

# SKRIPSI

«JUDUL BAHASA INDONESIA»



David Christopher

NPM: 2017730015

PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS  
UNIVERSITAS KATOLIK PARAHYANGAN

«tahun»



# UNDERGRADUATE THESIS

«JUDUL BAHASA INGGRIS»



David Christopher

NPM: 2017730015

DEPARTMENT OF INFORMATICS  
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES  
PARAHYANGAN CATHOLIC UNIVERSITY

«tahun»



# LEMBAR PENGESAHAN

«JUDUL BAHASA INDONESIA»

David Christopher

NPM: 2017730015

Bandung, «tanggal» «bulan» «tahun»

Menyetujui,

Pembimbing Utama

Pembimbing Pendamping

«pembimbing utama/1»

«pembimbing pendamping/2»

Ketua Tim Penguji

Anggota Tim Penguji

«penguji 1»

«penguji 2»

Mengetahui,

Ketua Program Studi

Mariskha Tri Adithia, P.D.Eng



## PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

**«JUDUL BAHASA INDONESIA»**

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,  
Tanggal «tanggal» «bulan» «tahun»

Meterai Rp. 6000
---------------------

David Christopher  
NPM: 2017730015





## ABSTRAK

«Tuliskan abstrak anda di sini, dalam bahasa Indonesia»

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

**Kata-kata kunci:** «Tuliskan di sini kata-kata kunci yang anda gunakan, dalam bahasa Indonesia»



## ABSTRACT

«Tuliskan abstrak anda di sini, dalam bahasa Inggris»

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

**Keywords:** «Tuliskan di sini kata-kata kunci yang anda gunakan, dalam bahasa Inggris»



*«kepada siapa anda mempersembahkan skripsi ini...?»*



## KATA PENGANTAR

«Tuliskan kata pengantar dari anda di sini ...»

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Quisque ullamcorper placerat ipsum. Cras nibh. Morbi vel justo vitae lacus tincidunt ultrices. Lorem ipsum dolor sit amet, consectetur adipiscing elit. In hac habitasse platea dictumst. Integer tempus convallis augue. Etiam facilisis. Nunc elementum fermentum wisi. Aenean placerat. Ut imperdiet, enim sed gravida sollicitudin, felis odio placerat quam, ac pulvinar elit purus eget enim. Nunc vitae tortor. Proin tempus nibh sit amet nisl. Vivamus quis tortor vitae risus porta vehicula.

Bandung, «bulan» «tahun»

Penulis





# DAFTAR ISI

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xix
DAFTAR TABEL	xxi
<b>1 PENDAHULUAN</b>	<b>1</b>
1.1 Latar Belakang . . . . .	1
1.2 Rumusan Masalah . . . . .	1
1.3 Tujuan . . . . .	1
1.4 Batasan Masalah . . . . .	2
1.5 Metodologi . . . . .	2
1.6 Sistematika Pembahasan . . . . .	2
<b>2 LANDASAN TEORI</b>	<b>3</b>
2.1 Latar Belakang . . . . .	3
2.2 Jsoup . . . . .	3
2.2.1 Jsoup . . . . .	3
2.2.2 Connection . . . . .	3
2.2.3 Response . . . . .	4
2.2.4 Document . . . . .	5
2.2.5 Elements . . . . .	5
2.2.6 Element . . . . .	5
2.3 <i>Android Design</i> . . . . .	6
2.3.1 <i>Material Design</i> . . . . .	6
2.3.2 Kualitas Aplikasi . . . . .	7
<b>3 ANALISIS</b>	<b>11</b>
3.1 Analisis Sistem Kini . . . . .	11
3.1.1 Analisis Fitur IFStudentPortal . . . . .	11
3.2 Analisis Sistem Usulan . . . . .	13
3.2.1 Pemenuhan Panduan <i>Material Design</i> . . . . .	13
3.2.2 Pemenuhan Panduan <i>Kualitas Aplikasi</i> . . . . .	14
3.2.3 Persiapan . . . . .	15
3.2.4 Analisis Use Case . . . . .	15
3.2.5 Analisis Kelas . . . . .	16
<b>DAFTAR REFERENSI</b>	<b>17</b>
<b>A KODE PROGRAM</b>	<b>19</b>
<b>B HASIL EKSPERIMEN</b>	<b>21</b>



## DAFTAR GAMBAR

B.1 Hasil 1 . . . . .	21
B.2 Hasil 2 . . . . .	21
B.3 Hasil 3 . . . . .	21
B.4 Hasil 4 . . . . .	21



## DAFTAR TABEL



# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang

Student Portal UNPAR adalah sistem informasi berbasis *web* yang digunakan oleh mahasiswa UNPAR [1]. IF Student Portal adalah sistem informasi berbasis *web* khusus mahasiswa Teknik Informatika UNPAR hasil pengembangan lebih lanjut dari Student Portal UNPAR [2]. Student Portal UNPAR adalah sistem informasi berbasis *web* yang digunakan oleh mahasiswa UNPAR [1]. IF Student Portal adalah sistem informasi berbasis *web* khusus mahasiswa Teknik Informatika UNPAR hasil pengembangan lebih lanjut dari Student Portal UNPAR [2].

Skripsi ini dibuat untuk membawa IF Student Portal ke platform *android* sebagai aplikasi *native android* IF Student Portal dengan dibantu *library* jsoup dan mengikuti pedoman *android design*.

