**LAPORAN TUGAS AKHIR**

**TEKNOLOGI BERGERAK**

**PENGEMBANGAN APLIKASI START-UP TEACH ME**

**Disusun oleh :**

**Sarah Shahab H1101171020**

**Dinda Almira Hidayati H1101171025**

**Uray Fasha August Putra H1101171026**

**Nagara Janatama H1101171042**

**Pembimbing :**

**Helen Sasty Pratiwi, S.T., M.Eng.**



**PRODI SISTEM INFORMASI**

**JURUSAN SISTEM INFORMASI**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS TANJUNGPURA**

**PONTIANAK**

# BAB 1

# PENDAHULUAN

## Latar Belakang

Pada zaman modern seperti sekarang, perkembangan teknologi sudah demikian hebatnya. Seperti tak bisa diredam atau ditahan, teknologi mempengaruhi hampir seluruh sendi-sendi kehidupan manusia. Masyarakat cenderung menyelesaikan permasalahan mereka dengan teknologi. Akan tetapi, terkadang masyarakat mengalami kesulitan karena tidak adanya teknologi yang sesuai dengan permasalahan mereka. Contohnya di bidang pendidikan, kurangnya informasi mengenai penyediaan jasa guru privat membuat masyarakat khususnya orang tua, kesulitan mendapatkan guru privat yang sesuai dengan keinginan mereka.

Melihat dari masalah tersebut, PT. Cendikia membangun sebuah platform yaitu Teach Me yang memiliki tujuan untuk menyediakan tenaga pengajar yang dibutuhkan oleh orang tua untuk memberikan bimbingan belajar langsung ke rumah untuk anak-anak mereka.

## Tujuan

1. Memberikan kemudahan bagi guru privat dalam mencari lowongan pekerjaan guru les privat
2. Memberikan kemudahan bagi murid dan orang tua dalam mencari guru privat yang sesuai dengan kriteria
3. Memberikan kemudahan dalam proses mencari informasi tentang profil guru privat dan murid
4. Menghubungkan antara guru privat dan murid dalam satu aplikasi

## Batasan

Batasan-batasan dalam pengembangan perangkat lunak Teach Me adalah:

1. Kebijaksanaan Umum

* Berpedoman pada tujuan dari pengembangan perangkat lunak Teach ME. Pengguna yang mengakses sistempun harus melakukan proses authorisasi.
* Dalam aplikasi ini terdapat beberapa batasan yang harus dipatuhi, misalnya:
* Aplikasi harus terhubung dengan internet.
* Front Office dapat digunakan untuk kelola data pengguna, cari guru, dan konseling online.
* Aplikasi hanya dapat digunakan pada platform android.
* Aplikasi hanya bisa dibuka dan dipakai oleh akun yang terdaftar.

1. Keterbatasan Perangkat Keras

Dapat diketahui setelah perangkat lunak ini berjalan. (sesuai dengan kebutuhan)

1. Keterbatasan Golongan Pengguna

Untuk awal peluncuran aplikasi, target pengguna kami mula-mula adalah masyarakat di Kota Pontianak. Pengguna aplikasi Teach Me terbagi menjadi 3 golongan, yaitu Murid, Guru, dan Admin.

# BAB 2

# STUDI PUSTAKA

## Sistem Operasi Android

Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat bergerak layar sentuh seperti *smartphone* dan computer tablet. Android awalnya dikembangkan oleh Android, Inc., dengan dukungan finansial dari *Google*, yang kemudian membelinya pada tahun 2005.

Sistem operasi ini dirilis secara resmi oada tahun 2007, bersamaan dengan didirikannya Open Handset Alliance, konsorsium dari perusahaan-perusahaan perangkat keras, perangkat lunak, dan telekomunikasi yang bertujuan untuk memajukan standar terbuka perangkat seluler.

Untuk masalah keamanan, sistem operasi Android dirancang untuk berjalan di sandbox ( sebuah area pada sistem yang terisolir ) yang memungkinkan pengguna tidak memiliki akses pada sistem pada saat memasang sebuah aplikasi, kecuali ijin akses secara eksplisit diberikan. Beberapa virus atau malware kemungkinan dapat menyerang perangkat seluler yang menggunakan sistem operasi Android, hal ini biasanya disebabkan oleh pemasangan aplikasi dari sumber tidak terpercaya ( selain Google Play Store ). Akan tetapi apabila pengguna memasang aplikasi melalui Google Play Store, kemungkinan perangkat pengguna terinfeksi virus atau malware sangat minim.

## Android Studio

Android Studio adalah Lingkungan Pengembangan Terpadu (Integrated Development Enviroment / IDE) resmi untuk pengembangan aplikasi Android yang didasarkan pada IntelliJ IDEA. Selain sebagai editor kode dan fitur developer IntelliJ yang andal, Android Studio menawarkan banyak fitur yang meningkatkan produktivitas penggunanya dalam membuat aplikasi Android, seperti:

* Sistem build berbasis Gradle yang fleksibel
* Emulator yang cepat dan kaya fitur
* Dapat digunakan untuk mengembangkan semua aplikasi perangkat Android
* Menerapkan perubahan untuk melakukan puash pada perubahan kode dan resource ke aplikasi yang sedang berjalan tanpa memulai ulang aplikasi.
* Template kode dan integrase GitHub untuk membantu pengguna membuat fitur aplikasi umum dan mengimpor kode sampel.
* Framework dan fitur pengujian yang lengkap.
* Fitur lint untuk merekam performa, kegunaan, kompatibilitas versi, dan masalah lainnya.
* Dukungan C++ dan NDK.

## Android Java

Aplikasi Android saat ini dikembangkan dengan bahasa Java. Sampai saat ini, hanya itu satu-satunya pilihan untuk aplikasi asli. Java adalah sebuah bahasa pemrograman yang sangat popular yang dikembangkan oleh Sun Microsystems (saat ini dimiliki oleh Oracle). Dikembangkan lama setelah C dan C++, Java menggabungkan banyak fitur-fitur canggih dari bahasa-bahasa canggih tersebut, sambil mengatasi beberapa kelemahan mereka. Walaupun demikian, tingkat kecanggihan bahasa pemrograman bergantung pada library mereka. Library ini ada untuk membantu para developer untuk membuat aplikasi. Beberapa fitur inti Java adalah:

* Mudah dipelajari dan dimengerti
* Didesain untuk tidak bergantung kepada platform dan aman, menggunakan  
  mesin virtual
* Bersifat object-oriented (fokus kepada objek program ketimbang logic)

Android sangat bergantung kepada sifat-sifat dasar dari Java tersebut. Android SDK mengandung banyak library Java standar (library struktur data, library matematika, library grafik, library networking, dll) dan juga library special Android yang dapat membantu pengguna mengembangkan aplikasi Android yang keren.

