

SISTEM INFORMASI AKADEMIK BERBASIS MOBILE MENGUNAKAN FLUTTER

Studi Kasus: Sistem Akademik Universitas Kristen Immanuel

Yo'el Pieter Sumihar^{*1}, Arnan Abdiel Theopilus²

PROGRAM STUDI INFORMATIKA FAKULTAS SAINS DAN KOMPUTER
UNIVERSITAS KRISTEN IMMANUEL YOGYAKARTA

Jl Solo Km 11.11 Yogyakarta, telp/fax : (0274) 496256

e-mail: pieter.haro@ukrimuniversity.ac.id^{*1}, arnantheopilus@student.ukrimuniversity.ac.id²

Abstrak

Sistem informasi akademik (SIA) adalah sebuah sistem untuk mengelola dan menyebarkan informasi yang berkaitan tentang akademik mahasiswa. SIA harus memberikan informasi yang lengkap dan dapat diakses oleh semua mahasiswanya. Oleh karena itu perlu dilakukan pengembangan pada SIA berbasis android versi beta yang dimiliki oleh Universitas Kristen Immanuel (UKRIM) yang masih memiliki beberapa kekurangan. Flutter merupakan sebuah mobile app SDK (Software Development Kit) untuk membuat aplikasi android dan ios dari satu codebase dengan performa tinggi. Artinya kita hanya perlu mempelajari flutter untuk membangun aplikasi mobile untuk dua platform. Flutter memiliki design pattern yaitu bloc pattern, yang digunakan untuk memisahkan komponen presentation dan business logic yang akan memudahkan dalam perawatan ataupun penambahan fitur pada pengembangan aplikasi selanjutnya. Hasil dari penelitian ini aplikasi SIA UKRIM dapat dikembangkan dengan mengimplementasikan hampir semua fitur yang ada pada SIA UKRIM berbasis web kecuali upload tugas dengan menggunakan bloc pattern pada flutter.

Kata kunci: Pengembangan, Aplikasi Mobile, Flutter, Bloc Pattern

Abstract

Academic information system (SIA) is a system for managing and disseminating information related to student academics. SIA must provide complete and accessible information for all students. Therefore, it is necessary to develop a beta version of the Android-based SIA which is owned by Immanuel Christian University (UKRIM), which still has several shortcomings. Flutter is a mobile app SDK (Software Development Kit) for creating Android and iOS applications from a single, high-performance codebase. This means we only need to learn flutter to build mobile apps for two platforms. Flutter has a design pattern, namely the block pattern, which is used to separate presentation and business logic components which will make it easier to maintain or add features to further application development. The results of this study the SIA UKRIM application can be developed by implementing almost all the features that exist in web-based SIA UKRIM except for uploading tasks using the bloc pattern on flutter.

Keywords: Development, Mobile Application, Flutter, Bloc Pattern

1. PENDAHULUAN

Sistem Informasi Akademik (SIA) merupakan suatu teknologi untuk mengelola dan menyebarkan informasi yang berkaitan tentang akademik. SIA digunakan untuk menyimpan data dan manipulasi data tersebut menjadi informasi akademik di suatu institusi pendidikan yang akan menyajikan informasi bagi setiap aktor yang terlibat dalam sistem.

Penyajian informasi akademik harus sesuai dengan kebutuhan penggunanya agar informasi dapat diberikan dengan lengkap. Di samping itu proses bisnis di perguruan tinggi bersifat dinamis sehingga sistem harus di evaluasi dan dikembangkan secara berkala agar dapat mencapai tujuan tersebut. Begitu juga dengan SIA berbasis *android* yang ada di Universitas Kristen Immanuel (UKRIM).

Sistem beta yang dimiliki UKRIM saat ini masih memiliki beberapa kekurangan yang harus dikembangkan. SIA berbasis *android* yang ada saat ini tidak dapat dijalankan pada sistem operasi *iOS* sedangkan mahasiswa UKRIM ada yang menggunakan sistem operasi *iOS*. Selain itu sistem ini hanya memiliki fitur-fitur untuk melihat data akademik mahasiswa saja.