Aplikasi akan dibangun dengan bahasa *native android* yaitu Java. Fitur-fitur yang tersedia di situs IF Student Portal akan dibawa ke platform *android* dan dibuat agar terintegrasi dengan fitur yang dimiliki *android*. Aplikasi *android* IF Student Portal akan mengambil data langsung dari Student Portal Unpar dengan dibantu *library* jsoup. Untuk melakukan pengambilan data, jsoup akan langsung berkomunikasi dengan Student Portal UNPAR.

### 1.2 Rumusan Masalah

Rumusan masalah yang akan dibahas di skripsi ini adalah sebagai berikut :

- Fitur apa saja yang akan tersedia di aplikasi *android* IF Student Portal?
- Bagaimana cara mengimplementasikan fungsi aplikasi *android* IF Student Portal dengan *library* jsoup?
- Bagaimana cara mengimplementasikan tampilan aplikasi *android* IF Student Portal dengan mengikuti pedoman *android design*?

### 1.3 Tujuan

Tujuan yang ingin dicapai dari penulisan skripsi ini sebagai berikut :

- Mengetahui fitur apa saja yang akan tersedia di IF Student Portal.
- Mengimplementasikan fungsional aplikasi *android* IF Student Portal menggunakan *library* jsoup..
- Mengimplementasikan tampilan aplikasi *android* IF Student Portal mengikuti pedoman *android design*.

## 1.4 Batasan Masalah

Beberapa batasan yang dibuat terkait dengan pengerjaan skripsi ini adalah sebagai berikut:

1. Aplikasi akan diuji menggunakan format NPM mahasiswa angkatan 2017 ke atas.
2. Aplikasi hanya akan diuji pada perangkat jenis ponsel pintar.

## 1.5 Metodologi

Metodologi yang dilakukan pada skripsi ini adalah sebagai berikut:

1. Melakukan studi mengenai *library* jsoup untuk mengambil data dari Portal Akademik Mahasiswa, *android design*, dan skripsi Herfan Heryandi untuk membangun aplikasi *android* IF Student Portal.
2. Menganalisis Student Portal UNPAR dan IF Student Portal.
3. Merancang model IF Student Portal.
4. Mengimplementasi aplikasi *android* IF Student Portal menggunakan *library* jsoup dengan mengikuti pedoman *android design*.
5. Melakukan eksperimen dan pengujian.
6. Membuat dokumentasi.

## 1.6 Sistematika Pembahasan

Sistematika penulisan setiap bab pada skripsi ini adalah sebagai berikut:

1. Bab Pendahuluan  
Bab 1 berisi latar belakang, rumusan masalah, tujuan, metode penelitian, dan sistematika penulisan yang digunakan untuk menyusun skripsi ini.
2. Bab Dasar Teori  
Bab 2 berisi teori-teori yang digunakan dalam pembuatan skripsi ini. Teori yang digunakan yaitu *library* jsoup, SIA Models, dan *android design*.
3. Bab Analisis  
Bab 3 berisi analisis yang dilakukan pada skripsi ini, meliputi analisis sistem, analisis kebutuhan aplikasi *android* IF Student Portal, analisis pedoman *android design*, analisis Student Portal UNPAR dan situs IF Student Portal untuk fitur IF Student Portal, analisis *use case* meliputi diagram *use case* dan skenario, serta analisis desain kelas.
4. Bab Perancangan  
Bab 4 berisi perancangan aplikasi, meliputi diagram kelas rinci beserta deskripsi kelas dan fungsinya dan perancangan antarmuka aplikasi.
5. Bab Implementasi dan Pengujian  
Bab 5 berisi implementasi dan pengujian aplikasi, meliputi lingkungan implementasi, hasil implementasi, pengujian fungsional, dan pengujian eksperimental.
6. Bab Kesimpulan dan Saran  
Bab 6 berisi kesimpulan dari hasil pembangunan aplikasi beserta saran untuk pengembangan berikutnya.



## BAB 2

### LANDASAN TEORI

#### 2.1 Latar Belakang

Pada bab ini akan dijelaskan dasar teori mengenai jsoup, *android design*, arsitektur *MVP*(*Model-View-Presenter*),

#### 2.2 Jsoup

Jsoup adalah *library* Java yang digunakan untuk mengambil data berupa HTML dari sebuah situs. Data HTML tersebut bisa digunakan untuk memperoleh informasi yang diperlukan dari suatu situs[3]. *Library jsoup* ini bisa digunakan saat sebuah situs tidak menyediakan API untuk memberikan datanya. Kelas-kelas yang digunakan dari *library* ini akan dijelaskan di sub bab berikut.

##### 2.2.1 Jsoup

Kelas ini merupakan kelas utama dalam *library* jsoup. Seluruh *method* dalam kelas ini adalah **static method** sehingga kelas ini tidak perlu dikonstruksi terlebih dahulu sebelum menggunakan *method* dari kelas ini. Salah satu *method* yang dimiliki kelas ini adalah sebagai berikut:

- **public static Connection connect(String url)**  
Berfungsi untuk membuat koneksi baru dengan suatu situs web.  
**Parameter:**
  - **url** : URL situs web dengan protokol HTTP atau HTTPS.**Kembalian:** koneksi dengan situs web.

##### 2.2.2 Connection

Kelas ini merupakan **interface** yang berguna untuk pengambilan data dari situs web. Beberapa *method* yang dimiliki kelas ini adalah sebagai berikut:

- **Connection cookies(Map<String,String> cookies)**  
Berfungsi untuk menambahkan *cookie*.  
**Parameter:**
  - **cookies** : Map dari *cookie*.**Kembalian:** koneksi yang sama tetapi sudah diberi *cookies*.
- **Connection data(String key, String value)**  
Berfungsi untuk menambahkan parameter data yang bisa dikirim melalui metode HTTP GET atau POST.  
**Parameter:**

- **key** : kunci data.
- **value** : nilai data.