## Unified Modeling Languange (UML)

UML adalah seperangkat aturan dan notasi untuk spesifikasi sistem perangakt lunak, dikelola dan dibuat oleh Object Management Group. Notasi ini menyediakan satu set elemen grafis untuk pemodelan sistem. Uml mendefinisikan diagram-diagram sebagai berikut:

1. Use Case Diagram
2. Class Diagram
3. Statechart Diagram
4. Activity Diagram
5. Sequence Diagram
6. Collaboration Diagram
7. Component Diagram
8. Deployment Diagram

Akan tetapi, untuk perancangan kali ini kami hanya akan menggunakan Use Case Diagram beserta spesifikasinya dan Activity Diagram.

1. Use Case Diagram

Use Case Diagram adalah gambaran graphical dari beberapa atau semua actor, use-case, dan interaksi diantara komponen-komponen tersebut yang memperkenalkan suatu sistem yang akan dibangun. Use case diagram menjelaskan manfaat suatu sistem jika dilihat menurut pandangan orang yang berada di luar sistem.

Diagram ini menunjukkan fungsionalitas suatu sistem atau kelas dan bagaimana sistem tersebut berinteraksi dengan dunia luar. Use case diagram dapat sigunakan selama proses analisis unutk menangkap requirement system dan untuk memahami bagaimana sistem harusnya berkerja.

Elemen-elemen pada Use Case Diagram:

* Actor

Mempresentasikan seseorang atau sesuatu yang berinteraksi dengan sistem. Actor hanya berinteraksi dengan use case tetapi tidak memiliki control atas use case.

* Use Case

Adalah gambaran fungsionalitas dari suatu sistem, sehingga customer atau pengguna sistem paham da mengerti mengenai kegunaan sistem yang akan dibangun.

* Association

Menghubugnkan link antar elemen.

* <<include>>

Yaitu tindakan yang harus dipenuhi agar sebuah event dapat terjadi, dimana pada konsisi ini sebah use case adalah bagian dari use case lainnya.

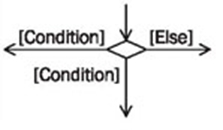
Relasi dalam Use Case:

* Association : Menghubungkan link antar elemen.
* Generalization : Disebut juga inheritance, sebuah elemen yang merupakan spesialisasi dari elemen lainnya.
* Dependency : Sebuah elemen yang bergantung dalam beberapa elemen lainnya.
* Aggregation : bentuk association dimana sebuah elemen berisi elemen lainnya.

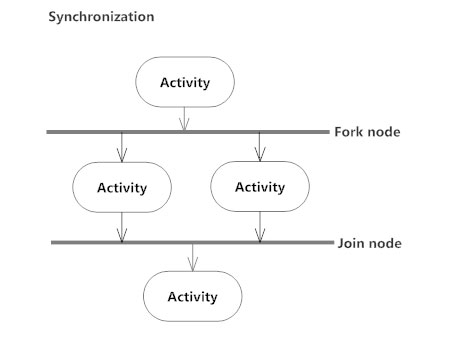
1. Activity Diagram

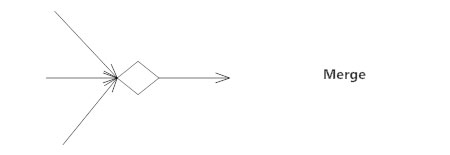
Activity Diagram adalah diagram yang menggambarkan tentang aktifitas yang terjadi pada sistem dari pertama sampai akhir. Diagram ini menunjukkan langkah-langkah dalam proses kerja sistem yang kira buat. Sebagai contoh, langkah-langkah memasak air. Struktur diagram ini juga mirip dengan *flowchart*.

Komponen yang ada pada Activity Diagram antara lain :

* **Start Point**Start Point merupakan lingkaran hitam kecil, yang menandakan tindakan awal atau titik awal aktivitasuntuk setiap diagram aktivitas.  
  
* **Activitiy(Aktivitas)**  
  Activity menunjukan aktivitas yang dilakukan atau yang sedang terjadi dalam activity diagram.  
  
* **Action Flow(Arah)**  
  Action Flow digunakan untuk transisi dari suatu tindakan ke tindakan yang lain atau menunjukan aktivitas selanjutnya setelah aktivitas sebelumnya.  
  
* **Decision(Keputusan)**  
  Decision adalah suatu titik atau point pada activity diagram yang mengindikasikan suatu kondisi dimana ada kemungkinan perbedaan transisi.  
  
* **Synchornization**  
  Synchornization dibagi menjadi 2 yaitu fork dan join.

Fork (percabangan) digunakan untuk memecah behaviour menjadi activity atau action yang paralel.

Join (penggabungan) untuk menggabungkan kembali activity atau action yang paralel.  


* **Merge Event(Menggabungkan)**  
  Merge Event berfungsi untuk menggabungkan flow yang dipecah oleh decission.  
  
* **Swimlanes**  
  Swimlanes berfungsi untuk memecah activity diagram menjadi baris dan kolom untuk membagi tangung jawab obyek-obyek yang melakukan aktivitas.
* **Final State/End Point(Titik Akhir)**  
  Final State menunjukan bagian akhir dari aktivitas.  
  

# BAB 3

# METODE PENELITIAN DAN PERANCANGAN

## Metodologi Penelitian

* + 1. Pengumpulan Data

Dalam Pembuatan Aplikasi TeachMe diperlukan adanya teknik pengumpulan data. Adapun data yang dikumpulkan berisi tentang kebutuhan orang tua selaku pelanggan terhadap pendidikan anak di luar jam sekolah, data terkait standar akademik bagi setiap mata pelajaran, data apa saja uang menjadi tahapan *recruitment* seorang guru, serta data terkait tugas dan peranan guru dalam proses belajar. Data tersebut dapat terkumpul dari observasi langsung terhadap prilaku orang tua terhadap pendidikan anaknya, dan studi literatur perihal poin-poin penting dalam membangun sebuah bimbel.

* + 1. Analisis

Pada proses analisis, peneliti merencanakan potensi dan tantangan terhadap pengembangan sebuah aplikasi yang bertujuan untuk mengidentifikasi berbagai faktor secara sistematis untuk merumuskan suatu strategi dalam pembuatan aplikasi dengan memanfaatkan data yang telah dikumpulkan sebagai informasi menjadi model-model kuantitatif perumusan strategi.

* + 1. Perancangan Sistem

Setelah menganalisis dan mengetahui kebutuhan apa saja yang diperlukan untuk membangun suatu aplikasi, peneliti melakukan tahap perancangan sistem terlebih dahulu, tujuannya agar pada proses pembuatan sistem ini lebih terstuktur. pertama peneliti membuat sebuah rancangan UML (*Unfied Modeling Language*) yaitu berupa *Use Case Diagram*, dan *Activity Diagram* serta rancangan tampilan antarmuka aplikasi berupa *Mock-Up*.