Dalam mengembangkan sistem informasi akademik ini diperlukan sebuah *tool* untuk mempercepat proses pengembangan yang sesuai dengan masalah yang ada pada sistem saat ini, seperti *flutter* yang merupakan sebuah *mobile app SDK (Software Development Kit)* untuk membuat aplikasi *android* dan *iOS* dari satu *codebase* dengan performa tinggi. Artinya peneliti hanya perlu mempelajari *flutter* untuk membangun aplikasi *mobile* untuk dua *platform* [1]. Selain *tool* dalam pengembangan sistem perlu menggunakan *design pattern* yang membuat pola struktur dan *coding* menjadi konsisten dan teratur. Dalam *flutter* sudah memiliki *design pattern* yaitu *bloc pattern*, yang digunakan untuk memisahkan komponen *presentation* dan *business logic* yang akan memudahkan dalam perawatan ataupun penambahan fitur pada pengembangan aplikasi selanjutnya [2].

2. METODE PENELITIAN

Pada tahap ini dilakukan pengumpulan kebutuhan secara lengkap dengan melakukan observasi, wawancara, dan studi pustaka kemudian dianalisis dan didefinisikan kebutuhan yang harus dipenuhi oleh program yang akan dibangun.

2.1 Analisis Kebutuhan Perangkat Lunak

Analisis sistem merupakan penguraian dari suatu sistem yang utuh ke dalam bagian-bagian komponen dengan maksud untuk mengidentifikasi kebutuhan sistem yang akan dibangun. Berikut analisis kebutuhan perangkat lunak pada pengembangan aplikasi SIA UKRIM.

a. Kebutuhan Fungsionalitas

Kebutuhan fungsionalitas yang akan dikerjakan pada pengembangan aplikasi ini adalah sebagai berikut.

1. Aplikasi memiliki fitur *home* yang didalamnya terdapat fungsi untuk menampilkan data informasi prodi berdasarkan program studi mahasiswa seperti pengumuman, berita, *event*, dan kalender akademik.
2. Aplikasi memiliki fitur *e-learning* yang didalamnya terdapat fungsi untuk menampilkan data pengumuman, diktat, tugas, silabus berdasarkan matakuliah yang diambil mahasiswa.
3. Aplikasi memiliki fitur presensi.
4. Aplikasi memiliki fitur KRS-Online.
5. Aplikasi memiliki fitur data akademik yang didalamnya terdapat fungsi untuk menampilkan data Kartu Hasil Studi (KHS), transkrip, biodata dan pembayaran.
6. Aplikasi memiliki fitur extra yang didalamnya terdapat fungsi download file dan ganti password.

b. Kebutuhan Non-fungsionalitas

Kebutuhan non fungsionalitas yang akan dikerjakan pada pengembangan aplikasi ini adalah aplikasi memiliki fungsi login dan logout otomatis jika token sudah expired.

2.2 Diagram Use Case

Diagram *use case* digambarkan dalam keadaan *user* telah melakukan *login*. *User* dapat melihat data pembayaran, detail pembayaran, KHS, biodata, transkrip, KRS-online, dan matakuliah yang didalamnya termasuk dapat melihat pengumuman matakuliah, tugas matakuliah, diktat matakuliah, silabus, dan melakukan presensi atau absen.



Gambar 2. Error! No text of specified style in document..1 Use Case Diagram

2.3 Use Case Narrative

Use case narrative merupakan dekrripsi secara tertulis mengenai semua peristiwa bisnis dan bagaimana akan berinteraksi dengan sistem untuk menyelesaikan suatu fungsi atau kebutuhan, Berikut adalah contoh use case narrative fitur home, fitur download file, dan fitur ganti password.

a. Fitur home.

