framework ini dapat digunakan untuk membuat atau mengembangkan aplikasi mobile yang dapat berjalan pada device iOS dan Android. Dibuat menggunakan

bahasa C, C++, Dart and Skia membuat Flutter ini menjadi salah satu framework yang sangat menarik dan worth untuk dipelajari[5].

Flutter terdiri dari beberapa komponen utama yaitu :

#### 1. Flutter Engine

Flutter Engine dibuat menggunakan bahasa pemrograman C++, yang memberikan dukungan rendering tingkat rendah dari graph library Skia yang dimiliki oleh Google dan berinteraksi dengan SDK platform specific seperti yang disediakan oleh Android dan iOS.

#### 2. Library Foundation

Dibuat dengan bahasa pemrograman Dart, menyediakan fungsi-fungsi dari berbagai class dasar yang berguna membangun aplikasi menggunakan Flutter, seperti API yang digunakan untuk berkomunikasi dengan engine.

#### 3. Widget

Widget merupakan blok bangunan dasar dari tampilan antarmuka aplikasi Flutter. Framework Flutter berisi dua set widget yang disesuaikan dengan bahasa desain tertentu, diantaranya yaitu widget material app yang menerapkan bahasa desain Google dengan nama yang sama, sedangkan widget Cupertino menerapkan desain meniru desain iOS milik Apple.

Berbagai widget tersedia di Flutter yang digunakan untuk membangun tampilan (layout). Pada dasarnya tampilan ini hanya disusun oleh 2 widget yang menjadi parent widget dari widget-widget lainnya, widget itu adalah Column (Kolom) dan Row (Baris) yang masing-masing menyusun widget secara vertical maupun horizontal.

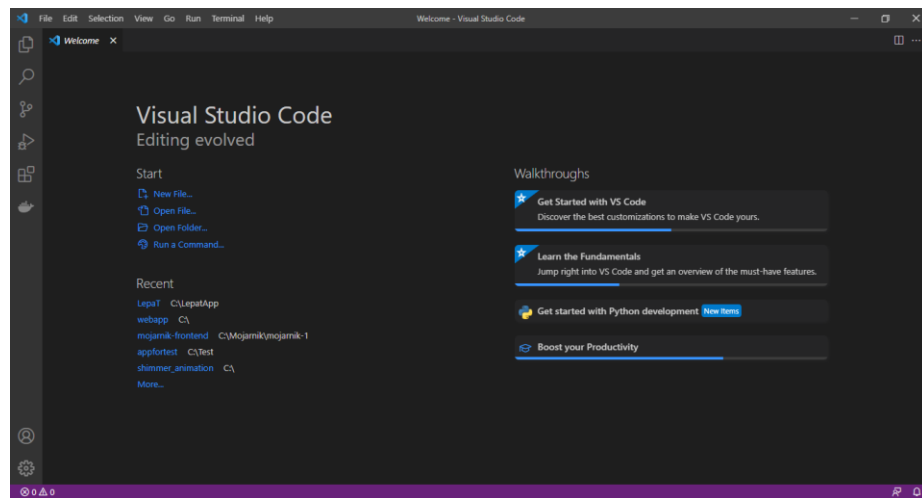
#### 4. Asset dan Dependencies

Asset dan dependencies akan mengatur segala *package* dan *asset* yang dibutuhkan oleh *programmer* Flutter. Dengan ini, programmer dapat memodifikasi judul, deskripsi, asset-aset dimana diatur di dalam file `pubspec.yaml`.

### 2.2.2. Visual Studio Code

VSCode adalah code editor pilihan dari penulis dalam mengembangkan aplikasi pada penelitian ini. Tentunya alasan penulis

memilih editor ini karena terdapat kemudahan tergantung masing-masing pengguna. Tampilan dari VSCode disajikan pada Gambar 2.

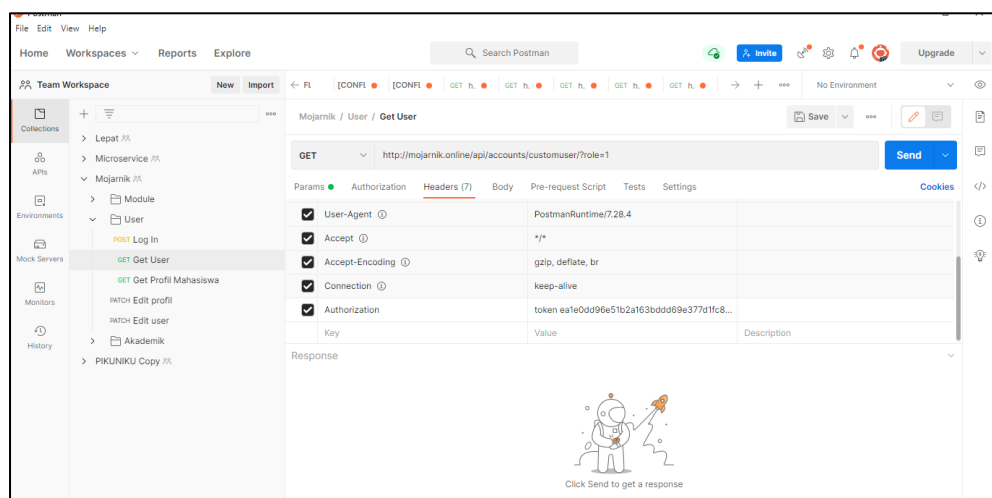


**Gambar 2.** Visual Studio Code

Layaknya aplikasi editor kode lain seperti Sublime atau Atom, Visual Studio Code mendukung berbagai jenis bahasa pemrograman. Mulai dari JavaScript, Java, PHP, C++, C#, Go, JSON, dan lainnya. Aplikasi editor ini bahkan secara otomatis mengidentifikasi jenis bahasa pemrograman yang digunakan dan memberikan variasi warna sesuai dengan setiap fungsi dalam rangkaian kode tersebut[7].

### 2.2.3. Postman

Aplikasi yang biasanya digunakan para *programmer* saat melakukan uji coba terhadap API yang diberikan oleh *Back-End Programmer*. Gambar dari Postman disajikan pada **Gambar 3** di bawah ini.



