

PROPOSAL TUGAS AKHIR
RANCANG BANGUN FRONT-END APLIKASI MOJARNIK
(MODUL PEMBELAJARAN ELEKTRONIK) DENGAN
MENGGUNAKAN FRAMEWORK FLUTTER DI POLITEKNIK
NEGERI PONTIANAK



OLEH :
PATRA PURBAYA
3201816094

PROGRAM STUDI D-III TEKNIK INFORMATIKA
JURUSAN TEKNIK ELEKTRO
POLITEKNIK NEGERI PONTIANAK
2021

HALAMAN PENGESAHAN

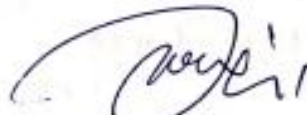
RANCANG BANGUN FRONT-END APLIKASI MOJARNIK (MODUL PEMBELAJARAN ELEKTRONIK) DENGAN MENGGUNAKAN FRAMEWORK FLUTTER DI POLITEKNIK NEGERI PONTIANAK

Oleh :

Patra Purbaya

3201816094

Dosen Pembimbing :



Fitri Wibowo, S.ST., MT.

NIP. 19851228 201504 1 002

Telah dipertahankan di depan penguji pada tanggal 25 Maret 2021 dan
dinyatakan memenuhi syarat sebagai Proposal Tugas Akhir

Dosen Penguji :

Penguji 1,



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

NIP. 19701016 199702 1 002

Penguji 2,



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

NIP. 19901020 201903 1 013



Mengetahui,

Ketua Program Studi
D-III Teknik Informatika,



Suheri, S.T., M.Cs.

NIP. 19830717 200812 1 005

Koordinator Tugas Akhir,



Lindung Siswanto, S.Kom., M.Eng.

NIP. 19840611 201903 1 012

HALAMAN PERNYATAAN ORISINALITAS

Saya yang bertandatangan di bawah ini :

Nama : Patra Purbaya

NIM : 3201816094

Jurusan/Program Studi : Teknik Elektro / D-III Teknik Informatika

Judul Proposal : Rancang Bangun Front-End Aplikasi Mojarnik (Modul Pembelajaran Elektronik) Dengan Menggunakan Framework Flutter Di Politeknik Negeri Pontianak

Menyatakan dengan sebenar-benarnya bahwa penulisan Proposal Tugas Akhir ini berdasarkan hasil penelitian, pemikiran dan pemaparan asli dari saya sendiri, baik untuk naskah proposal maupun kegiatan yang tercantum sebagai bagian dari Proposal 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, 25 Maret 2021

Yang membuat pernyataan

 
METERAI TEMPEL
0004 JK 153304999
Patra Purbaya
3201816094

DAFTAR TABEL

Tabel 1. Jadwal Penyelesaian Tugas Akhir.....	14
--	----

DAFTAR GAMBAR

Gambar 1. Diagram Alir Metodologi Penelitian	4
Gambar 2. Halaman <i>Login</i>	8
Gambar 3. Halaman <i>Home</i>	9
Gambar 4. Halaman <i>Detail Modul</i>	10
Gambar 5. Halaman <i>Reading</i>	11
Gambar 6. Halaman <i>Searc</i>	11
Gambar 7. Halaman <i>Settings</i>	12
Gambar 8. Halaman <i>About</i>	12
Gambar 9. <i>Drawer</i>	13
Gambar 10. <i>Use Case Diagram</i>	13

DAFTAR ISI

HALAMAN PENGESAHAN	i
HALAMAN PERNYATAAN ORISINALITAS	ii
DAFTAR TABEL	iii
DAFTAR GAMBAR.....	iv
1. Judul	1
2. Latar Belakang	1
3. Rumusan Masalah	2
4. Batasan Masalah.....	2
5. Tujuan Penelitian	2
6. Manfaat Penelitian	2
7. Metodologi Penelitian.....	3
7.1. Studi Literatur	4
7.2. Identifikasi Data	4
7.3. Pembuatan Tampilan Antarmuka Pengguna suatu Halaman.....	4
7.4. Pembuatan Kode Program	5
7.5. Uji Coba Aplikasi	5
8. Landasan Teori.....	5
8.1. Tinjauan Pustaka	5
8.2. Dasar Teori	6
8.2.1. Flutter.....	6
8.2.2. Visual Studio Code (VSCode)	7
9. Rancangan Sistem	7
9.1. User Interface	7
9.1.1. Halaman <i>Login</i>	7
9.1.2. Halaman Home.....	8