* + 1. Coding

Tahap *coding* atau kode program komputer yang merupahan hasil terjemahan dari desain logis dan fisik. Pada tahap ini, desain UML diimplementasukan kepada baris kode-kode program dalam bahasa Java Android. Pada prosesnya, peneliti mengunakan Android Studio sebagai Integrated Development Environmen (IDE), dengan tujuan dari implementasi ini ialah menghasilkan produk awal, dan selanjutnya dapat dites atau sijalankan dalam komputer untuk memastikan apakah hasilnya sesuai dengan hasil yang diinginkan atau tidak.

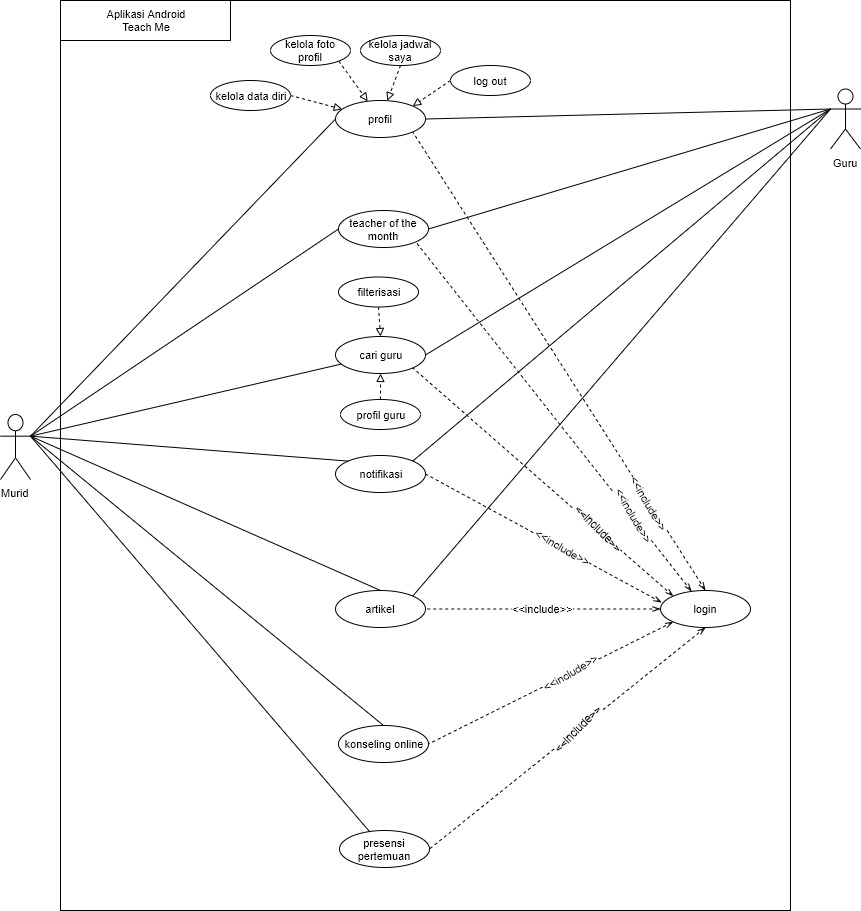
* + 1. Testing

Pada pengembangan akhir aplikasi akan diujicoba pada emulator dan perangkat *handphone*. Dalam tahap pengujian pertama, dilakukan secara pribadi oleh peneliti tanpa ada peranserta pihak lain dengan tujuan hasil produk sudah sesuai dengan tujuan yang diharapkan sebelumnya. Selanjutnya juka produk lulus pada tahap uji pertama, produk akan memasuki sesi pengujian lapangan dengan tujuan mengetahui tanggapan pengguna secara langsung.

* + 1. Maintenance

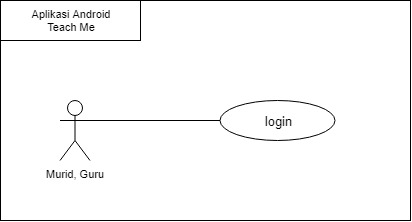
Tahapan terakhir ialah tahapan lanjutan yang menuntut pemgembang dalam melakukan perubahan ataupun pengembangan hingga memperbaiki fungsionalitas aplikasi TeachMe sesuai permintaan komunitas (*users*) dengan tujuan membuat aplikasi ini dapat berjalan lebih baik lagi.

## Use Case Diagram



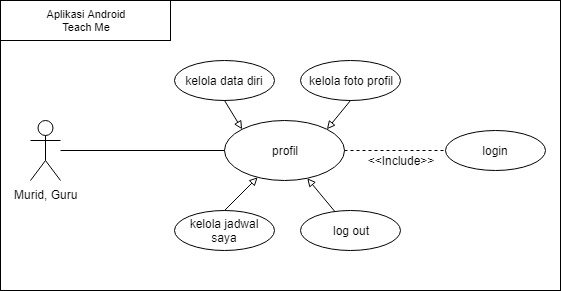
## Spesifikasi Use Case Diagram

* + 1. Spesikasi Use Case Diagram : Login



|  |  |
| --- | --- |
| Nama Use Case | Login |
| Aktor | Murid atau Guru |
| Aktor pendukung | - |
| Deskripsi singkat | Use case ini digunakan oleh aktor untuk melakukan login dengan menggunakan username dan password agar dapat masuk ke dalam sistem. |
| Kondisi sebelum | 1.Aktor telah mengakses sistem  2.Aktor mempunyai akun yang terdaftar |
| Alur dasar | 1. Use case ini dimulai ketika aktor memilih fungsi login  2. Sistem menampilkan tampilan login  3. Aktor memasukkan username dan password  4. Sistem memeriksa username dan password yang dimasukkan oleh aktor  E.1 - Username dan password salah  5. Sistem memberikan akses dan menampilkan form beranda aktor  6. Use case selesai |
| Alur alternatif | - |
| Alur kesalahan | E.1 - 1. Username dan password salah  2. Sistem menampilkan peringatan bahwa data yang diinputkan salah  3. Aktor kembali pada alur dasar langkah ke-3 |
| Kondisi sesudah | Aktor berhasil melakukan fungsi login dan masuk kedalam sistem |

* + 1. Spesifikasi Use Case Diagram : Profil



|  |  |
| --- | --- |
| Nama Usecase | Profil |
| Aktor | Murid atau Guru |
| Aktor pendukung | - |
| Deskripsi singkat | Use case ini digunakan oleh aktor ke menu profil |
| Kondisi sebelum | 1. Aktor telah melakukan use case login <<include login>>  2. Sistem mengecek hak akses  3. Aktor berhasil login sebagai user murid atau guru |
| Alur dasar | 1. Use case ini dimulai ketika aktor memilih fungsi profil  2. Sistem menampilkan tampilan profil yang ada dalam sistem  3. Aktor memilih salah satu fungsi profil  A.1 – Kelola data diri  A.2 – Kelola foto profil  A.3 - kelola jadwal saya  A.4- log out  4. Sistem menampilkan tampilan yang dipilih aktor  5. Use case selesai |
| Alur alternative | - |
| Alur kesalahan | - |
| Kondisi sesudah | Aktor berhasil meangkses fungsi profil. |