**Kembalian:** koneksi yang sama tetapi sudah diberi parameter data.

- **Connection method(Connection.Method method)**

Berfungsi untuk mengatur metode permintaan HTTP, GET atau POST. Metode pengiriman secara *default* adalah GET

**Parameter:**

- **method** : metode pengiriman permintaan HTTP.

**Kembalian:** koneksi yang sama tetapi sudah diatur metode permintaannya.

- **Connection timeout(int millis)**

Berfungsi untuk mengatur batas waktu *request*. Batas waktu nol akan dianggap sebagai batas waktu yang tak terhingga.

**Parameter:**

- **millis** : batas waktu dalam milidetik.

**Kembalian:** koneksi yang sama tetapi sudah diberi batas waktu untuk *request*.

- **Connection validateTLSCertificates(boolean value)**

Berfungsi untuk mengatur pemeriksaan sertifikat TLS untuk permintaan HTTPS. Nilai **true** untuk memeriksa dan nilai **false** untuk tidak memeriksa.

**Parameter:**

- **value** : status pemeriksaan sertifikat TLS.

**Kembalian:** koneksi yang sama tetapi sudah diatur pemeriksaan sertifikatnya.

- **Connection.Response execute()**

Berfungsi untuk mengirim permintaan HTTP.

**Kembalian:** objek **Response**.

### 2.2.3 Response

Kelas ini merepresentasikan permintaan HTTP. Beberapa *method* yang dimiliki kelas ini adalah sebagai berikut:

- **Map<String,String> cookies()**

*Method* ini berfungsi untuk mendapatkan seluruh *cookies*.

**Kembalian:** seluruh *cookies*.

- **Document parse()**

Berfungsi untuk merubah *body* jawaban menjadi dokumen.

**Kembalian:** dokumen yang sudah dirubah.

- **String body()**

Berfungsi untuk mendapatkan *body* jawaban dalam bentuk *string*.

**Kembalian:** *body* jawaban dalam bentuk *string*.

### 2.2.4 Document

Kelas ini merepresentasikan dokumen HTML. Salah satu *method* yang dimiliki kelas ini adalah sebagai berikut:

- **public Elements select(String cssQuery)**

*Method* ini diturunkan dari kelas Element, berfungsi untuk menemukan elemen HTML yang sesuai dengan kueri CSS.

**Parameter:**

- **cssQuery** : kueri CSS berupa CSS Selector.

**Kembalian:** elemen-elemen HTML yang sesuai dengan kueri CSS.

### 2.2.5 Elements

Kelas ini merepresentasikan kumpulan elemen HTML. Beberapa *method* yang dimiliki kelas ini adalah sebagai berikut:

- **public Elements select(String query)**

Berfungsi untuk menemukan elemen-elemen yang sesuai dalam *list* elemen.

**Parameter:**

- **query** : kueri CSS berupa CSS Selector.

**Kembalian:** elemen-elemen yang sudah diseleksi sesuai kueri.

- **public String val()**

Berfungsi untuk mendapatkan nilai dari elemen pertama.

**Kembalian:** nilai elemen.

- **public String text()**

*Method* Berfungsi untuk mendapatkan kombinasi teks dari seluruh elemen yang sesuai.

**Kembalian:** seluruh teks dalam *string*.

### 2.2.6 Element

Kelas ini merepresentasikan sebuah elemen HTML yang berisikan *tag*, atribut, dan anak elemen. Beberapa *method* yang dimiliki kelas ini adalah sebagai berikut:

- **public Element child(int index)**

Berfungsi untuk mendapatkan anak elemen berdasarkan nomor indeks.

**Parameter:**

- **index** : nomor index.

**Kembalian:** anak elemen.

- **public Element children()**

Berfungsi untuk mendapatkan seluruh anak elemen.

**Kembalian:** seluruh anak elemen.

- **public String className()**

Berfungsi untuk mendapatkan nama kelas elemen.

**Kembalian:** nama kelas elemen.

- **public String text()**

Berfungsi untuk mendapatkan teks dari elemen.

**Kembalian:** teks dalam *string*.

## 2.3 *Android Design*

Dalam mengembangkan aplikasi *android*, ada pedoman desain yang diberikan. Desain tampilan aplikasi *android* mengacu pada *Android Material design*. Pedoman kualitas aplikasi juga diberikan untuk aspek kompatibilitas, keamanan, performa, dll. Kedua pedoman ini diberikan agar aplikasi yang dihasilkan bisa memiliki tampilan dan perilaku yang konsisten dengan platform *android*.

### 2.3.1 *Material Design*

*Material Design* terdiri dari panduan, komponen, dan alat-alat untuk mendukung pembuatan tampilan antarmuka yang baik. *Material Design* bertujuan mempermudah kolaborasi antara desainer dan pengembang aplikasi untuk membuat produk yang cantik dengan cepat[4]. Ada 2 panduan utama yang harus diikuti dari *Material Design* yaitu panduan aksesibilitas dan panduan platform.