**Gambar 3.** Postman

## BAB III

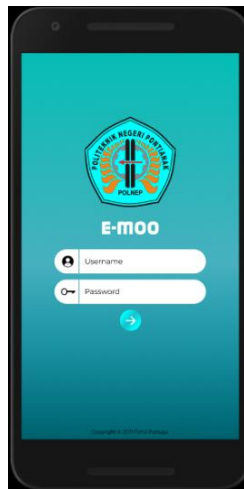
### PERANCANGAN SISTEM

#### 3.1. Tampilan Antarmuka

Tampilan antarmuka adalah bentuk tampilan yang berhadapan langsung dengan pengguna secara grafis. Berikut ini tampilan yang sudah dirancang sebelumnya pada proposal tugas akhir..

##### 3.1.1. Halaman *Login*

Pada halaman ini, *user* memasukkan *username* dan *password* sebagai syarat agar dapat masuk ke halaman utama aplikasi. Dari *username* dan *password* yang dimasukkan kemudian akan dicek di *database* apakah ada akun yang memiliki *username* dan *password* tersebut. Apabila benar, maka akan langsung masuk ke halaman utama (*Home Page*). *Mock-up* halaman disajikan pada **Gambar 4**.



**Gambar 4.** Halaman *Login*

##### 3.1.2. Halaman *Home*

Setelah *user* melakukan *login*, maka layar akan menampilkan halaman utama (*Home Page*) dimana *user* dapat menemukan modul yang ditampilkan berdasarkan jurusan, program studi, dan semester masing-masing dari mahasiswa. Modul yang terdapat di *database* akan tersaring secara otomatis saat mengecek program studi dan semester mahasiswa di *database* kemudian dalam modul-modul tersebut dibagi lagi menjadi beberapa *file* yang dipisah dari beberapa bagian. Saat modul disentuh maka layar akan berpindah ke halaman detail modul tersebut (*Module Details Page*).

Pada halaman ini juga terdapat tombol pencarian yang akan berpindah ke halaman pencarian (*Search Page*) apabila disentuh dan akan menampilkan data sesuai yang diketik oleh *user*. *Mock-up* halaman disajikan pada **Gambar 5**.



**Gambar 5.** Halaman *Home*

### 3.1.3. Halaman Detail Modul

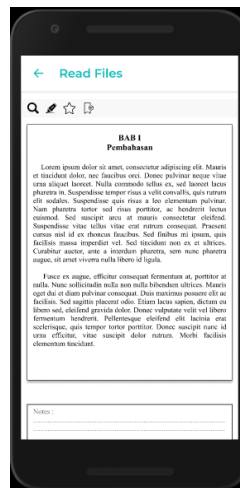
Halaman detail modul akan menampilkan beberapa *file* yang dipecah menjadi beberapa bagian dari bentuk asalnya. Dalam satu *file* bisa saja terdapat beberapa halaman lagi di dalamnya agar *user* dapat membaca secara terstruktur dan tidak merasa malas terlebih dahulu saat melihat halaman yang terlalu banyak. Kemudian apabila salahsatu *file* disentuh maka layar akan berpindah menuju halaman membaca (*Reading Page*).. *Mock-up* halaman disajikan pada **Gambar 6**.



**Gambar 6.** Halaman Detail Modul

### 3.1.4. Halaman *Reading*

Pada halaman ini, *file* yang berasal dari suatu modul akan ditampilkan secara berurutan tergantung jumlah halaman yang dimiliki pada *file*. *User* dapat melakukan beberapa Tindakan terhadap *file* ini seperti membuat anotasi, menambah catatan, menambah *bookmark*, dan fitur lainnya. Apabila terdapat lebih dari 1 halaman maka layar dapat melakukan *scroll* secara vertikal saat berganti halaman *file*. *File* akan diperbarui dan tersimpan secara otomatis di database saat tombol kembali disentuh agar saat *user* ingin menyimpan *file*, tidak repot-repot melakukan penyimpanan *file* secara manual. *Mock-up* halaman disajikan pada **Gambar 7**.



**Gambar 7.** Halaman *Reading*

### 3.1.5. Halaman *Search*

Terdapat kasus yang umumnya terjadi bahwa apabila terdapat banyak modul yang ditampilkan sehingga menyebabkan *user* kesulitan menemukan modul yang diinginkan. Maka dari itu, pada halaman *search* ini *user* dapat mencari modul yang diinginkan melalui kolom yang dapat diisi dan akan langsung menampilkan data seiring *user* mengetikkan kata kunci modul yang dicari. *Mock-up* halaman disajikan pada **Gambar 8**.

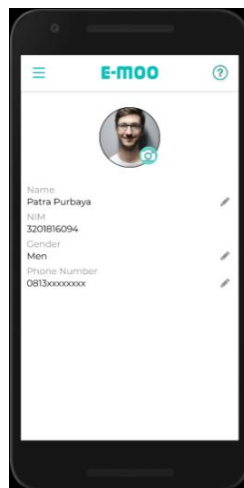




**Gambar 8.** Halaman *Search*

### 3.1.6. Halaman *Settings*

Halaman pengaturan ini berfungsi untuk mahasiswa memodifikasi beberapa identitas yang dapat diubah dan diisi oleh masing-masing mahasiswa itu sendiri. Beberapa identitas tidak dapat diubah dikarenakan untuk mencegah adanya penyimpangan tertentu mengenai data asli dari mahasiswa yang dapat mengakibatkan kesalahan pada database. *Mock-up* halaman disajikan pada **Gambar 9**.



**Gambar 9.** Halaman *Settings*

### 3.1.7. Halaman *About*

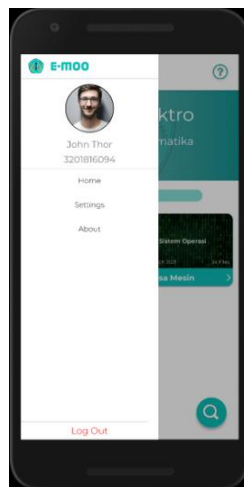
Halaman *about* ini berisi segala informasi tentang aplikasi yang dibuat dan hanya dapat dimodifikasi oleh *developer*. Informasi yang ditampilkan bisa terdiri dari deskripsi, identitas *developer*, versi aplikasi, dan lainnya. *Mock-up* halaman disajikan pada **Gambar 10**.