Tabel Deskripsi Use Case : Melihat Informai Program Studi

Tabel 2.1 Dskripsi Use Case : Melihat Informasi Program Studi

| Nama Use Case | Melihat data Informasi dari Prodi |
|-----------------|-----------------------------------|
| Use Case ID | - |
| Primary Actor | Mahasiswa |
| Secondary Actor | - |

| | | |
|-----------------------------|--|--|
| Use Case Description | <i>Use case</i> dilakukan ketika mahasiswa membuka halaman home pada aplikasi | |
| Precondition | Aktor berada pada tampilan login dan akan masuk ke aplikasi. | |
| Trigger | Aktor menekan tombol home pada halaman login untuk masuk ke aplikasi. | |
| Basic Flow | Action | System Response |
| | Step 1: Mahasiswa berada pada tampilan login, lalu menekan tombol login. Step 3: Mahasiswa dapat melihat data informasi prodi sesuai prodi mahasiswa. | Step 2: Sistem akan menampilkan data informasi prodi baik itu pengumuman, berita, event, dan kalender akademik di halaman home jika berhasil melakukan login. |
| Error Flow | - | |
| Post Condition | Mahasiswa telah melihat data informasi prodi berdasarkan prodi mahasiswa. | |

b. Fitur Download File

Tabel Deskripsi Use Case : Download File

Tabel 2.2 Deskripsi Use Case : Download File

| | | |
|-----------------------------|--|--|
| Nama Use Case | Melakukan Download File | |
| Use Case ID | - | |
| Primary Actor | Mahasiswa | |
| Secondary Actor | - | |
| Use Case Description | <i>Use case</i> dilakukan ketika mahasiswa menekan salah satu materi pada halaman tugas | |
| Precondition | Aktor atau mahasiswa berada pada tampilan halaman tugas | |
| Trigger | Mahasiswa memilih dan menekan salah satu materi pada halaman tugas | |
| Basic Flow | Action | System Response |
| | Step 1: Mahasiswa berada pada tampilan tugas, lalu memilih salah satu data tugas pada <i>modal bottom sheet</i> Step 3: Mahasiswa dapat melihat data tugas yang di download | Step 2: Sistem akan melakukan proses download file pada data tugas yang dipilih |
| Error Flow | - | |
| Postcondition | Mahasiswa telah mendownload data tugas. | |

c. Fitur Ganti Password

Tabel Deskripsi Use Case : Ganti Password

Tabel 2.3 Deskripsi Use Case : Ganti Password

| | | |
|-----------------------------|--|--|
| Nama Use Case | Melakukan Ganti Password | |
| Use Case ID | - | |
| Primary Actor | Mahasiswa | |
| Secondary Actor | - | |
| Use Case Description | <i>Use case</i> dilakukan ketika mahasiswa menekan menu ganti password | |
| Precondition | Aktor atau mahasiswa berada pada tampilan halaman pengaturan | |

| | | |
|----------------------|---|--|
| Trigger | Mahasiswa menekan menu ganti password pada halaman pengaturan. | |
| Basic Flow | <p>Action</p> <p>Step 1: Mahasiswa berada pada tampilan pengaturan lalu memilih menu ganti password.</p> <p>Step 3: Mahasiswa memasukkan password lama, password baru, dan konfirmasi password baru.</p> <p>Step 5: Mahasiswa telah mengganti password.</p> | <p>System Response</p> <p>Step 2: Sistem akan membuka halaman ganti password.</p> <p>Step 4: Sistem akan melakukan proses ganti password. Dan akan menampilkan pesan hasil proses ganti password.</p> |
| Error Flow | - | |
| Postcondition | Mahasiswa telah mengganti password. | |

2.4 Rancangan Arsitektur

Rancangan arsitektur berisi tentang rancangan arsitektur pada aplikasi yang dibangun, meliputi rancangan arsitektur software dan rancangan arsitektur hardware.