#### Panduan Aksesibilitas

Aksesibilitas dapat diartikan sebagai tingkat kemudahan saat pengguna mempelajari dan menggunakan tampilan antarmuka aplikasi. Meningkatkan aksesibilitas akan meningkatkan *usability* atau tingkat kebergunaan aplikasi kepada berbagai kalangan pengguna [4]. Hal penting yang harus diperhatikan agar aplikasi memiliki aksesibilitas yang baik :

- Hierarki : Elemen-elemen harus terlihat dengan jelas baik dari ukuran, kontras, dan informasi yang terkandung di dalamnya. Elemen-elemen juga harus terurut berdasarkan kepentingan dan elemen dengan fungsi tertentu diletakkan di tempat yang mudah dijangkau.
- Warna dan Kontras : Kontras yang cukup tinggi akan membuat elemen terlihat jelas, namun jika terlalu tinggi juga akan kurang nyaman dilihat terlalu lama, jika terlalu rendah maka elemen-elemen akan sulit dibedakan satu sama lain (misal warna teks dengan latar belakangnya). Jika warna digunakan sebagai suatu indikator, maka perlu ada tambahan keterangan lain agar pengguna yang menderita buta warna tetap bisa menerima informasi dengan baik.
- Tata Letak dan Tipografi : Menggunakan *layout* yang fleksibel dan responsif akan membantu isi konten menyesuaikan dengan skala layar agar konten tidak ada yang terpotong dengan tidak sengaja. Pastikan menggunakan format sp untuk ukuran *font* agar ukuran teks ikut terskala dengan baik jika pengguna merubah ukuran *font* dari pengaturan sistem perangkat. Pastikan juga ada ruang yang cukup untuk menampung ukuran teks yang diperbesar.
- Gambar : Gunakan gambar untuk memperjelas informasi yang disajikan. Gambar logo boleh tidak mematuhi panduan warna, kontras, dan ukuran teks, namun sebaiknya tetap memiliki fungsi (misalnya logo sebagai tombol ke halaman utama).

#### Panduan Platform

Panduan platform membantu menentukan bagaimana ketentuan yang akan digunakan untuk setiap platform. Platform yang dimaksud pada bagian ini adalah bagian-bagian dari komponen aplikasi android. Platform yang dimaksud adalah :

- Notifikasi : notifikasi digunakan untuk memberi tahu komunikasi dari pengguna lain dan mengingatkan hal yang perlu dilakukan. Notifikasi bisa ditampilkan di halaman terkunci, di *status bar*, dengan kedipan lampu LED, dan dengan suara dan getaran. Sebaiknya notifikasi tidak digunakan untuk promosi, meminta rating aplikasi, dan memberi tahu proses yang tidak berhubungan dengan pengguna. Pada saat layar terkunci sebaiknya notifikasi tidak menampilkan informasi yang sensitif. Notifikasi harus punya bagian *header*, bagian konten, dan bagian aksi.

- Izin Akses (*permission*) : Secara normal, aplikasi punya izin akses ke beberapa hal tanpa perlu memintanya kepada pengguna. Namun beberapa hal berikut membutuhkan konfirmasi dari pengguna agar aplikasi memiliki izin akses :
  - Kalender : Mengakses dan mengatur kalender.
  - Kamera : Mengambil foto dan merekam video.
  - Kontak : Membaca dan mengatur kontak.
  - Lokasi : Membaca lokasi perangkat saat ini.
  - Mikrofon : Merekam suara.
  - Telepon : Membuat dan mengatur panggilan telepon.
  - Sensor tubuh : Membaca detak jantung dan hal sejenis lainnya.
  - Media penyimpanan : Mengakses dan menggunakan media penyimpanan.
  - SMS : Mengirim dan membaca SMS (pesan singkat).

### 2.3.2 Kualitas Aplikasi

Pengguna *android* tentu menginginkan aplikasi berkualitas. Kualitas aplikasi akan menentukan keberhasilan aplikasi dalam hal instalasi, ulasan, loyalitas dan keterlibatan pengguna untuk jangka panjang [5]. Dokumentasi *android developers* memberikan kriteria untuk mengukur kualitas aplikasi sebagai berikut :

- Desain visual dan interaksi pengguna
- Fungsionalitas
- Kompatibilitas, performa, dan stabilitas
- Keamanan
- *Google Play*

Untuk keterangan lebih lanjut dari setiap poin diatas akan dijelaskan di bagian berikutnya.

#### Desain Visual dan Interaksi Pengguna

Kriteria ini bertujuan memastikan aplikasi akan memiliki desain visual dan pola interaksi standar agar pengalaman pengguna konsisten dan intuitif[6]. Kriteria ini dapat dijabarkan sebagai berikut :

- Aplikasi tidak boleh merubah definisi ikon sistem dengan fungsinya, jika aplikasi menyediakan ikon yang disesuaikan, maka tampilannya harus mirip ikon standar dan memicu perilaku sesuai fungsi standarnya.
- Aplikasi hanya menggunakan notifikasi untuk memberitahu perubahan yang terjadi dengan konteks yang berkaitan dengan pengguna pribadi dan untuk memberi tahu informasi/kontrol terhadap kejadian yang sedang berlangsung.

#### Fungsionalitas

Kriteria ini bertujuan memastikan aplikasi memberikan perilaku fungsional yang diharapkan, dengan tingkat izin yang sesuai [6]. Kriteria ini dapat dijabarkan sebagai berikut :

- Aplikasi hanya meminta izin untuk mendukung fungsionalitas aplikasi tersebut. Aplikasi tidak boleh meminta izin untuk mengakses data sensitif atau menggunakan layanan yang bisa membebani pengguna kecuali jika fitur inti aplikasi memerlukan izin tersebut.

- Aplikasi harus berfungsi normal jika dipasang di kartu *SD*.
- Audio tidak boleh diputar di layar utama, saat layar mati, dibalik layar, atau saat layar dikunci kecuali memutar audio adalah fitur utama.
- Jika memungkinkan aplikasi mendukung orientasi *landscape* dan *portrait*, dan menggunakan seluruh layar untuk kedua orientasi.
- Aplikasi tidak boleh membiarkan layanan tetap aktif saat di latar belakang layar, kecuali jika diperlukan fitur utama.
- Aplikasi mempertahankan status pengguna atau aplikasi saat meninggalkan latar depan dan mencegah kehilangan data tanpa sengaja akibat navigasi mundur dan perubahan status lainnya.