**Gambar 10.** Halaman *About*

### 3.1.8. *Drawer*

*Drawer* ini berfungsi sebagai navigasi kepada *user* apabila ingin berpindah menuju halaman lain yang terdiri dari halaman utama (*Home Page*), halaman pengaturan (*Settings Page*), dan halaman tentang aplikasi (*About Page*). Dan *user* dapat melakukan *log out* melalui *drawer* ini apabila ingin mengeluarkan atau mengganti akun yang akan digunakan. *Mock-up* disajikan pada **Gambar 11**.

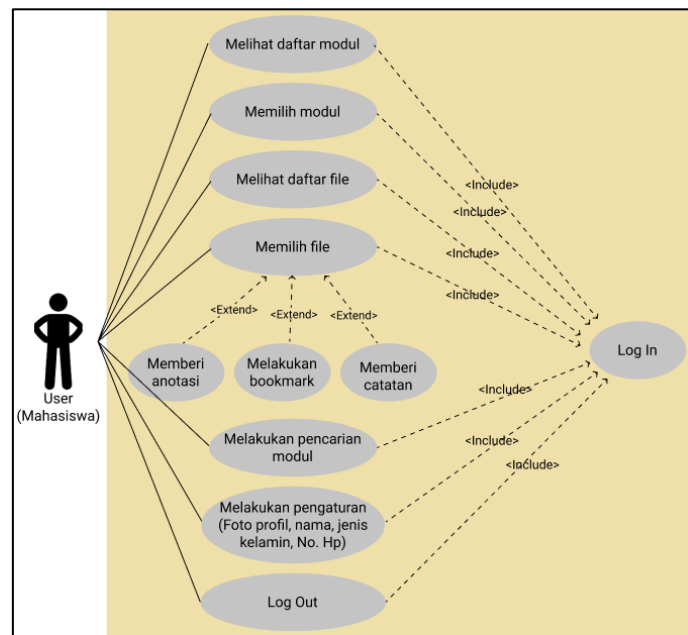


**Gambar 11.** *Drawer*

## 3.2. Use Case Diagram

Use case adalah salah satu pengganti metode analisis berorientasi objek dan desain berorientasi objek yang digunakan untuk memodelkan hubungan antara user dengan sistem. Pada aplikasi ini hanya menggunakan use case yang simpel

dalam penggunaannya. Use case aplikasi pada penelitian ini dijabarkan pada Gambar .



**Gambar 12.** Use Case Diagram

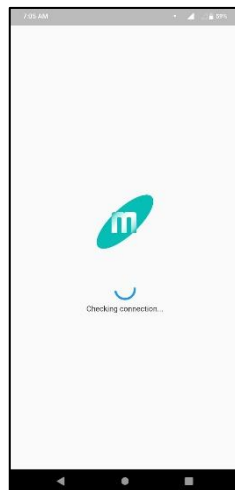
## **BAB IV**

### **HASIL DAN PEMBAHASAN**

#### **4.1. Implementasi Tampilan Antarmuka**

Tampilan antarmuka yang diimplementasikan pada bagian ini adalah seperti yang sudah dirancang di proposal yang berbentuk mock-up dan sudah diterjemahkan ke dalam kode program berbahasa Dart.

Pada saat aplikasi dijalankan akan mengecek koneksi terlebih dahulu. Kemudian akan mengecek apakah ada token yang tersimpan di perangkat maka akan langsung masuk ke halaman *Home*, jika tidak ada maka akan masuk ke halaman *login*. Berikut tampilan awal saat aplikasi dijalankan disajikan dalam pada Gambar

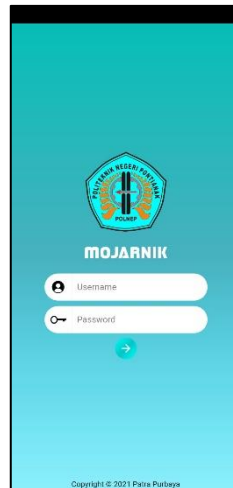


**Gambar 13.** Cek koneksi

Kemudian berikuk halaman-halaman lain yang diimplementasikan menggunakan kode pemrograman.

##### **4.1.1. Halaman *Login***

Halaman ini dibuat menggunakan 1 Image untuk Logo Politeknik Negeri Pontianak, 1 Text untuk judul aplikasi, dan 2 TextField untuk username dan password . Hasil halaman disajikan pada **Gambar 14**.



**Gambar 14.** Halaman *Login* Baru

#### 4.1.2. Halaman *Home*

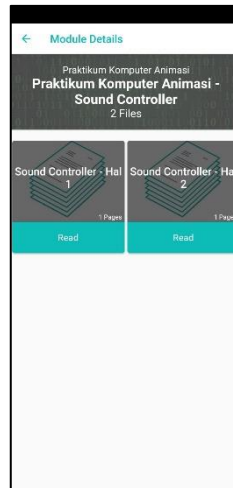
Halaman ini dibuat menggunakan Scaffold dan di bagian body ditampilkan berdasarkan index terpilih dari suatu array yang diisi oleh beberapa halaman berupa menu drawer . Hasil halaman disajikan pada **Gambar 15** sekaligus menampilkan modul.



**Gambar 15.** Halaman *Home* Baru

#### 4.1.3. Halaman *Detail Modul*

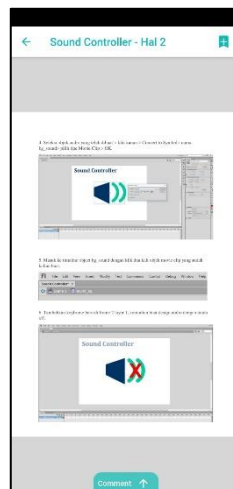
Halaman ini menampilkan detail dari modul yang dipilih dan detail-detail modul tersebut ditampilkan menggunakan Stateless Widget yang diciptakan sendiri oleh penulis untuk memudahkan penulisan kode. Hasil implementasi kode ditampilkan pada **Gambar 16**