* + 1. Spesifikasi Use Case Diagram : Kelola Data Diri

|  |  |
| --- | --- |
| Nama Usecase | Kelola data diri |
| Aktor | Murid atau Guru |
| Aktor pendukung | - |
| Deskripsi singkat | Use case ini digunakan oleh aktor untuk melakukan kelola data diri |
| Kondisi sebelum | 1. Aktor telah melakukan use case login <<include login>>  2. Sistem mengecek hak akses  3. Aktor berhasil login sebagai user murid atau guru  4. Aktor telah memilih fungsi profil |
| Alur dasar | 1. Use case ini dimulai ketika aktor memilih fungsi kelola data diri  2. Sistem menampilkan tampilan untuk kelola data diri  3. Aktor dapat melakukan kelola data profil  E.1 - data tidak lengkap  4. Sistem menampilkan pesan bahwa data profil berhasil diperbaharui  5. Use case selesai |
| Alur alternative | - |
| Alur kesalahan | E.1 - 1. Data tidak lengkap  2. Sistem menampilkan peringatan bahwa data yang diinputkan tidak lengkap  3. Aktor kembali pada alur dasar langkah ke-2 |
| Kondisi sesudah | Aktor berhasil melakukan fungsi kelola data diri |

* + 1. Spesifikasi Use Case Diagram : Kelola Foto Profil

|  |  |
| --- | --- |
| Nama Usecase | Kelola foto profil |
| Aktor | Murid atau Guru |
| Aktor pendukung | - |
| Deskripsi singkat | Use case ini digunakan oleh aktor untuk kelola foto diri |
| Kondisi sebelum | 1. Aktor telah melakukan use case login <<include login>>  2. Sistem mengecek hak akses  3. Aktor berhasil login sebagai user murid atau guru  4. Aktor telah memilih fungsi profil. |
| Alur dasar | 1. Use case ini dimulai ketika aktor memilih fungsi kelola foto profil  2. Sistem menampilkan tampilan untuk kelola foto profil  3. Aktor dapat melakukan kelola foto profil  E.1 – Kapasitas foto terlalu besar  4. Sistem menampilkan pesan bahwa kelola foto profil berhasil  5. Use case selesai |
| Alur alternative | - |
| Alur kesalahan | E.1 - 1. Kapasitas foto terlalu besar atau terlalu kecil  2. Sistem menampilkan peringatan bahwa foto terlalu besar  3. Aktor kembali pada alur dasar langkah ke-2 |
| Kondisi sesudah | Aktor berhasil melakukan kelola foto profil dan foto profil di dalam sistem berhasil diperbaharui, |

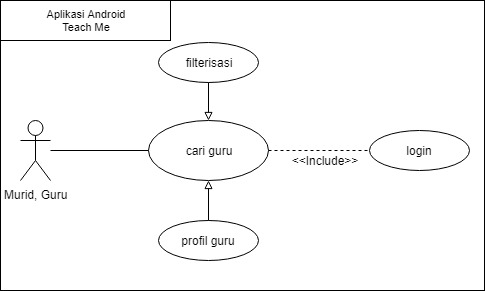
* + 1. Spesifikasi Use Case Diagram : Kelola Jadwal Saya

|  |  |
| --- | --- |
| Nama Usecase | Kelola Jadwal saya |
| Aktor | Murid atau Guru |
| Aktor pendukung | - |
| Deskripsi singkat | Use case ini digunakan oleh aktor untuk kelola jadwal saya. |
| Kondisi sebelum | 1. Aktor telah melakukan use case login <<include login>>  2. Sistem mengecek hak akses  3. Aktor berhasil login sebagai user murid atau guru  4. Aktor telah memilih fungsi profil. |
| Alur dasar | 1. Use case ini dimulai ketika aktor memilih fungsi profil  2. Sistem menampilkan tampilan untuk kelola jadwal saya  3. Aktor dapat melakukan kelola jadwal saya  4. Sistem menampilkan pesan bahwa kelola jadwal saya berhasil  5. Use case selesai |
| Alur alternative | - |
| Alur kesalahan | - |
| Kondisi sesudah | Aktor berhasil melakukan fungsi kelola jadwal saya. |

* + 1. Spesifikasi Use Case Diagram : Logout

|  |  |
| --- | --- |
| Nama Usecase | Logout |
| Aktor | Murid atau Guru |
| Aktor pendukung | - |
| Deskripsi singkat | Use case ini digunakan oleh aktor untuk keluar dari sistem |
| Kondisi sebelum | 1. Aktor telah melakukan use case login <<include login>>  2. Sistem mengecek hak akses  3. Aktor berhasil login sebagai user murid atau guru  4. Aktor telah memilih fungsi logout . |
| Alur dasar | 1. Use case ini dimulai ketika aktor memilih fungsi profil  2. Sistem menampilkan tampilan untuk logout  3. Aktor memilih fungsi logout  E-1 Sistem menampilkan pesan bahwa aktor logout dari sistem  4. Sistem menampilkan pesan bahwa log out berhasil  5. Use case selesai |
| Alur alternative | - |
| Alur kesalahan | E.1 - 1. Sistem menampilkan pesan bahwa aktor logout dari sistem  2. Sistem menampilkan peringatan pesan bahwa logot dari sistem  3. Aktor kembali pada alur dasar langkah ke-3 |
| Kondisi sesudah | Aktor berhasil melakukan fungsi logout . |

* + 1. Spesifikasi Use Case Diagram : Cari Guru



|  |  |
| --- | --- |
| Nama Usecase | Cari Guru |
| Aktor | Murid atau Guru |
| Aktor pendukung | - |
| Deskripsi singkat | Use case ini digunakan oleh aktor ke menu Cari guru |
| Kondisi sebelum | 1. Aktor telah melakukan use case login <<include login>>  2. Sistem mengecek hak akses  3. Aktor berhasil login sebagai user murid atau guru |
| Alur dasar | 1. Use case ini dimulai ketika aktor memilih fungsi cari guru  2. Sistem menampilkan tampilan cari guru yang ada dalam sistem  3. Aktor memilih salah satu fungsi cari guru  A.1 – Filterisasi  A.2 – Profil guru  4. Sistem menampilkan tampilan yang dipilih aktor  5. Use case selesai |
| Alur alternative | - |
| Alur kesalahan | - |
| Kondisi sesudah | Aktor berhasil melakukan fungsi cari guru. |