### Kompatibilitas, Performa, dan Stabilitas

Kriteria ini bertujuan memastikan aplikasi memberikan kompatibilitas, performa, stabilitas, dan daya respons yang diharapkan oleh pengguna [6]. Kriteria ini dapat dijabarkan sebagai berikut :

- Aplikasi diharapkan tidak macet, berfungsi tidak normal, menutup sendiri di perangkat yang menjalankan.
- Aplikasi dimuat dengan cepat atau memberikan indikator kepada pengguna tentang kapan aplikasi selesai dimuat.
- Aplikasi dibuat dengan *SDK* terbaru dan berjalan di *android* versi terbaru tanpa kendala.
- Aplikasi memutar video dan audio dengan lancar, tidak tersendat, suara dan gambar tidak pecah, atau cacat lainnya.
- Aplikasi menampilkan elemen antarmuka tanpa pikselasi, distorsi, dan tidak bergerigi pada tepian.
- Aplikasi menyediakan grafik berkualitas tinggi untuk semua ukuran layar yang ditargetkan.
- Aplikasi mendukung fitur pengelolaan daya baterai (*Android 6.0+*).

### Keamanan

Kriteria ini bertujuan memastikan aplikasi menangani dan mengamankan data pengguna dan informasi pribadi dengan benar[6]. Kriteria ini dapat dijabarkan sebagai berikut :

- Aplikasi harus menyimpan data pribadi di penyimpanan internal aplikasi dan tidak boleh mencatat data pribadi di log.
- Aplikasi harus memverifikasi data eksternal sebelum digunakan.
- Semua komponen aplikasi yang berbagi konten dengan aplikasi lain menetapkan (dan mem-berlakukan) izin yang sesuai, termasuk aktivitas, layanan, penerima siaran, dan khususnya penyedia konten.
- Aplikasi hanya boleh mengekspor komponen aplikasi yang membagikan data dengan aplikasi lain, atau komponen yang harus dipanggil oleh aplikasi lain.
- Aplikasi harus menyatakan konfigurasi keamanan jaringan dan semua lalu lintas jaringan dilakukan melalui *SSL*.

- Jika aplikasi menggunakan layanan *Google Play*, inisialisasi keamanan dilakukan saat aplikasi dimulai.
- Aplikasi harus menggunakan dependensi, *library* dan *SDK* terbaru.
- Aplikasi tidak boleh menjalankan kode dari luar aplikasi secara dinamis.
- Aplikasi harus menggunakan algoritma kriptografi kuat yang disediakan oleh platform.

### ***Google Play***

Kriteria ini bertujuan memastikan aplikasi yang dibuat sudah layak, memenuhi standar dan syarat untuk dipublikasikan di layanan *Google Play*[6]. Kriteria ini dapat dijabarkan sebagai berikut :

- Aplikasi mematuhi Kebijakan Materi Pengembang *Google Play* (tidak menawarkan materi tidak pantas, tidak menggunakan hak kekayaan intelektual atau merk orang lain, dll).
- Aplikasi sudah memenuhi kriteria yang sudah diuraikan sebelum bagian ini.
- Pengembang aplikasi harus mengatasi *bug* yang disampaikan di halaman ulasan di layanan *Google Play* jika *bug* tersebut ditemukan di banyak perangkat dan berulang kali atau ditemukan di perangkat terbaru atau perangkat paling populer.





## BAB 3

### ANALISIS

Pada bab ini akan dijelaskan analisis sistem yang sudah berjalan dan sistem yang akan dibangun. Analisis yang akan dibahas meliputi analisis *use case*, analisis kelas, analisis kebutuhan fungsional dan non fungsional.

#### 3.1 Analisis Sistem Kini

IFStudentPortal adalah situs yang diperuntukan bagi mahasiswa IF UNPAR[?]. IFStudentPortal dibuat dengan tujuan menjadi pengembangan lebih lanjut dari Student Portal UNPAR pada masanya. IFStudentPortal adalah hasil skripsi Herfan Heryandi [2] dan kontributor lainnya. Saat ini IFStudentPortal telah mendukung kurikulum 2018 berkat kontribusi skripsi Andrianto Sugiarto [7]<sup>1</sup>. Untuk mengakses Portal Akademik Mahasiswa, mahasiswa harus *login* menggunakan akun email *student*.

SIA Models merupakan *library* Java yang merepresentasikan Sistem Informasi Akademik Teknik Informatika UNPAR [8]. Saat ini SIAModels mendukung kurikulum 2018.