9.1.3.	Halaman Detail Modul	9
9.1.4.	Halaman Reading.....	9
9.1.5.	Halaman Search	10
9.1.6.	Halaman Settings	10
9.1.7.	Halaman <i>About</i>	11
9.1.8.	<i>Drawer</i>	11
9.2.	<i>Use case Diagram</i>	12
10.	Jadwal Penyelesaian Tugas Akhir	13
DAFTAR PUSTAKA.....		14

1. Judul

Rancang Bangun Front-End Aplikasi Mojarnik (Modul Pembelajaran Elektronik) Dengan Menggunakan Framework Flutter Di Politeknik Negeri Pontianak.

2. 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.

3. 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 ?

4. 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* yang terdiri dari mahasiswa.
3. Aplikasi menyajikan modul-modul untuk dibaca oleh mahasiswa.
4. Mahasiswa memberikan *bookmark*, anotasi, dan *notes* pada modul.
5. Dosen mengelola dan memberikan modul dari web (*Back-end*).
6. Data yang akan digunakan aplikasi diambil dari *API* yang dibuat oleh *Back-end Developer* melalui *URL* yang diberikan.
7. Bahasa Pemrograman yang digunakan adalah Dart.
8. *Tool* yang digunakan adalah Visual Studio Code.

5. 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.

6. Manfaat Penelitian

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

6.1. 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

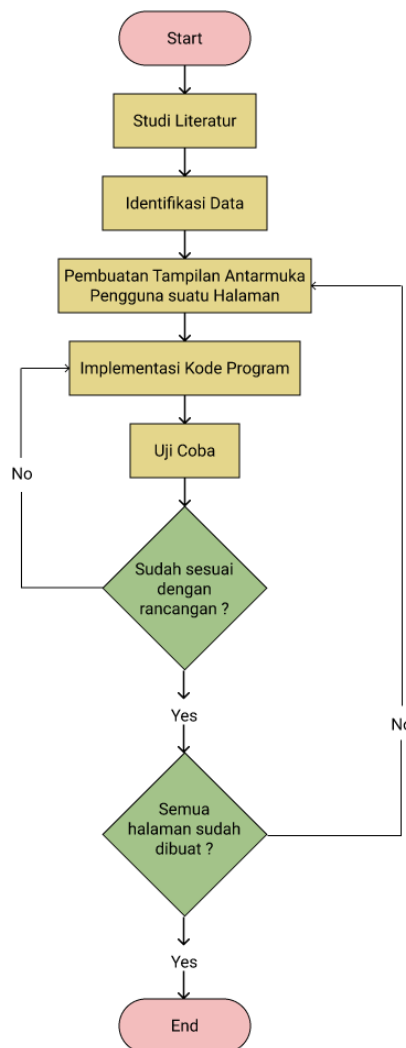
pada umumnya dapat menggunakan biaya lebih, memenuhi kapasitas ruang, bahkan memberatkan apabila dibawa.

6.2. 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.

7. Metodologi Penelitian

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].

7.1. 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.

7.2. 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.

7.3. 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.

7.4. 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.

7.5. 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.

8. Landasan Teori

Bagian ini terdiri atas dua bagian, yaitu tinjauan pustaka dan dasar teori.

8.1. 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 dan iOS.

8.2. Dasar Teori

Aplikasi ini dikembangkan menggunakan Flutter sebagai *framework* untuk membangun tampilan dan Visual Studio sebagai *code editor*.

8.2.1. 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. Foundation Library

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*.

8.2.2. Visual Studio Code (VSCode)

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.

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].

9. Rancangan Sistem

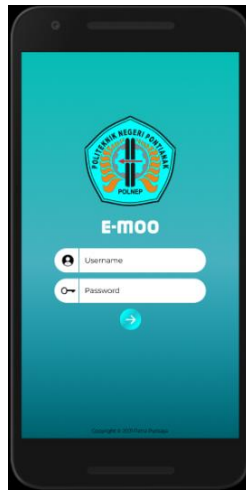
Rancangan sistem yang dibuat dibagi menjadi 2 bagian yaitu :

9.1. User Interface

User Interface atau tatap muka pengguna adalah tampilan yang akan menjadi penghubung antara sistem dan *user* dalam mengoperasikan aplikasi. Dalam sebuah aplikasi terdapat beberapa halaman yang memiliki isi dan fungsinya masing-masing. Berikut adalah beberapa *mock-up* beserta penjelasan secara singkat dari tiap halaman.