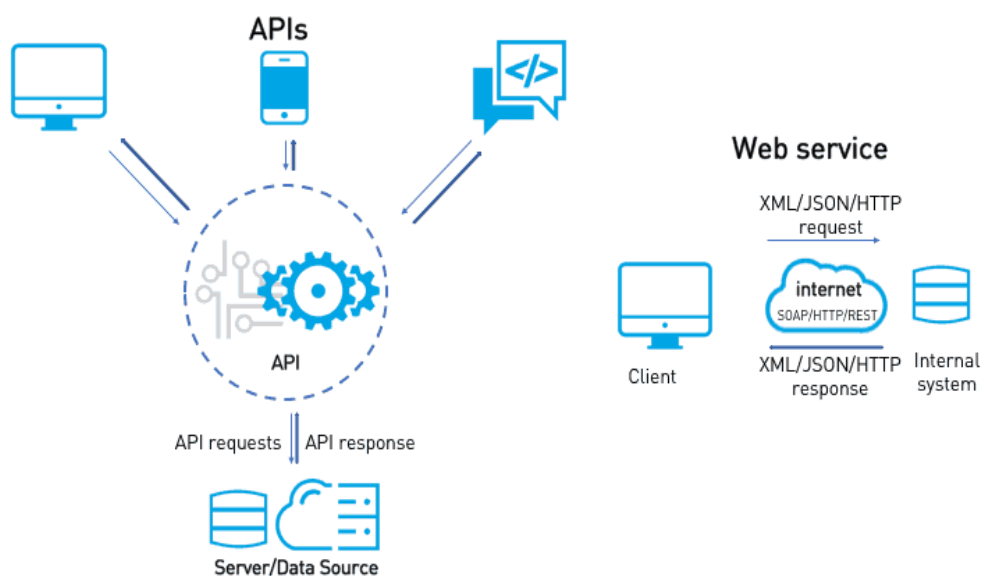
a. Arsitektur Software

Perancangan arsitektur software ini menggunakan BLoC Pattern (Business Logic Component). Design pattern ini akan membantu untuk memisahkan presentation (view) dengan business logic.

Pada penerapannya presentation (view) akan menerima interaksi dari pengguna untuk di proses ke dalam business logic sesuai dengan event yang dilakukan. Dalam business logic ini akan mengakses data baik dari API ataupun data yang disimpan secara lokal yang akan diteruskan kembali ke presentation sesuai dengan state yang ditentukan dalam business logic.

b. Arsitektur Hardware

Arsitektur hardware pada perancangan SIA UKRIM ini terdiri dari database, web server api, dan aplikasi atau client. Konsepnya adalah ketika aplikasi akan melakukan request data dengan database maka aplikasi harus melakukan request terlebih dahulu dengan web server api, setelah itu request akan diteruskan kedalam database untuk mendapatkan response yang akan dikembalikan ke web server api dan diteruskan lagi ke aplikasi atau client.



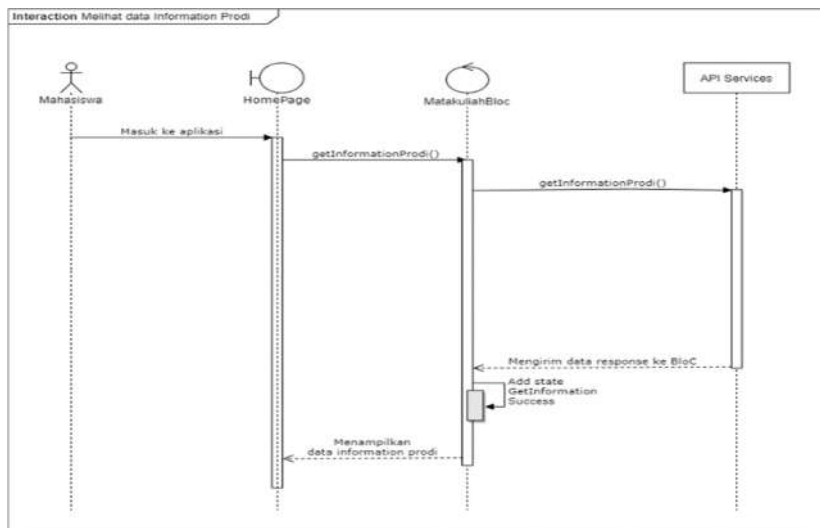
Gambar 2. ~~Error! No text of specified style in document.~~2 Arsitektur Hardware