##### 3.1.1 Analisis Fitur IFStudentPortal

IFStudentPortal pada awalnya dibuat untuk melengkapi informasi yang belum ada di student portal UNPAR pada tahun 2015. Fitur-fitur yang diimplementasikan di IFStudentPortal adalah sebagai berikut :

1. *Login* : Untuk dapat menggunakan situs IFStudentPortal, mahasiswa Teknik Informatika UNPAR harus *login* menggunakan akun dan kata sandi yang sama dengan yang digunakan untuk *login* ke Student Portal UNPAR.
  - Nama: *Login*
  - Aktor: Mahasiswa
  - Deskripsi: *Login* ke IFStudentPortal dengan membuka koneksi ke Student Portal UNPAR
  - Kondisi awal: Mahasiswa adalah mahasiswa program studi Teknik Informatika UNPAR
  - Kondisi akhir: Halaman utama IFStudentPortal ditampilkan
  - Skenario utama:

---

<sup>1</sup>Mahasiswa Teknik Informatika dapat mengakses IFStudentPortal melalui URL <https://ifstudentportal.herokuapp.com/>

No	Aksi Aktor	Reaksi Sistem
1	Pengguna mengakses situs IFStudentPortal	Sistem menampilkan halaman <i>login</i>
2	Pengguna mengisi akun dan kata sandi lalu menekan tombol <i>login</i>	Sistem membuka koneksi ke Student Portal UNPAR untuk melakukan pengecekan identitas <i>login</i>
3		Jika login berhasil maka sistem akan menampilkan halaman utama IFStudentPortal

2. Prasyarat Mata Kuliah Mahasiswa bisa melihat mata kuliah apa saja yang dibuka di semester berjalan dan memeriksa prasyarat untuk mata kuliah tersebut.

- Nama: Memeriksa prasyarat mata kuliah
- Aktor: Mahasiswa
- Deskripsi: Memeriksa prasyarat mata kuliah yang dibuka pada semester terkini
- Kondisi awal: Mahasiswa telah *login*
- Kondisi akhir: Halaman prasyarat mata kuliah ditampilkan dan berisi mata kuliah yang dibuka pada semester terkini beserta status prasyaratnya
- Skenario utama:

No	Aksi Aktor	Reaksi Sistem
1	Mahasiswa memilih menu prasyarat mata kuliah.	Sistem mendapatkan data mahasiswa kemudian menampilkan halaman prasyarat mata kuliah

- Eksepsi: Mahasiswa sedang menempuh semester 1

3. Jadwal kuliah yang tersusun Pada saat IFStudentPortal pertama kali dibangun, Student Portal UNPAR saat itu belum memiliki jadwal dengan tampilan grafik yang terurut berdasarkan hari dan jamnya. Fitur ini dibuat untuk mempermudah melihat susunan jadwal.

- Nama: Melihat jadwal kuliah
- Aktor: Mahasiswa
- Deskripsi: Melihat jadwal kuliah yang sudah tersusun dan terurut berdasarkan hari dan jam
- Kondisi awal: Mahasiswa telah *login*
- Kondisi akhir: Halaman jadwal ditampilkan dan berisi jadwal kuliah yang sudah tersusun dan terurut berdasarkan hari dan jam
- Eksepsi: Jadwal kuliah belum keluar
- Skenario utama:

No	Aksi Aktor	Reaksi Sistem
1	Mahasiswa memilih menu jadwal.	Sistem menyusun dan mengurutkan jadwal mahasiswa berdasarkan hari kemudian menampilkan halaman jadwal

4. Melihat ringkasan data akademik Pada saat IFStudentPortal pertama kali dibangun, Student Portal UNPAR saat itu belum memberikan rincian data akademik yang detil. Fitur ini dibuat untuk melengkapi informasi data akademik tersebut.

### 5. Melihat Ringkasan Data Akademik

- Nama: Melihat ringkasan data akademik
- Aktor: Mahasiswa
- Deskripsi: melihat data mengenai mata kuliah apa saja yang sudah lulus beserta jenis mata kuliahnya(wajib atau pilihan), sisa SKS untuk mencapai kelulusan, dan mata kuliah wajib yang belum ditempuh. Mahasiswa juga dapat melihat IPS dan IPK yang lebih *ter-update*
- Kondisi awal: Mahasiswa telah *login*
- Kondisi akhir: Halaman yang berisi jadwal ringkasan data akademik ditampilkan
- Skenario utama:

No	Aksi Aktor	Reaksi Sistem
1	Mahasiswa memilih menu ringkasan data akademik.	Sistem meringkas data akademik mahasiswa kemudian menampilkan halaman ringkasan data akademik

## 3.2 Analisis Sistem Usulan

IFStudentPortal akan dibangun untuk platform *android*. Dokumentasi *android* menyediakan panduan dalam membangun aplikasi agar aplikasi yang dibangun memiliki tampilan dan perilaku yang konsisten dengan platform *android*. Panduan tersebut sudah dijelaskan di sub-bab 2.3. Pada sub-bab ini akan dijelaskan cara mencapai standar yang diberikan oleh panduan tersebut.

### 3.2.1 Pemenuhan Panduan *Material Design*

Panduan aksesibilitas bertujuan memberikan arahan agar aplikasi yang dibangun mudah untuk dipelajari dan digunakan. Meningkatkan aksesibilitas akan meningkatkan kebergunaan aplikasi kepada berbagai kalangan pengguna. Berikut akan dipaparkan langkah-langkah yang dilakukan untuk meningkatkan aksesibilitas aplikasi *android* IFStudentPortal :

#### 1. Hierarki :

- Warna latar belakang akan memiliki kontras yang cukup dengan warna teks.
- Ukuran teks akan menyesuaikan dengan ukuran layar sehingga tidak ada teks yang terlalu kecil sehingga sulit dibaca
- Pengisian *form* akan mengalir dari atas ke bawah, dan tombol untuk *submit form* tersebut akan berada di bawah form.
- Penyampaian informasi akan mengalir terurut dari atas ke bawah
- Tombol diletakan di tempat yang mudah dijangkau dengan memberikan area tekan yang besar atau meletakkannya di dekat ujung layar.

#### 2. Warna dan Kontras :

- Warna latar belakang akan memiliki kontras yang cukup dengan warna teks.
- Peringatan atau pesan *error* akan berwarna merah disertai dengan keterangan .