9.1.1. Halaman Login

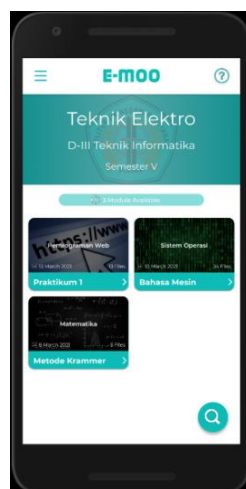
Pada halaman ini, *user* memasukkan *username* dan *password* sebagai syarat agar dapat masuk ke halaman utama aplikasi. Dari *username* dan *password* yang dimasukkan kemudian akan dicek di *database* apakah ada akun yang memiliki *username* dan *password* tersebut. Apabila benar, maka akan langsung masuk ke halaman utama (*Home Page*). *Mock-up* halaman disajikan pada **Gambar 2**.



Gambar 2. Halaman *Login*

9.1.2. Halaman Home

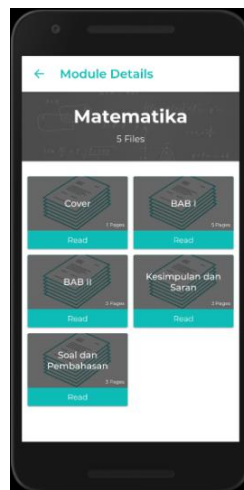
Setelah *user* melakukan *login*, maka layar akan menampilkan halaman utama (*Home Page*) dimana *user* dapat menemukan modul yang ditampilkan berdasarkan jurusan, program studi, dan semester masing-masing dari mahasiswa. Modul yang terdapat di database akan tersaring secara otomatis saat mengecek program 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 3**.



Gambar 3. Halaman *Home*

9.1.3. Halaman Detail Modul

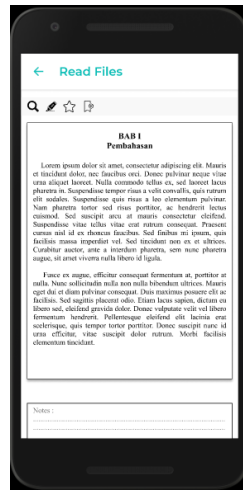
Halaman detail modul akan menampilkan beberapa *file* yang dipecah menjadi beberapa bagian dari bentuk asalnya. Dalam satu *file* bisa saja terdapat beberapa halaman lagi di dalamnya agar *user* dapat membaca secara terstruktur dan tidak merasa malas terlebih dahulu saat melihat halaman yang terlalu banyak. Kemudian apabila salahsatu *file* disentuh maka layar akan berpindah menuju halaman membaca (*Reading Page*). *Mock-up* halaman disajikan pada **Gambar 4**.



Gambar 4. Halaman Detail Modul

9.1.4. Halaman Reading

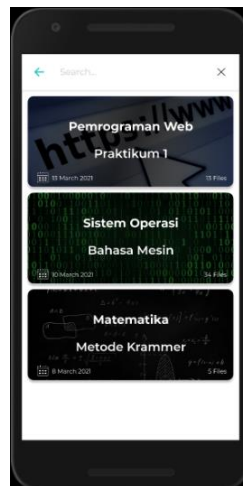
Pada halaman ini, *file* yang berasal dari suatu modul akan ditampilkan secara berurutan tergantung jumlah halaman yang dimiliki pada *file*. *User* dapat melakukan beberapa Tindakan terhadap *file* ini seperti membuat anotasi, menambah catatan, menambah *bookmark*, dan fitur lainnya. Apabila terdapat 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 5**.



Gambar 5. Halaman *Reading*

9.1.5. 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 6**.