2.5 Sequence Diagram

Sequence diagram adalah suatu diagram yang akan menggambarkan interaksi antar objek didalam dan disekitar sistem berupa message. Berikut beberapa contoh sequence diagram dari aplikasi SIA UKRIM.

a. Fitur Home

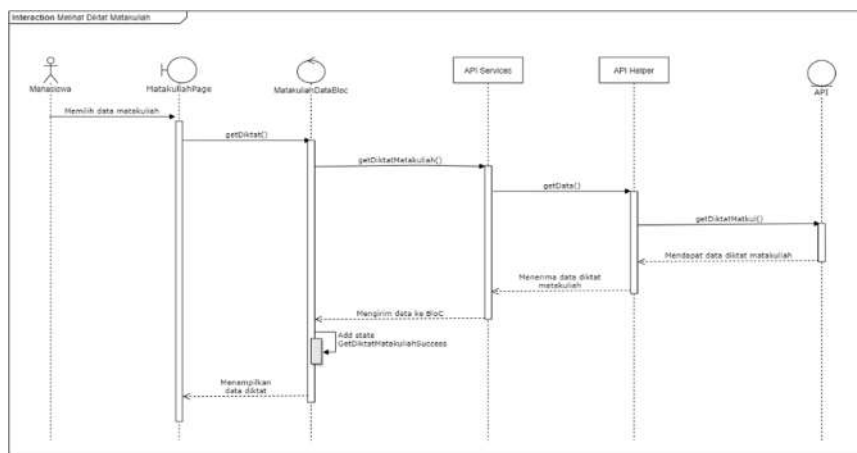
Sequence diagram fitur home dapat dilihat pada Gambar.



Gambar 2. **Error! No text of specified style in document.**3 Sequence diagram fitur home

b. Fitur E-learning (Data Diktat Matakuliah)

Sequence diagram fungsi melihat data diktat matakuliah dapat dilihat pada Gambar.



Gambar 2. **Error! No text of specified style in document.**4 Sequence diagram fungsi melihat data diktat matakuliah

2.6 Rancangan User Interface

Rancangan *user interface* aplikasi meliputi halaman fitur *home*, fitur *e-learning*, fitur presensi, fitur *KRS-Online*, fitur data akademik.

a. Halaman Home

Halaman untuk fitur *home* akan meliputi tampilan untuk data pengumuman, berita, *event*, dan kalender akademik dari program studi mahasiswa.

- b. Halaman Fitur E-learning
Bagian utama dari tampilan fitur ini adalah halaman matakuliah yang akan memiliki anak tampilan antara lain tampilan untuk data pengumuman, tugas, diktat, dan silabus dari matakuliah.
- c. Halaman Fitur Presensi
Halaman ini akan menampilkan data presensi mulai dari nama materi yang diajar, status presensi, tanggal absen , dan lain-lain.
- d. Halaman KRS-Online
Halaman ini terdiri dari dua bagian. Bagian *card header* yang akan digunakan untuk menampilkan nama mahasiswa, nim mahasiswa, dan jumlah sks yang dapat diambil. Bagian kedua adalah *list view* yang digunakan untuk membungkus *card* berisi data-data matakuliah yang dapat diambil pada saat KRS.
- e. Halaman Fitur Data Akademik
Halaman ini akan terbagi menjadi tiga bagian yaitu halaman transkrip untuk menampilkan data nilai keseluruhan, halaman KHS untuk menampilkan data kartu hasil studi setiap semester, halaman biodata untuk menampilkan data informasi pribadi, dan halaman pembayaran untuk menampilkan data pembayaran mahasiswa.
- f. Halaman Fitur Extra
Halaman ganti password akan menampilkan tiga inputan bagi user atau mahasiswa yang akan digunakan untuk memasukkan inputan password lama, password baru, dan mengulangi input password baru.