#### 3. Tata Letak dan Tipografi :

- Menggunakan *layout* yang fleksibel dan responsif.
- Menggunakan format sp untuk satuan ukuran *font* agar ukuran teks menyesuaikan dengan ukuran layar.

## 4. Gambar :

- Menggunakan foto profil mahasiswa untuk melengkapi informasi.

## 5. Notifikasi :

- \*belum ditentukan\*

## 6. Izin akses :

- Menampilkan kotak dialog dan tombol untuk meminta izin pengguna sebelum mengakses dan mengatur kalender.
- Hal lain di bagian izin akses 2.3.1 tidak diimplementasikan/digunakan.

### 3.2.2 Pemenuhan Panduan *Kualitas Aplikasi*

Aplikasi yang berkualitas tentu saja diinginkan oleh pengguna *android*. Kualitas akan mempengaruhi keberhasilan aplikasi dalam instalasi, ulasan, loyalitas, dan keterlibatan pengguna dalam jangka panjang. Berikut akan dipaparkan langkah-langkah yang dilakukan untuk memastikan aplikasi *android* IFStudentPortal memiliki kualitas yang baik :

## 1. Desain visual dan interaksi pengguna :

- Aplikasi tidak akan menggunakan ikon yang ambigu sehingga pengguna tidak kebingungan dalam mengartikan fungsi tombol dengan ikon tersebut.
- Notifikasi akan digunakan untuk memberi tahu pengguna tentang ... \*belum ditentukan\*

## 2. Fungsionalitas :

- Aplikasi akan meminta izin akses dari pengguna sebelum menggunakan kalender.
- Aplikasi tidak menggunakan audio sebagai fitur utama.
- Aplikasi tidak menggunakan orientasi *landscape*.
- Aplikasi akan memberhentikan layanan saat pengguna menutup aplikasi sehingga aplikasi tidak aktif saat di belakang layar.
- Aplikasi akan mempertahankan *state login* selama ada aktifitas atau sampai batas waktu yang ditentukan.

## 3. Kompaatibilitas, Performa, dan Stabilitas :

- Aplikasi akan dibangun dengan *library android* versi terbaru yang sudah stabil untuk meminimalisir kemungkinan aplikasi macet. Aplikasi akan diuji sebelum dirilis untuk memperbaiki *bug* yang ditemukan.
- Aplikasi tidak menggunakan algoritma yang kompleks dan melakukan proses yang berat sehingga tidak perlu waktu lama untuk memuat aplikasi.
- Aplikasi dibangun dengan SDK terbaru dan diuji di perangkat terbaru dan di perangkat yang populer.
- Grafik yang ditampilkan adalah gambar dengan resolusi tinggi

## 4. Keamanan :

- Aplikasi tidak menyimpan data pengguna baik di internal maupun di log.
- Aplikasi tidak berbagi konten dengan aplikasi lain.
- Aplikasi tidak mengekspos komponen aplikasi dengan aplikasi lain.

- Aplikasi membuka koneksi dengan Student Portal UNPAR yang sudah diamankan dengan SSL.
- Aplikasi dibangun dengan *dependency*, *library*, dan *SDK* terbaru.
- Aplikasi tidak mengeksekusi kode external secara dinamis.

#### 5. Google Play :

- Aplikasi tidak mengandung materi yang tidak pantas dan tidak menggunakan hak kekayaan intelektual atau merk orang lain.
- Aplikasi akan dibangun dengan mengikuti panduan yang sudah diuraikan sebelum bagian ini.
- Aplikasi tidak mengekspos komponen aplikasi dengan aplikasi lain.
- Pengembang akan memperbaiki *bug* yang ditemukan.

### 3.2.3 Persiapan

Sebelum dan selama proses pengembangan aplikasi *android* IFStudentPortal berjalan, ada beberapa pekerjaan yang dilakukan sebagai berikut :

- Perawatan situs IFStudentPortal  
IFStudentPortal beberapa kali mengalami gangguan sehingga tidak bisa digunakan sebagaimana mestinya, sehingga dilakukan perbaikan-perbaikan yang diperlukan. Perubahan yang dilakukan adalah sebagai berikut :
  - Memperbaiki *bug* alamat foto profil dari Student Portal UNPAR<sup>2</sup>.
  - Memperbaiki *bug* mengambil kode semester dari Student Portal UNPAR dengan memindahkan halaman sumber kode semester dari halaman frs ke halaman nilai<sup>3</sup>.
  - Merubah penggunaan SIA Models dari *submodule* ke *Maven*<sup>4</sup>.
- Perawatan SIA Models  
SIA Models versi 3.1.0 belum cocok untuk digunakan di platform *android* sehingga perlu sedikit modifikasi agar bisa digunakan untuk pembangunan *android* IFStudentPortal. Perubahan yang dilakukan adalah sebagai berikut :
  - Merubah nilai kembalian dari *method* *getPhotoImage* menjadi *byte*<sup>5</sup>.

### 3.2.4 Analisis Use Case

Diagram *use case* hanya akan memiliki 1 aktor yaitu mahasiswa Teknik Informatika UNPAR. Diagram *use case* dapat dilihat di gambar ??.

Berdasarkan hasil analisis kebutuhan pada subbab ??, dari x fitur yang akan dibuat, terdapat y buah *use case* yaitu :

#### 1. Melihat Ringkasan Data Akademik

- Nama: Melihat ringkasan data akademik
- Aktor: Mahasiswa

<sup>2</sup>Kode dapat dilihat di <https://github.com/ftisunpar/IFStudentPortal/commit/2534764>

<sup>3</sup>Kode dapat dilihat di <https://github.com/ftisunpar/IFStudentPortal/commit/4775b17>

<sup>4</sup>Kode dapat dilihat di <https://github.com/ftisunpar/IFStudentPortal/commit/d88bba>

<sup>5</sup>Kode dapat dilihat di <https://www.github.com/pascalalfadian/siamodels/commit/0df064f>.