**Gambar 16.** Halaman Detail Modul Baru

#### 4.1.4. Halaman *Reading*

Halaman ini dibuat menggunakan library Syncfusion untuk menampilkan dokumen pdf berupa url yang kemudian diakses dan dibaca dengan tipe data Uint8List agar dapat diberikan fitur anotasi. Hasil halaman ditampilkan pada **Gambar 17**



**Gambar 17.** Halaman *Reading* Baru

#### 4.1.5. Halaman *Search*

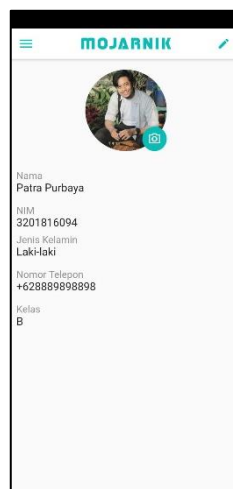
Pada halaman ini AppBar diberikan TextField sebagai alat input untuk mencari modul berdasarkan mata kuliah atau judul modul. Hasil halaman disajikan pada **Gambar 18**.



**Gambar 18.** Halaman *Search* Baru

#### 4.1.6. Halaman *Settings*

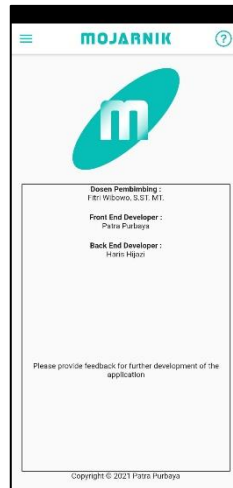
Halaman pengaturan menggunakan data dari API yang diambil kemudian dipasang ke Text untuk menampilkan dan apabila dalam mode edit, maka data tersebut akan dimasukkan ke dalam TextField, RadioButton. Serta gambar dapat diubah menggunakan foto galeri atau kamera. Hasil disajikan pada **Gambar**



**Gambar 19.** Halaman *Settings* Baru

#### 4.1.7. Halaman *About*

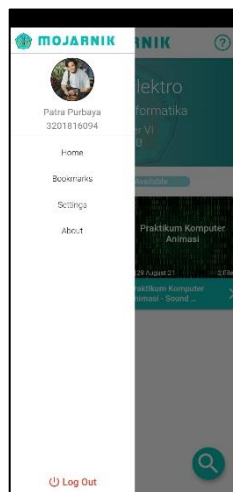
Halaman ini memuat tampilan biasa yang menampilkan deskripsi singkat mengenai aplikasi ini. Hasil halaman disajikan pada **Gambar 20**.



**Gambar 20.** Halaman *About* Baru

#### 4.1.8. *Drawer*

Menu yang menampung beberapa widget sebagai navigasi antarmenu yang jika ditekan akan berpindah ke halaman lain serta identitas singkat dari pengguna yang sudah *login*. Tampilan disajikan pada **Gambar 21**.



**Gambar 21.** Drawer Baru

## 4.2. Pengambilan Data API

Pengambilan data API diperlukan untuk mendapatkan data-data yang diperlukan seperti pada tabel di bawah ini.

**Tabel 1.** Data API

No.	Endpoint API (URL)	Deskripsi	Metode HTTP
1.	/api/token-auth/	Autentikasi	POST



2.	/api/accounts/customuser/	List Pengguna	GET
3.	/api/accounts/customuser/<pk>/	Spesifikasi Pengguna	GET, PATCH
4.	/api/accounts/profilmahasiswa/	List Profil Mahasiswa	GET
5.	/api/accounts/profilmahasiswa/<pk>/	Spesifikasi Profil Mahasiswa	GET, PATCH
6.	/api/akademik/jurusan/	List Jurusan	GET
7.	/api/akademik/jurusan/<pk>/	Spesifikasi Jurusan	GET
8.	/api/akademik/programstudi/	List Program Studi	GET
9.	/api/akademik/programstudi/<pk>/	Spesifikasi Program Studi	GET
10.	/api/akademik/matakuliah/	List Mata Kuliah	GET
11.	/api/akademik/matakuliah/<pk>/	Spesifikasi Mata Kuliah	GET
12.	/api/emodul/emodul/	Liat Modul	GET
13.	/api/emodul/emodul/<pk>/	Spesifikasi Modul	GET
14.	/api/emodul/emoduldetail/	List Detail Modul	GET
15.	/api/emodul/emoduldetail/<pk>	Spesifikasi Detail Modul	GET
16.	/api/emodul/emodulbookmark/	List Bookmark	GET, POST
17.	/api/emodul/emodulbookmark/<pk>/	Spesifikasi Bookmark	GET
18.	/api/emodul/emodulcomment/	List Komentar	GET, POST

19.	/api/emodul/emodulcomment/<pk>/	Spesifikasi Komentar	GET
-----	---------------------------------	-------------------------	-----

Pengambilan data API dilakukan dengan menggunakan HTTP yang berasal dari library http. Metode HTTP yang digunakan terdiri dari get untuk mengambil data, post untuk mengirim data, patch untuk mengubah data, kemudian menggunakan *multipart request* untuk mengupload foto sekaligus menjadi foto profil.

Terdapat beberapa end-point yang sama namun bisa saja berbeda fungsi seperti get/patch customUser, get/patch profilMahasiswa sehingga penting untuk memperhatikan method yang digunakan.

Untuk mendapatkan response, terdapat beberapa cara yang bisa digunakan. Berikut 3 cara untuk mengolah data response yang dapat digunakan :

#### **Mengambil single data menggunakan objek json**

```
If (respon.statusCode==200) {
    var rawJson = response.body;
    String jsonDataString = rawJson.toString();
    final json = jsonDecode(jsonDataString);
    contohHasil = json["namaField"];
}
```

#### **Mengambil list data menggunakan Future<Map<String, dynamic>>**

Cara ini digunakan saat programmer menggunakan FutureBuilder untuk melakukan request saat aplikasi dijalankan

```
if (response.statusCode == 200) {
    List mapResponse=json.decode(response.body);
    return
    mapResponse.map((e)=>ClassTertentu.fromJson(e))
    ).toList().cast();
}
```

#### **Mengambil list data kemudian dijadikan suatu variabel**

```

List<ClassTertentu> namaVariabel =[];
if (response.statusCode == 200) {
    var jsonData = jsonDecode(response.body);
    for (var i = 0; i < jsonData.length; i++) {

namaVariabel.add(ClassTertentu.fromJson(jsonData[i]
));

```

#### 4.3. Implementasi Fungsi

Seperti yang sudah dijelaskan sebelumnya bahwa aplikasi ini dapat melakukan *login*, memilih modul, memilih modul detail, membaca dokumen, memberikan bookmark, memberikan komentar, melakukan pencarian, pengaturan profile, logout, dan memberikan anotasi,.