2.7 Rancangan Web Services Restful API

Web services restful api akan digunakan untuk menerima *request* data dari aplikasi SIA UKRIM dan *web services restful api* akan melakukan *request* ke *server* berkaitan dengan data yang di minta. Data yang diterima dari *server* akan dikembalikan ke aplikasi. Contoh rancangan endpoint yang digunakan untuk berkomunikasi dengan API diuraikan seperti berikut.

- a. Presensi
Endpoint yang terdapat pada presensi ada dua yaitu endpoint untuk mengambil data presensi berdasarkan *id_semester* dengan method GET.
[Base URL]mahasiswa/absen?id_semester=20161
Dan melakukan presensi dengan method POST dengan membawa parameter *id_absen_dosen*.
[Base URL]mahasiswa/absen?id_absen_dosen=1
- b. Ganti Password
[Base URL]mahasiswa/change-password
Endpoint ini digunakan untuk melakukan proses ganti password dengan method POST dan membawa data *nim mahasiswa*, data password baru, dan data password lama.

3. HASIL DAN PEMBAHASAN

3.1 API Helper

Api helper digunakan sebagai penghubung antara aplikasi SIA dengan *REST Api* UKRIM menggunakan *library* Dio. Api helper memiliki dua fungsi untuk melakukan interaksi dengan method *POST* dan *GET*.

- a. POST
Fungsi post data akan menerima lima parameter yaitu endpoint, basic auth, token, params, dan body.

```

Future<dynamic> postData(String endpoint, {String
  basicAuth, String token, Map<String, String> params,
  dynamic body}) async {
  try {
    final response = await _dio.post( baseUrl + endpoint,
data: body, queryParameters: params, options: Options(headers:
headers, followRedirects: false,)),
    );
    responseJson = response;
  }
}

```

Gambar Error! No text of specified style in document..5 API Helper POST

b. GET

Fungsi post data akan menerima lima parameter yaitu endpoint, token, params.

```

Future<dynamic> getData(String endpoint, String token,
  {Map<String, String> params}) async {
  //Header
  try {
    final response = await _dio.get(baseUrl + endpoint,
queryParameters: params, options: Options(
headers: headers, followRedirects: false, validateStatus:
(status) {return status <= 500;},),
    );
    responseJson = response;
  }
}

```

Gambar Error! No text of specified style in document..6 API Helper GET

3.2 Pemanggilan Event

Event ini akan di panggil ketika *user* atau pengguna melakukan interaksi pada *event* tersebut. Sebagai contoh pengguna melakukan sebuah *event* untuk melihat data presensinya, maka *event* tersebut yang akan mendefinikasikan proses yang akan dilakukan oleh *Bloc*.

```

if (titleClick == 'Presensi') {
  // Event get data presensi
  BlocProvider.of<PresensiBloc>(context).add(GetPresensi(idDosenS
emester: idDosen));
  BlocProvider.of<PageBloc>(context).add(GoToPresensiPage(matKul))
;
}

```

Gambar Error! No text of specified style in document..7 Pemanggilan Event

3.3 Api Services

Api services akan menghubungkan bagian *bloc* dan bagian *Api Helper* sebagai penyalur data dan melakukan *prepare data response API* sebelum digunakan oleh *bloc*. Sebagai contoh berikut adalah fungsi *getAbsen()* pada *Api Services*.

```
Future<AbsenResponse> getAbsen(String token, String id) async {
  var params = {'id_dosen_semester': id};
  final responseData = await _apiHelper.getData(
    endpointAbsenMhs, token, params: params);
  if (responseData.runtimeType != String) {
    return
      AbsenResponse.fromJson(responseData.data);
  } else {
    return
      AbsenResponse.withError(
        responseData.toString());
  }
}
```

Gambar **Error! No text of specified style in document.**8 API Services getAbsen()