- Deskripsi: melihat data mengenai mata kuliah apa saja yang sudah lulus beserta jenis mata kuliahnya(wajib atau pilihan), sisa SKS untuk mencapai kelulusan, dan mata kuliah wajib yang belum ditempuh.
- Kondisi awal: Mahasiswa telah *login*
- Kondisi akhir: Halaman ditampilkan dan berisi ringkasan data akademik
- Skenario utama:

No	Aksi Aktor	Reaksi Sistem
1	Mahasiswa memilih menu ringkasan data akademik.	Sistem meringkas data akademik mahasiswa kemudian menampilkan halaman ringkasan data akademik

### 3.2.5 Analisis Kelas

## DAFTAR REFERENSI

- [1] Tim Pengembang Student Portal (2018) Portal Akademik Mahasiswa. [https://studentportal.unpar.ac.id/assets/BUKU\\_PANDUAN\\_PENGGUNAAN\\_FRS\\_GABUNGAN.pdf](https://studentportal.unpar.ac.id/assets/BUKU_PANDUAN_PENGGUNAAN_FRS_GABUNGAN.pdf). [Online; diakses 1-Oktober-2020].
- [2] Heryandi, H. (2015) Informatika student portal: Pengembangan portal akademik mahasiswa untuk mahasiswa teknik informatika unpar. Skripsi. Universitas Katolik Parahyangan, Indonesia.
- [3] Hedley, J. (2009-2015) jsoup: Java HTML Parser. <http://jsoup.org>. [Online; diakses 1-Oktober-2020].
- [4] Banes, C. (2014) Material Design. <https://github.com/material-components/material-components-android>. [Online; diakses 1-Oktober-2020].
- [5] Android Developers (2014) Android Design. <https://developer.android.com/design>. [Online; diakses 1-Oktober-2020].
- [6] Android Developers (2014) Android Documentation. <https://developer.android.com>. [Online; diakses 1-Oktober-2020].
- [7] Sugiarto, A. (2018) Penyesuaian siamodels dan if student portal ke kurikulum 2018. Skripsi. Universitas Katolik Parahyangan, Indonesia.
- [8] Nugroho, P. A. (2015) SIA Models. <https://github.com/pascalalfadian/SIAModels>. [Online; diakses 1-Oktober-2020].





# LAMPIRAN A

## KODE PROGRAM

Listing A.1: MyCode.c

```
1 // This does not make algorithmic sense,
2 // but it shows off significant programming characters.
3
4 #include<stdio.h>
5
6 void myFunction( int input, float* output ) {
7     switch ( array[i] ) {
8         case 1: // This is silly code
9             if ( a >= 0 || b <= 3 && c != x )
10                 *output += 0.005 + 20050;
11             char = 'g';
12             b = 2^n + ~right_size - leftSize * MAX_SIZE;
13             c = (--aaa + &daa) / (bbb++ - ccc % 2 );
14             strcpy(a,"hello_$@?");
15         }
16         count = ~mask | 0x00FF00AA;
17     }
18 }
19
20 // Fonts for Displaying Program Code in LATEX
21 // Adrian P. Robson, nepsweb.co.uk
22 // 8 October 2012
23 // http://nepsweb.co.uk/docs/progfonts.pdf
```

Listing A.2: MyCode.java

```
1 import java.util.ArrayList;
2 import java.util.Collections;
3 import java.util.HashSet;
4
5 //class for set of vertices close to furthest edge
6 public class MyFurSet {
7     protected int id; //id of the set
8     protected MyEdge FurthestEdge; //the furthest edge
9     protected HashSet<MyVertex> set; //set of vertices close to furthest edge
10    protected ArrayList<ArrayList<Integer>> ordered; //list of all vertices in the set for each trajectory
11    protected ArrayList<Integer> closeID; //store the ID of all vertices
12    protected ArrayList<Double> closeDist; //store the distance of all vertices
13    protected int totaltrj; //total trajectories in the set
14
15    /*
16     * Constructor
17     * @param id : id of the set
18     * @param totaltrj : total number of trajectories in the set
19     * @param FurthestEdge : the furthest edge
20     */
21    public MyFurSet(int id,int totaltrj,MyEdge FurthestEdge) {
22        this.id = id;
23        this.totaltrj = totaltrj;
24        this.FurthestEdge = FurthestEdge;
25        set = new HashSet<MyVertex>();
26        ordered = new ArrayList<ArrayList<Integer>>();
27        for (int i=0;i<totaltrj;i++) ordered.add(new ArrayList<Integer>());
28        closeID = new ArrayList<Integer>(totaltrj);
29        closeDist = new ArrayList<Double>(totaltrj);
30        for (int i = 0;i <totaltrj;i++) {
31            closeID.add(-1);
32            closeDist.add(Double.MAX_VALUE);
33        }
34    }
35
36 }
```



## LAMPIRAN B

### HASIL EKSPERIMEN

Hasil eksperimen berikut dibuat dengan menggunakan TIKZPICTURE (bukan hasil excel yg diubah ke file bitmap). Sangat berguna jika ingin menampilkan tabel (yang kuantitasnya sangat banyak) yang datanya dihasilkan dari program komputer.



Gambar B.1: Hasil 1



Gambar B.2: Hasil 2



Gambar B.3: Hasil 3



Gambar B.4: Hasil 4