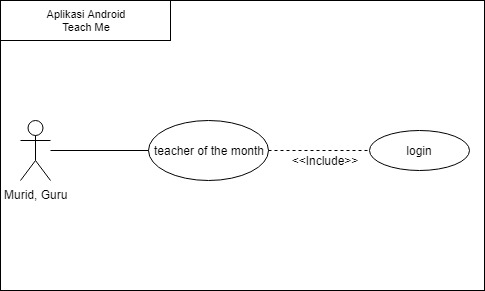
* + 1. Spesifikasi Use Case Diagram : Filterisasi

|  |  |
| --- | --- |
| Nama Usecase | Filterisasi |
| Aktor | Murid atau Guru |
| Aktor pendukung | - |
| Deskripsi singkat | Use case ini digunakan oleh aktor untuk melakukan filterisasi guru-guru agar sistem menampilkan guru yang sesuai dengan kebutuhannya |
| Kondisi sebelum | Aktor telah mengakses sistem |
| Alur dasar | 1. Use case ini dimulai ketika aktor memilih fungsi cari guru.  2. Sistem menampilkan halaman filterisasi dan aktor memasukkan data yang sesuai dengan kebutuhannya.  3. Sistem menampilkan daftar guru yang sesuai dengan kebutuhan pengguna  4. Use case selesai. |
| Alur alternative | - |
| Alur kesalahan | - |
| Kondisi sesudah | Aktor berhasil melakukan filterisasi cari guru dan sistem menampilkan hasil pencaharian yang sesuai dengan kebutuhan murid. |

* + 1. Spesifikasi Use Case Diagram : Profil Guru

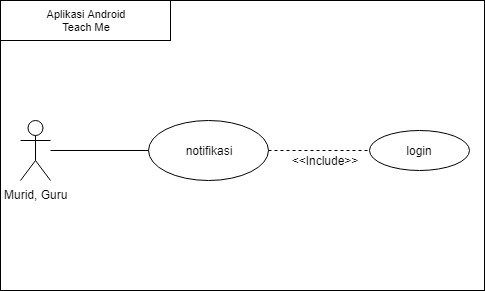
|  |  |
| --- | --- |
| Nama Usecase | Profil Guru |
| Aktor | Murid atau Guru |
| Aktor pendukung | - |
| Deskripsi singkat | Use case ini digunakan oleh aktor untuk melihat informasi profil guru. |
| Kondisi sebelum | Aktor telah mengakses sistem. |
| Alur dasar | 1. Use case ini dimulai ketika aktor memilih fungsi cari guru.  2. Sistem menampilkan tampilan daftar guru yang tersimpan dalam sistem.  3. Aktor dapat memilih salah satu guru yang ingin ditampilkan datanya.  4. Sistem menampilkan tampilan deskripsi profil guru yang dipilih aktor.  5. Use case selesai. |
| Alur alternative | - |
| Alur kesalahan | - |
| Kondisi sesudah | Aktor berhasil mengakses halaman profil guru. |

* + 1. Spesifikasi Use Case Diagram : Teacher of The Month



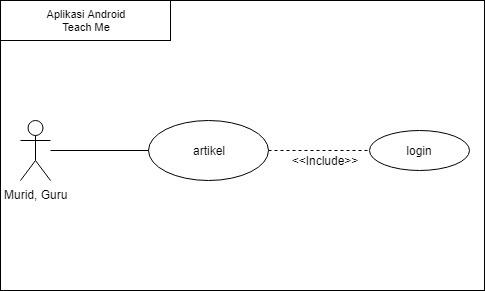
|  |  |
| --- | --- |
| Nama Usecase | Teacher of the month |
| Aktor | Murid atau Guru |
| Aktor pendukung | - |
| Deskripsi singkat | Use case ini digunakan oleh aktor untuk melakukan Melihat guru favorit di aplikasi Teach Me. |
| Kondisi sebelum | 1. Aktor telah melakukan use case login <<include login>>  2.Sistem mengecek hak akses  3.Aktor berhasil login sebagai user murid atau guru |
| Alur dasar | 1. Use case ini dimulai ketika aktor memilih fungsi Teacher of The Month.  2. Sistem menampilkan tampilan daftar teacher of the month pada sistem di halaman beranda  3. Aktor memilih salah satu guru untuk ditampilkan datanya  4. Sistem menampilkan halaman profil guru yang diinginkan murid  5. Use case selesai |
| Alur alternative | - |
| Alur kesalahan | - |
| Kondisi sesudah | Aktor berhasil mengakses fungsi teacher of the month |

* + 1. Spesifikasi Use Case Diagram : Notifikasi



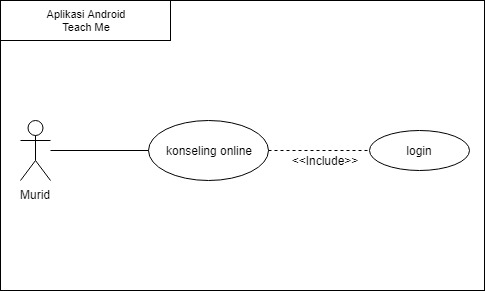
|  |  |
| --- | --- |
| Nama Usecase | Notifikasi |
| Aktor | Murid atau Guru |
| Aktor pendukung | - |
| Deskripsi singkat | Use case ini digunakan oleh aktor untuk melihat notifikasi yang masuk di dalam akun pengguna Teach Me baik itu murid atau guru. |
| Kondisi sebelum | 1. Aktor telah melakukan use case login <<include login>>  2.Sistem mengecek hak akses  3.Aktor berhasil login sebagai user murid atau guru |
| Alur dasar | 1. Use case ini dimulai ketika aktor memilih fungsi notifikasi  2. Sistem menampilkan tampilan daftar notifikasi yang masuk  3. use case selesai |
| Alur alternative | - |
| Alur kesalahan | - |
| Kondisi sesudah | Aktor berhasil mengakses fungsi notifikasi dan melihat notifikasi yang masuk |

* + 1. Spesifikasi Use Case Diagram : Artikel



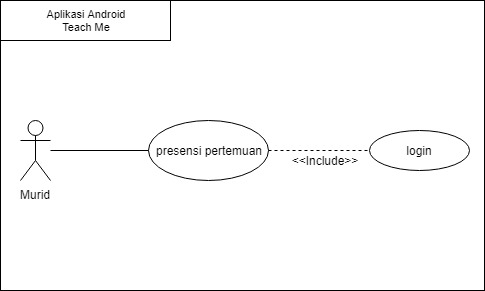
|  |  |
| --- | --- |
| Nama Usecase | Artikel |
| Aktor | Murid atau Guru |
| Aktor pendukung | - |
| Deskripsi singkat | Use case ini digunakan oleh aktor untuk melakukan melihat artikel |
| Kondisi sebelum | 1. Aktor telah melakukan use case login <<include login>>  2.Sistem mengecek hak akses  3.Aktor berhasil login sebagai user murid atau guru |
| Alur dasar | 1. Use case ini dimulai ketika aktor memilih fungsi artikel  2. Sistem menampilkan daftar artikel yang tersedia  3. Aktor memilih salah satu artikel untuk dibaca  4. Sistem menampilkan isi artikel yang dipilih aktor  5. Use case selesai |
| Alur alternative | - |
| Alur kesalahan | - |
| Kondisi sesudah | Aktor berhasil mengakses halaman artikel |