3.4 Proses Bloc

Proses ini merupakan proses yang dilakukan setelah menerima sebuah *event*. *Event* tersebut akan diterjemahkan dan proses oleh *bloc*. Sebagai contoh proses *bloc* untuk melihat data presensi, maka proses yang dilakukan adalah melakukan *request* ke *API* melalui *ApiServices()* dengan membawa parameter yang sudah ditentukan. *Response* dari *request* tersebut akan menentukan *state* apa yang akan diberikan kembali kepada bagian *view* atau tampilan aplikasi. Contoh jika hasil *request* berhasil maka akan diberikan *state GetPresensiSuccess()* dengan membawa data presensi matakuliah yang diterima dari *request*.

```
if (event is GetPresensi) {
  yield GetPresensiLoading();
  try {
    String hasToken = cekToken(_authenticationBloc);
    final dataAbsen = await ApiService().getAbsen(
      hasToken.toString(), event.idDosenSemester
    );
    if (dataAbsen.absen.isNotEmpty) {
      yield GetPresensiSuccess(
        dataAbsen.absen
      );
    } else if (dataAbsen.absen.isEmpty) {
      yield PresensiIsEmpty();
    }
    authenticationBloc.add(UserLoggedOut());
  }
  } else {
    yield PresensiError(
      dataAbsen.error.toString(),
      event.idDosenSemester);
  }
} catch (e) {
}
```

Gambar **Error! No text of specified style in document.**9 Proses Bloc

3.5 Bloc Builder

Bloc Builder akan menerima perubahan *state* yang diberikan oleh *bloc*. *State* yang diberikan akan menentukan perubahan tampilan pada halaman. Sebagai contoh berikut adalah *bloc builder* pada fitur *e-learning*.

3.6 Tampilan Aplikasi

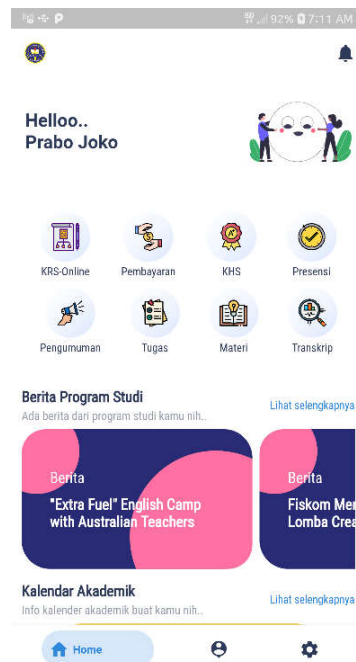
Beikut adalah tampilan aplikasi SIA UKRIM yang sudah dikembangkan menggunakan *flutter* dengan mengimplementasikan *bloc pattern*.

```
BlocBuilder<MatakuliahDataBloc, MatkulilahDataState>(
  builder: (context, matkulDataState) {
    if (matkulDataState is GetMatakuliahDataLoading) {}
    if (matkulDataState is GetMatakuliahDataError) {}
    if (matkulDataState is GetMatakuliahDataIsEmpty) {}
    if (matkulDataState is
      GetPengumumanMatakuliahSuccess) {}
    if (matkulDataState is GetTugasMatakuliahSuccess) {}
    if (matkulDataState is GetDiktatMatakuliahSuccess) {}
```

Gambar *Error! No text of specified style in document.*10 Bloc Builder

a. Tampilan Home

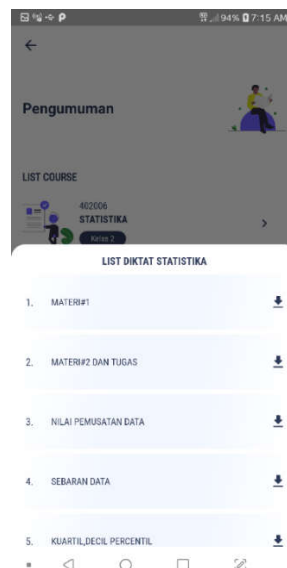
Tampilan halaman home adalah tampilan halaman utama dari aplikasi SIA UKRIM.