Berikut beberapa fungsi yang terdapat pada fungsi aplikasi :

##### ***Login***

*Login* dilakukan sebagai autentikasi pengguna yang memiliki akses untuk masuk ke dalam aplikasi. Dengan struktur kode seperti di bawah ini:

```

import 'package:http/http.dart' as http;
Login() {
http.Response response = await http.post(Uri.parse(url),
body: {
    "username":string,
    "password":string,
});

```

##### **Memilih Modul dan Memilih Detail Modul**

Sebelum memilih modul, tentunya modul-modul harus ditampilkan pada halaman aplikasi. Metode yang digunakan untuk mengambil modul adalah *get*. Namun untuk menampilkan modul-modul perlu adanya widget yang digunakan sebagai tampilan dari modul dan perlu dilakukan pemetaan pada saat return data modul. Cara yang sama juga digunakan untuk mengambil data detail modul, hanya saja perbedaannya terletak pada parameter di detail modul membutuhkan id modul. Dengan struktur kode seperti di bawah ini:

```
import 'package:http/http.dart' as http;
Future<List<TipeData>> namaMethod() async{
  http.Response response = await http.get(Uri.parse(url),
  headers: {
    "Authorization:"token "+string,
  });
}
```

Perlu adanya token adalah sebagai adanya akses yang dimiliki user saat setelah *login* kemudian token tersebut akan disimpan di perangkat agar pada saat akun yang sudah pernah *login* tidak perlu *login* kembali.

### **Membaca Dokumen**

Membaca dokumen pada aplikasi adalah fitur utama yang dimiliki. Tipe dokumen yang akan ditampilkan berbentuk pdf dan sumber utama pdf yang ditampilkan adalah dari link hasil response API.

### **Memberikan Bookmark dan Memberikan Komentar**

Method http yang digunakan antara kedua fitur ini hampir sama yaitu GET. Namun perbedaan terletak pada parameter yang akan dimasukkan. Dengan struktur kode seperti di bawah ini:

```
import 'package:http/http.dart' as http;
namaMethod() async{
  http.Response response = await http.post(Uri.parse(url),
  headers: {
    "Authorization:"token "+string,
  },
  Body:{
    "parameter1":parameter1,
  },
  );
}
```

### **Pencarian Modul**

Pencarian modul digunakan untuk mencari modul tertentu di antara kumpulan modul yang. Implementasi sama seperti melakukan get list modul, namun seiring teks didalam textfield *search* berubah maka data yang ditampilkan akan berubah karena data akan disaring berdasarkan teks yang

berada di dalam textfield. Untuk saat ini, pencarian dapat dilakukan dengan berdasarkan mata kuliah dan judul modul.

### Mengubah profil mahasiswa

Method http yang digunakan dari fitur ini adalah PATCH, dimana end-point ditambahkan dengan primary key dan ubah data yang perlu diubah berdasarkan kondisi tertentu. Fitur ini sama-sama digunakan untuk mengubah user ataupun profil mahasiswa. Dengan struktur kode sebagai berikut :

```
import 'package:http/http.dart' as http;
namaMethod() async{
  http.Response response;
  response = await http.patch(Uri.parse(url+ primaryKey),
  headers: {
    "Authorization:"token "+string,
  },
  Body:{
    "parameter1":parameter1,
    "parameter2":parameter2,
    "foto": await MultipartFile.fromFile(imagePath),
  },
  );
}
```

### Logout

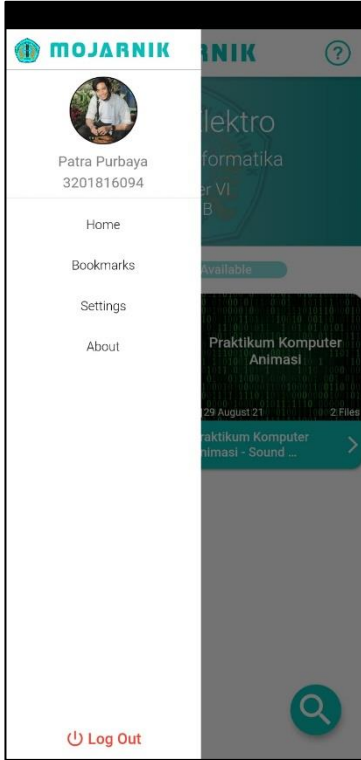
Fitur log out ini dilakukan dengan mereset token dari akun yang tersimpan di perangkat, sehingga apabila ingin masuk ke aplikasi pengguna harus *login* kembali menggunakan akun yang valid. Setelah pengguna menekan tombol logout, aplikasi akan langsung menampilkan halaman *login*.

## 4.4. Uji Coba

Metode yang akan digunakan dalam pengujian aplikasi ini adalah menggunakan Blackbox Testing. Berikut ini beberapa skenario sudah disesuaikan berdasarkan fitur-fitur yang dimiliki aplikasi