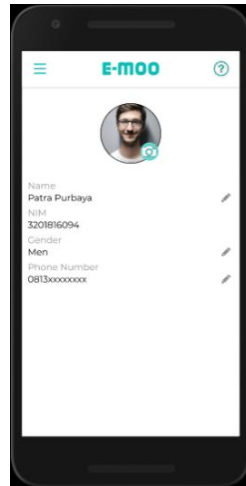


Gambar 6. Halaman *Search*

9.1.6. 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

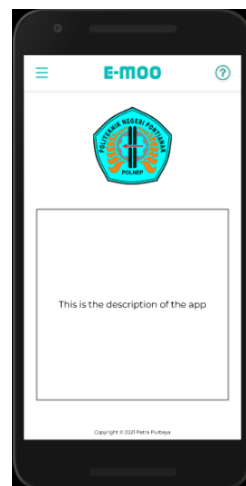
penyimpangan tertentu mengenai data asli dari mahasiswa yang dapat mengakibatkan kesalahan pada database. *Mock-up* halaman disajikan pada **Gambar 7**.



Gambar 7. Halaman *Settings*

9.1.7. 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 8**.

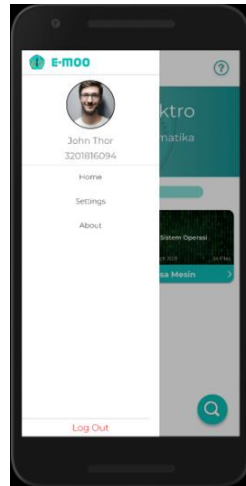


Gambar 7. Halaman *About*

9.1.8. Drawer

Drawer ini berfungsi sebagai navigasi kepada *user* apabila ingin berpindah menuju halaman lain yang terdiri dari halaman utama (*Home Page*), halaman

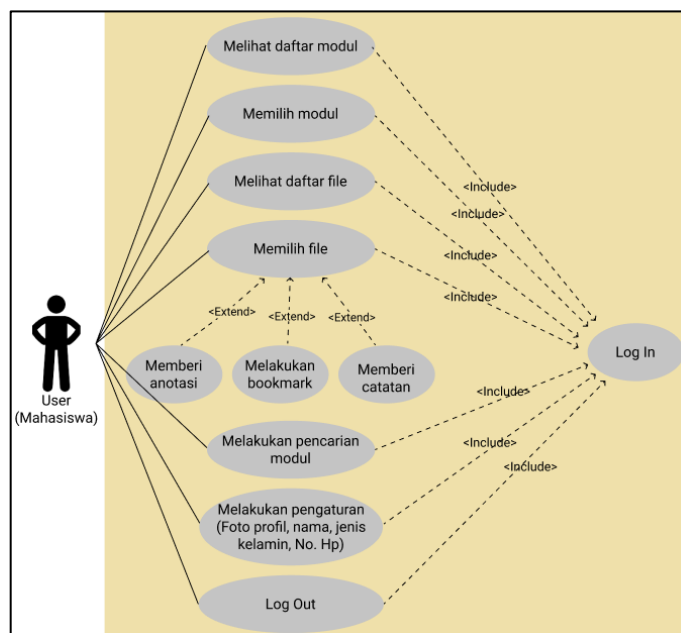
pengaturan (*Settings Page*), dan halaman tentang aplikasi (*About Page*). Dan *user* dapat melakukan *log out* melalui *drawer* ini apabila ingin mengeluarkan atau mengganti akun yang akan digunakan. *Mock-up* disajikan pada **Gambar 9**.



Gambar 9. *Drawer*

9.2. 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 10**.



Gambar 10 . *Use Case Diagram*

10. Jadwal Penyelesaian Tugas Akhir

Tabel 1. Jadwal Penyelesaian Tugas Akhir

No.	Kegiatan	Tahun 2021																											
		Maret				April				Mei				Juni				Juli				Agustus				September			
1	Studi Pustaka dan Penulisan Proposal TA	■	■	■	■																								
2	Seminar Judul dan Revisi				■	■	■																						
3	Studi Literatur							■																					
4	Identifikasi Data								■																				
5	Pembuatan Tampilan Antarmuka									■	■	■	■	■	■	■													
6	Implementasi Kode Program									■	■	■	■	■	■	■													
7	Uji Coba Sistem									■	■	■	■	■	■	■													
8	Penulisan Tugas Akhir																■												
9	Sidang Tugas Akhir																	■											

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_\(perangkat_lunak\)\)](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. [\(https://id.techinasia.com/visual-studio-code-editor-kode-microsoft\)](https://id.techinasia.com/visual-studio-code-editor-kode-microsoft) (accessed Mar. 20, 2021).