Gambar *Error! No text of specified style in document.*11 Tampilan Home

b. Tampilan E-Learning

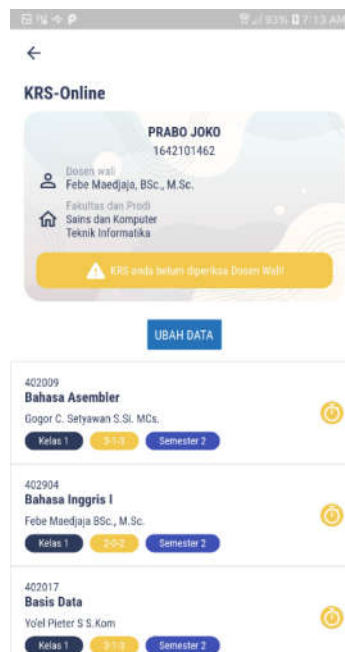
Berikut adalah tampilan salah satu fungsi pada fitur e-learning yaitu melihat data pengumuman dari matakuliah yang di ambil oleh mahasiswa dan setiap informasi peng umumannya dapat diunduh secara langsung oleh pengguna.



Gambar Error! No text of specified style in document..12 Tampilan E-Learning

c. Tampilan KRS-Online

Berikut adalah tampilan halaman KRS-Online ketika mahasiswa sudah menginputkan data KRS. Maka halaman ini akan menampilkan status KRS dan data-data matakuliah yang sudah dipilih sebelumnya.



Gambar Error! No text of specified style in document..13 Tampilan KRS-Online

4. KESIMPULAN

Berdasarkan dari hasil studi literatur, analisis, perancangan, implementasi dan pengujian pada pengembangan sistem ini, maka kesimpulan yang didapat adalah sebagai berikut. Pengembangan aplikasi SIA UKRIM sudah berhasil dikembangkan dengan mengimplementasikan hampir semua fitur yang ada pada SIA UKRIM berbasis *web* dengan menggunakan *bloc pattern* pada *flutter*. Proses pengimplementasian *bloc pattern* dalam pengembangan aplikasi SIA UKRIM menggunakan *flutter* adalah sebagai berikut. (a) Membuat *event* yang akan dilakukan oleh pengguna. (b) Mendefinisikan *state-state* yang akan terjadi ketika suatu *event* dilakukan. (c) Membuat *bloc* yang akan mengatur setiap *event* kemudian memprosesnya dan menerjemahkannya kembali ke *state-state* yang telah terdefinisi untuk menentukan tampilan yang akan diberikan ke pengguna. *Flutter* mempercepat proses pengembangan karena *flutter* memberikan dua aplikasi yang dapat dijalankan pada sistem operasi *iOS* dan *android*. Berdasarkan hasil pengujian aplikasi SIA UKRIM berbasis *mobile* menggunakan *flutter* dapat menjalankan semua fitur yang dibuat berdasarkan fungsi-fungsinya dengan baik.

5. SARAN

Adapun saran-saran yang dapat diberikan untuk pengembangan dan perbaikan sistem ini selanjutnya dari aplikasi SIA UKRIM, diantaranya adalah: Mengembangkan atau menambahkan API supaya lebih banyak fitur-fitur yang di dapat diimplementasikan pada aplikasi SIA UKRIM. Menambahkan fitur *notification* untuk memberikan pemberitahuan kepada mahasiswa seperti informasi terbaru jika dosen wali sudah menyetujui KRS dan lain-lain. Sehingga mahasiswa dapat mendapat informasi secara cepat tanpa harus membuka aplikasi SIA UKRIM terlebih dahulu

6. DAFTAR PUSTAKA

- [1] Suryono, Sigit dan Hardiansah. 2020. *Falling In Love with Flutter: Panduan Lengkap membuat aplikasi Android dan iOS dari Nol*. Surelabs Pub.
- [2] Syaputra, Rizki & Ganda, Yusmi Putra Wira. 2019. *Happy Flutter: Membuat Aplikasi Andorid dan iOS dengan Mudah menggunakan Flutter - UDACODING*. Tangerang Selatan: UDACODING.