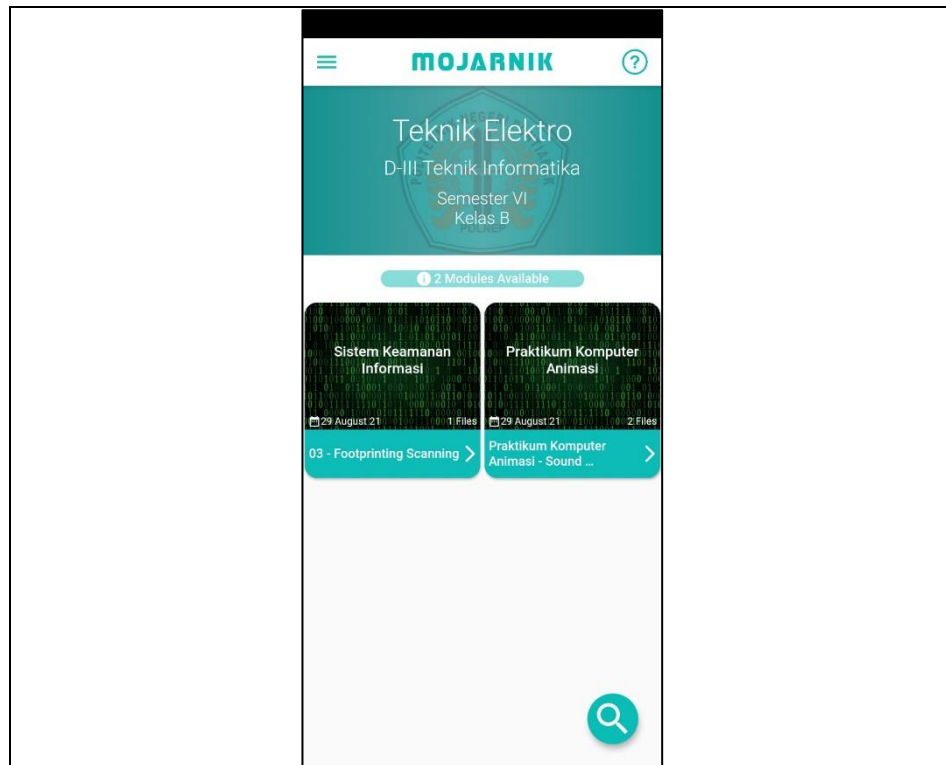
### 4.4.1. Login

API Endpoint	/api/token-auth
Metode HTTP	POST

Input	Username, password
Respon yang diharapkan	<pre>{   "token": str,   "user_id": int }</pre>
Valid	Ya
Hasil: Berhasil <i>login</i> dengan menampilkan profil pengguna 	

#### 4.4.2. Menampilkan Modul

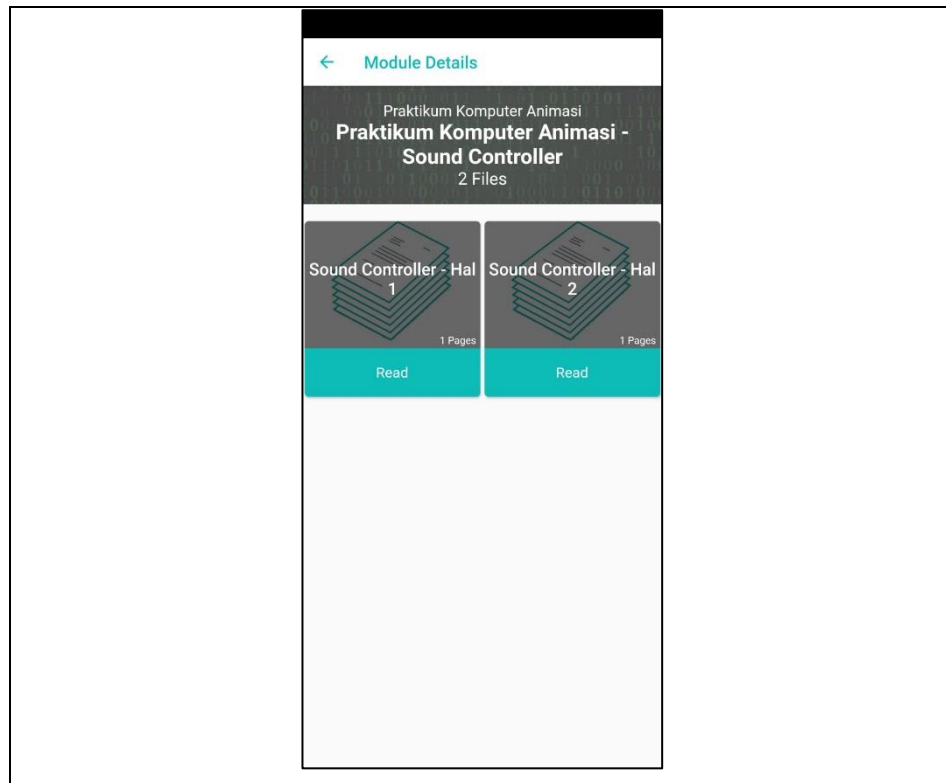
API Endpoint	/api/emodul/emodul
Metode HTTP	GET
Input	Token
Respon yang diharapkan	<pre>[{   "id": int,   "judul": str,   "jumlah_modul": int,   "cover": str link gambar,   "tanggal": DateTime,   "mata_kuliah": int }]</pre>
Valid	Ya
Hasil: Berhasil menampilkan modul	



Modul yang ditampilkan sudah difilter sesuai program studi dan semester dari masing-masing mahasiswa. Proses filtering dilakukan di dalam kode program menggunakan id mata kuliah pada modul, list mata kuliah, id program studi pada mata kuliah, dan program studi mahasiswa. Dan modul ditampilkan berdasarkan urutan modul terbaru.