* + 1. Spesifikasi Use Case Diagram : Konseling Online



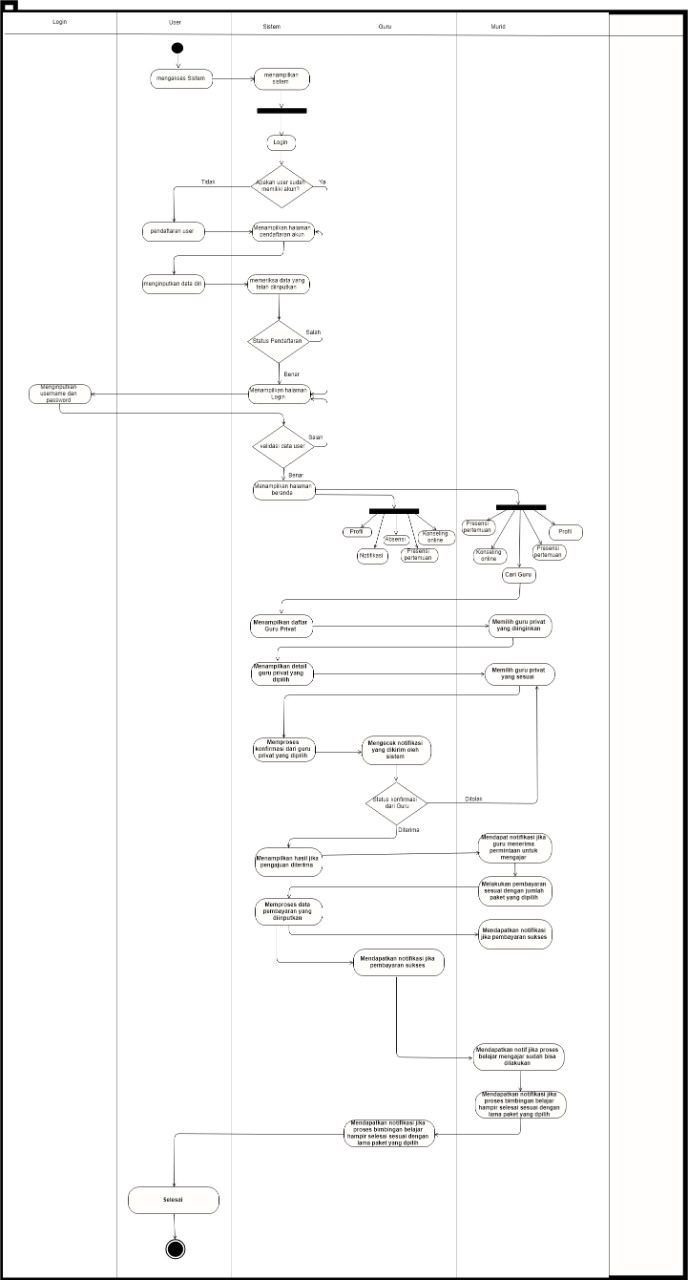
|  |  |
| --- | --- |
| Nama Usecase | Konseling online |
| Aktor | Murid |
| Aktor pendukung | - |
| Deskripsi singkat | Use case ini digunakan oleh aktor untuk konsultasi online |
| Kondisi sebelum | 1. Aktor telah melakukan use case login <<include login>>  2. Sistem mengecek hak akses  3. Aktor berhasil login sebagai user Murid |
| Alur dasar | 1. Use case ini dimulai ketika aktor memilih fungsi konseling online  2. Sistem menampilkan tampilan konseling online yaitu sebuah fitur chat  3. Aktor dapat mengirimkan pesan  4. Sistem menampilkan pesan yang dikirim actor  5. Aktor dapat menerima balasan pesan di dalam konseling online  6. Sistem menampilkan pesan yang masuk  7. Use case selesai |
| Alur alternative | - |
| Alur kesalahan | - |
| Kondisi sesudah | Aktor berhasil melakukan fungsi konseling online |

* + 1. Spesifikasi Use Case Diagram : Presensi Pertemuan



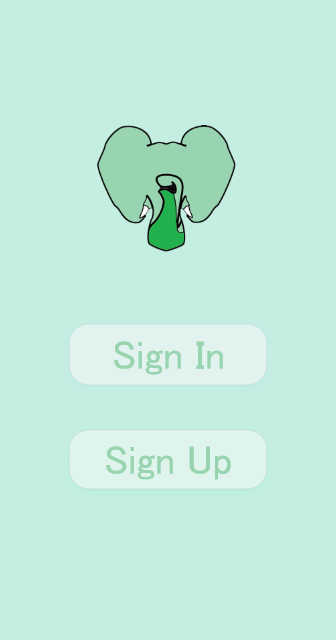
|  |  |
| --- | --- |
| Nama Usecase | Presensi pertemuan |
| Aktor | Murid |
| Aktor pendukung | - |
| Deskripsi singkat | Use case ini digunakan oleh aktor untuk melakukan presensi online setiap kali pertemuan |
| Kondisi sebelum | 1. Aktor telah melakukan use case login <<include login>>  2. Sistem mengecek hak akses  3. Aktor berhasil login sebagai murid  4. Aktor telah memilih fungsi presensi pertemuan |
| Alur dasar | 1. Use case ini dimulai ketika aktor memilih presensi  2. Sistem menampilkan tampilan presensi  3. Aktor dapat melakukan memasukkan data di presensi pertemuan seperti tanggal, deskripsi, kritik dan saran, serta lampiran foto  4. Data presensi pertemuan berhasil diunggah ke daam sistem  E.1 – Data tidak lengkap  5. Use case selesai |
| Alur alternative | - |
| Alur kesalahan | E.1 – 1. Data tidak lengkap |
| Kondisi sesudah | Aktor berhasil melakukan fungsi presensi pertemuan. |