#### 4.4.3. Menampilkan Detail Modul

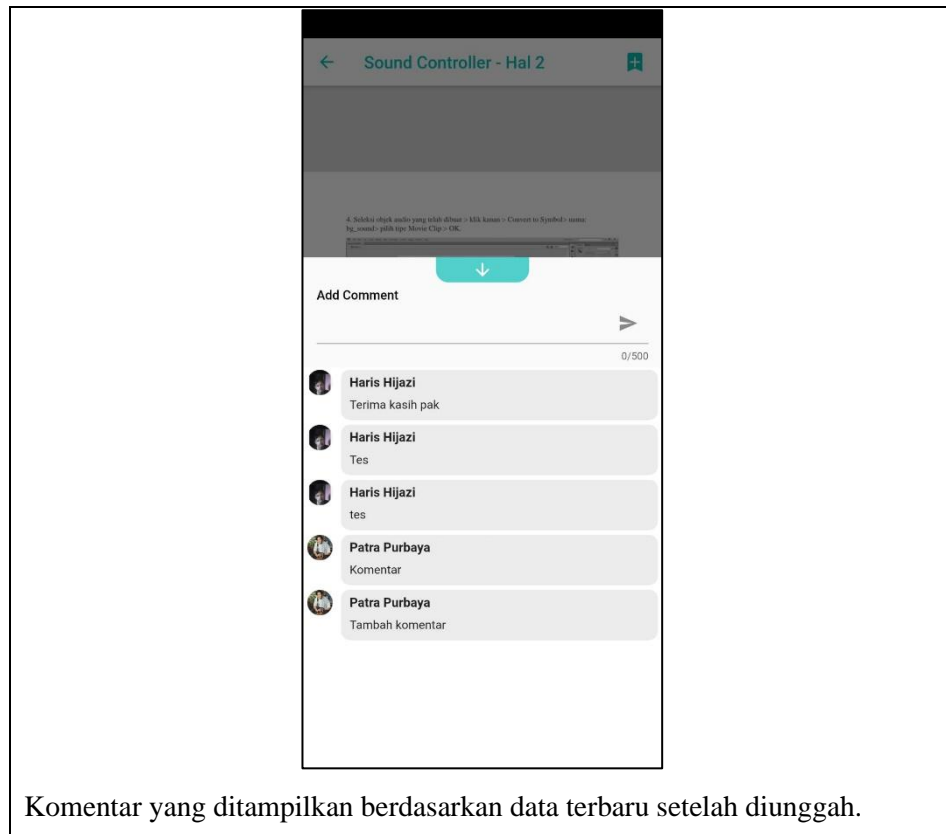
API Endpoint	/api/emodul/emoduldetail
Metode HTTP	GET
Input	Token
Respon yang diharapkan	<pre>{   "id": int,   "judul": str,   "jumlah_halaman": int,   "file": str link pdf,   "emodul": int }</pre>
Valid	Ya
Hasil: Berhasil menampilkan modul detail	



#### 4.4.4. Memberi dan menampilkan komentar

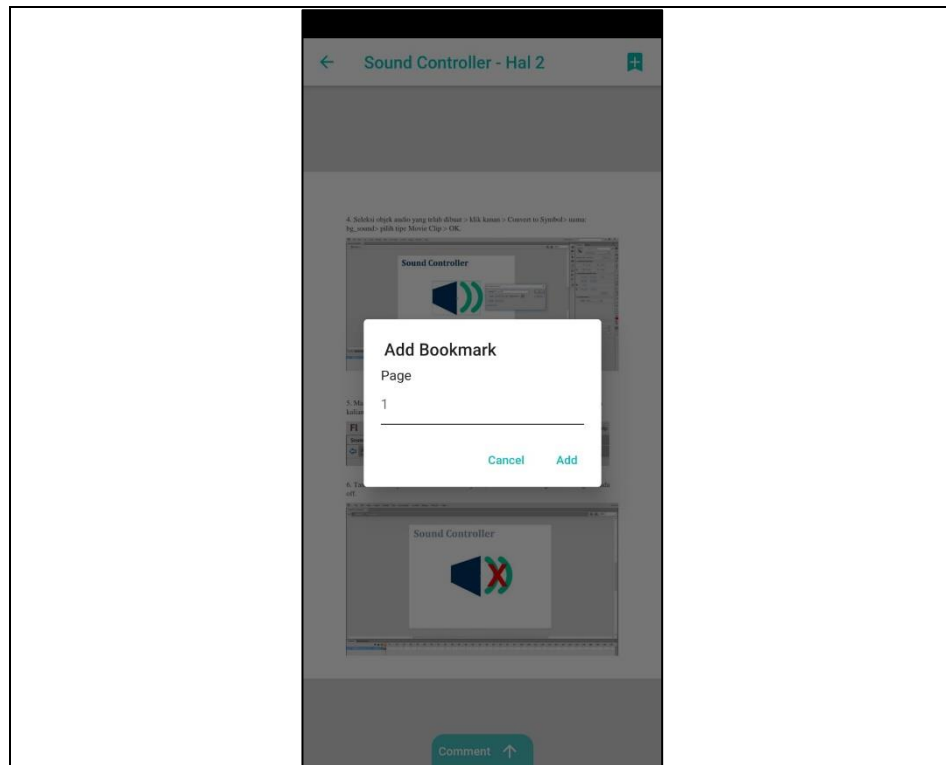
API Endpoint	/api/emodul/emodulcomment
Metode HTTP	POST, GET
Input	Token, user, modulDetail, modul
Respon yang diharapkan	<pre>{   "id": int,   "comment": str,   "dokumen": int,   "user": int }</pre>
Valid	Ya
Hasil: Berhasil menambahkan dan menampilkan komentar	





#### 4.4.5. Memberi dan menampilkan bookmark

API Endpoint	/api/emodul/emodulbookmark
Metode HTTP	POST, GET
Input	Token, user, halaman, dokumen, id
Respon yang diharapkan	<pre>{   "id": int,   "halaman": int,   "tanggal": DateTime,   "dokumen": int   "user": int }</pre>
Valid	Ya
Hasil: Berhasil memberi bookmark	



Berhasil menampilkan bookmark



Bookmark ditampilkan berdasarkan data terbaru setelah penambahan bookmark.

#### 4.4.6. Melakukan *search*

API Endpoint	/api/emodul/emodul
Metode HTTP	GET
Input	Token
Respon yang diharapkan	<pre>[{   "id": int,   "halaman": int,   "tanggal": DateTime,   "dokumen": int   "user": int }]</pre>
Valid	Ya