## Activity Diagram

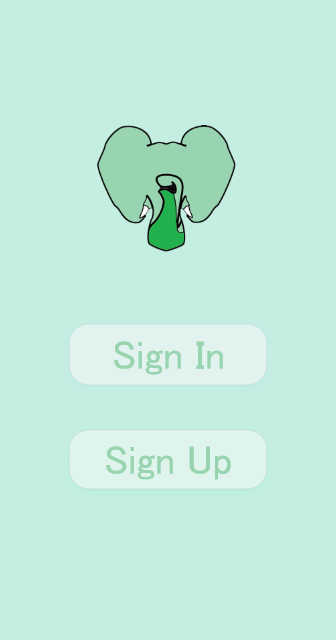


## Perancangan Antarmuka

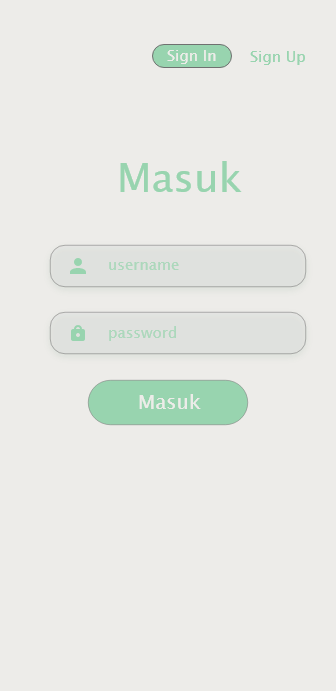
* + 1. Halaman Pembuka Aplikasi



* + 1. Halaman Login dan Sign Up



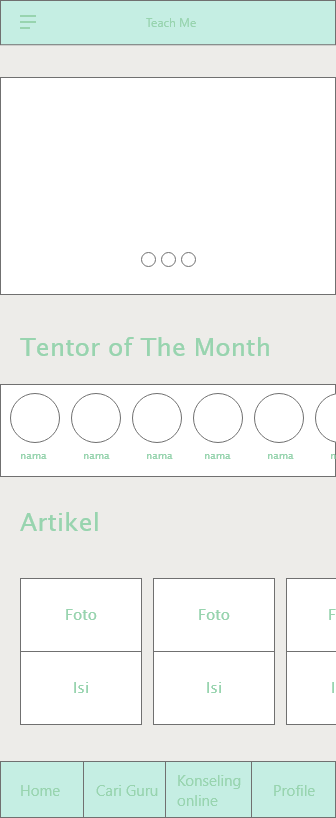
* + 1. Halaman Login



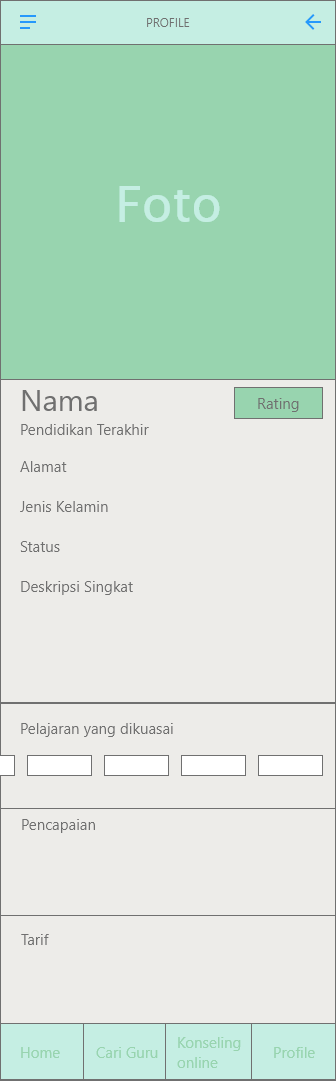
* + 1. Halaman Sign Up



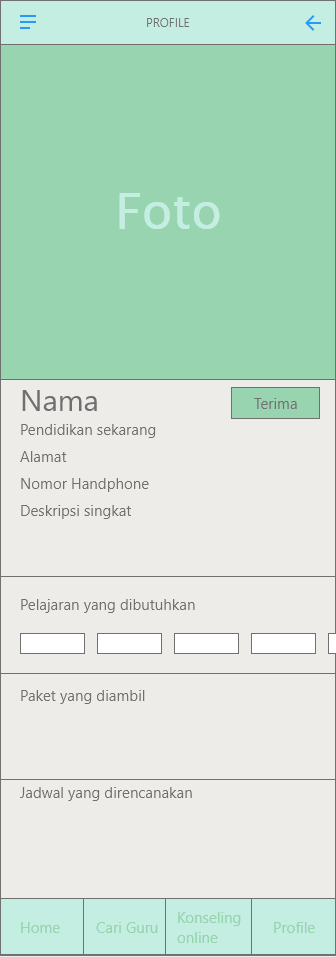
* + 1. Halaman Beranda



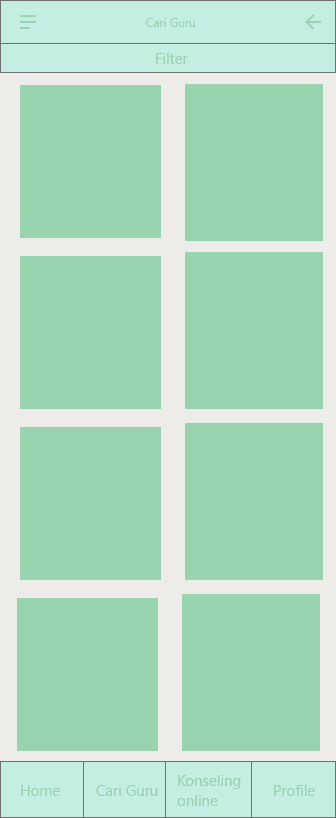
* + 1. Halaman Profil Guru



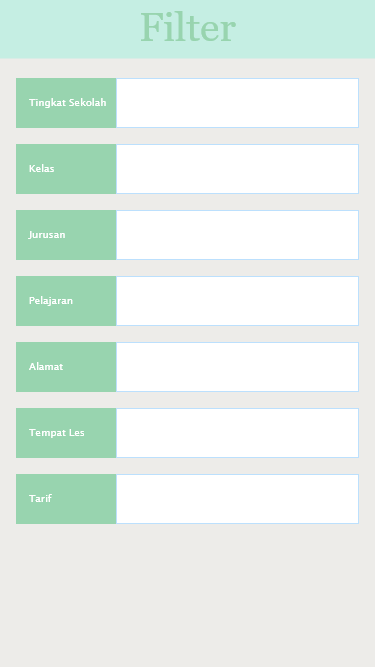
* + 1. Halaman Profil Siswa