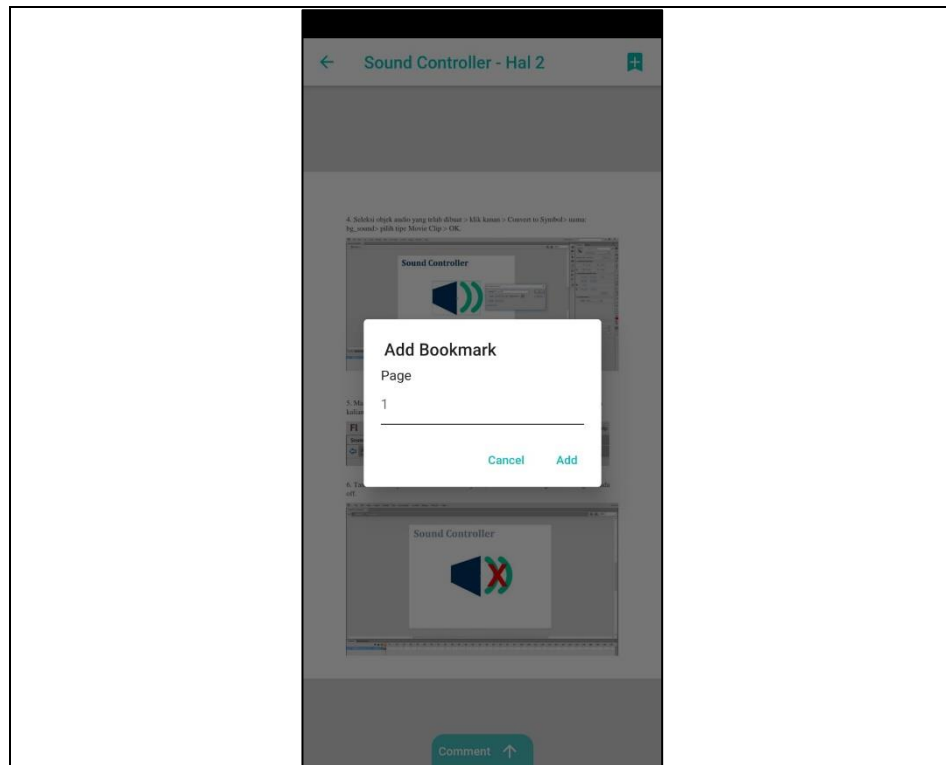
Hasil: Berhasil menampilkan modul di *search*



Modul yang ditampilkan sudah difilter sesuai program studi dan semester dari masing-masing mahasiswa. Proses filtering dilakukan di dalam kode program menggunakan id mata kuliah pada modul, list mata kuliah, id program studi pada mata kuliah, dan program studi mahasiswa. Dan modul ditampilkan berdasarkan urutan modul terbaru.

#### 4.4.7. Mengubah profil

API Endpoint	/api/accounts/customuser/user_id/ /api/accounts/profilmahasiswa/profil_id
Metode HTTP	PATCH
Input	Token, first_name, last_name, gender, no_hp, kelas,foto, dll
Respon yang diharapkan	CustomUser{ "id": int, "last_login": DateTime, "is_superuser": bool, "username": str, "first_name": str, "last_name": str, "email": str, "is_staff": bool, "is_active": bool, "date_joined": DateTime "role": int, "gender": int, "no_hp": str, "foto": str link foto, "profil_user_lengkap": false, "groups": [ 1 ], "user_permissions": []}
Valid	Ya
Hasil: Berhasil memberi bookmark	



Berhasil menampilkan bookmark



## **BAB V**

### **PENUTUP**

#### **5.1. Kesimpulan**

Aplikasi MOJARNIK dibuat sesuai dengan desain dan rancangan yang sudah direncanakan sebelumnya pada Proposal Tugas Akhir mengenai aplikasi ini. Bagus atau tidaknya tampilan antarmuka aplikasi tergantung dari masing-masing selera yang dimiliki pengguna. Implementasi dari kode program menghasilkan fungsi dan fitur yang dibutuhkan oleh pengguna.

#### **5.2.Saran**

Diperlukan adanya library yang memadai dalam pengembangan aplikasi untuk memaksimalkan performa aplikasi berbasis Flutter khususnya pada fitur anotasi untuk menampilkan anotasi di setiap halaman. Kualitas server juga sangat penting dalam aplikasi ini karena pengambilan data berasal dari back-end dan tentunya menggunakan database sebagai penyimpanan. Untuk saat ini build aplikasi hanya dapat berbentuk debug.apk dan belum dalam build release.apk, diharapkan ada pengembangan lebih lanjut terkait aplikasi dalam penambahan fitur dan tentunya aplikasi sudah berbentuk build release.

## DAFTAR PUSTAKA

- [1] A. Noertjahyana, “Studi Analisis Rapid Application Development Sebagai Salah Satu Alternatif Metode Pengembangan Perangkat Lunak,” *J. Inform.*, vol. 3, no. 2, pp. 64–68, 2002, doi: 10.9744/informatika.3.2.pp.64-68.
- [2] A. H. Moffitt, “November 2020,” *Am. J. Orthod. Dentofac. Orthop.*, vol. 158, no. 5, pp. 773.e1-773.e2, 2020, doi: 10.1016/j.ajodo.2020.09.006.
- [3] F. N. PAMBUDIANTO, “Analisis dan perancangan desain antarmuka aplikasi penjualan makanan sehat pada RSI Jemursari Surabaya dengan metode Design Sprint,” 2019.
- [4] R. D. Zaningsih, “Pengembangan Aplikasi E-Modul Mobile Pembelajaran Matematika Berbasis Android Studio Pokok Bahasan Matriks,” p. 129, 2018.
- [5] B. Liantriana, “Pengenalan Flutter.io Bagi Developer React,” *Medium, Skyshi*, 2018. [https://medium.com/skyshidigital/flutter-io-416c63020732#:~:text=Flutter adalah SDK untuk pengembangan,pada device iOS dan Android. \(accessed Mar. 20, 2021\).](https://medium.com/skyshidigital/flutter-io-416c63020732#:~:text=Flutter adalah SDK untuk pengembangan,pada device iOS dan Android. (accessed Mar. 20, 2021).)
- [6] Wikipedia, “Flutter (Perangkat Lunak),” *Wikipedia*, 2018. [https://id.wikipedia.org/wiki/Flutter\\_\(perangkat\\_lunak\) \(accessed Mar. 20, 2021\).](https://id.wikipedia.org/wiki/Flutter_(perangkat_lunak) (accessed Mar. 20, 2021).)
- [7] Ketut Krisna Wijaya, “Visual Studio Code: Aplikasi Editor Kode dari Microsoft untuk Windows, Linux, dan OS X,” *TechInAsia*, 2016. [https://id.techinasia.com/visual-studio-code-editor-kode-microsoft \(accessed Mar. 20, 2021\).](https://id.techinasia.com/visual-studio-code-editor-kode-microsoft (accessed Mar. 20, 2021).)