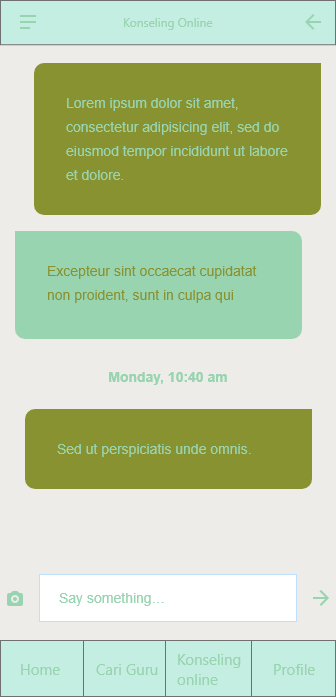
* + 1. Halaman Cari Guru



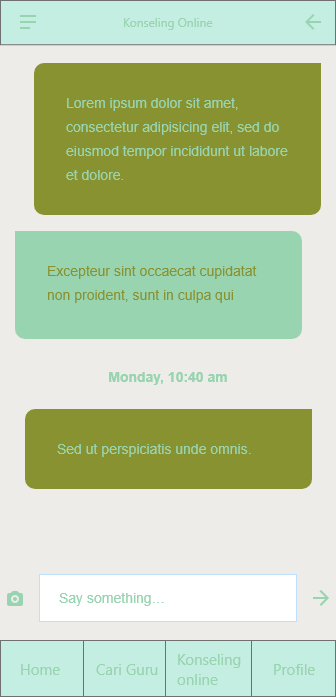
* + 1. Halaman Filterisasi Cari Guru



* + 1. Halaman Konseling Online



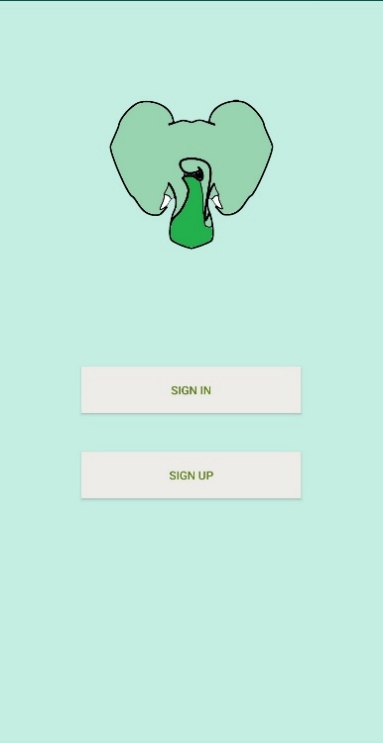
* + 1. Halaman Pengaturan



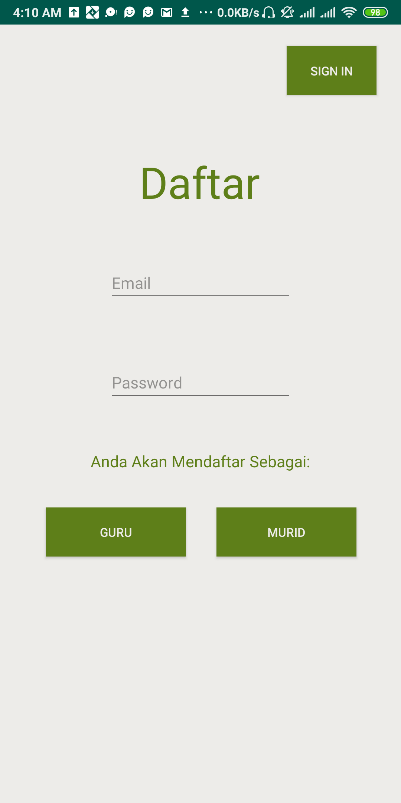
# BAB 4

# IMPLEMENTASI SISTEM

## Halaman Awal



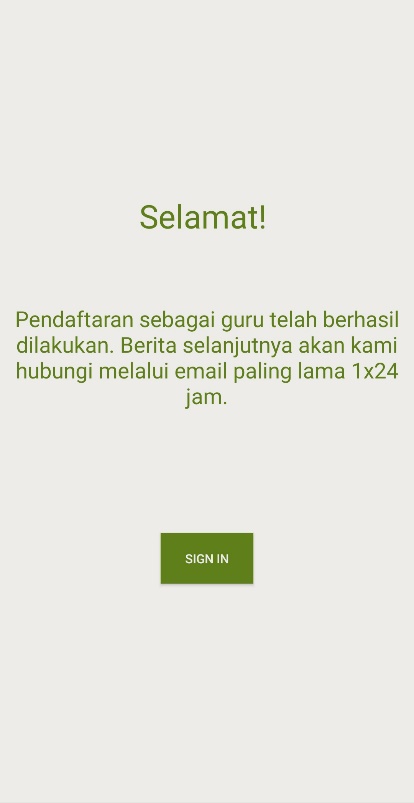
## Pendaftaran Akun



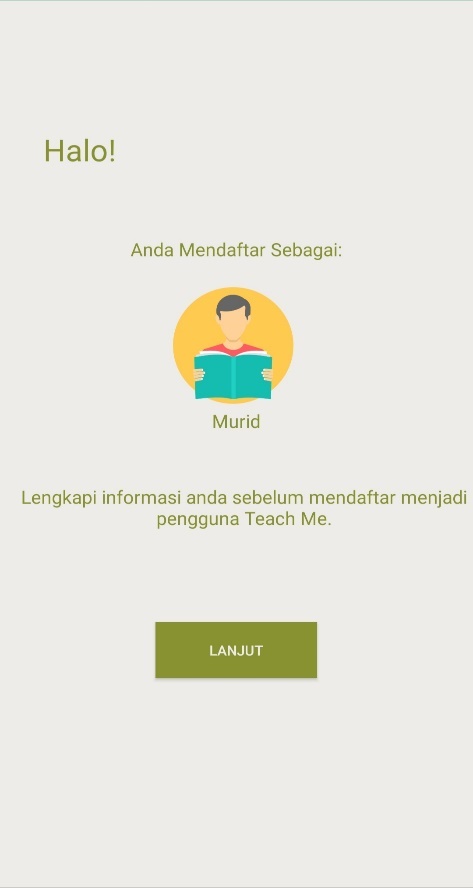
## Pendaftaran Akun Sebagai Guru



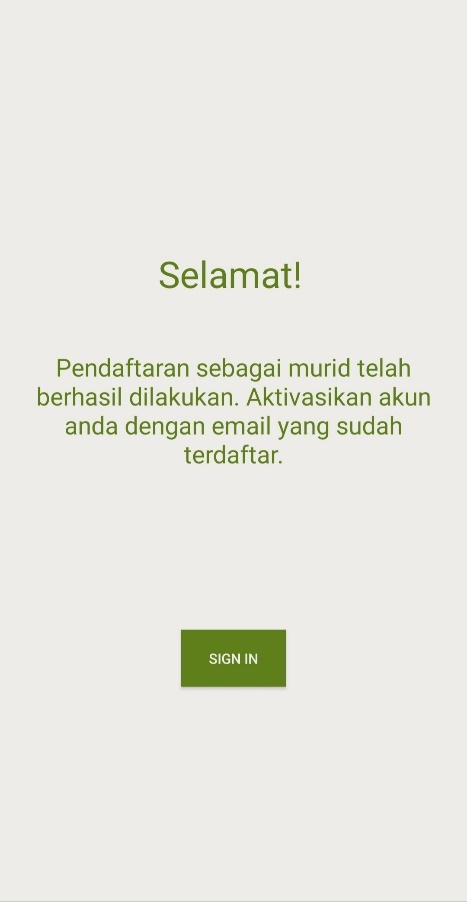




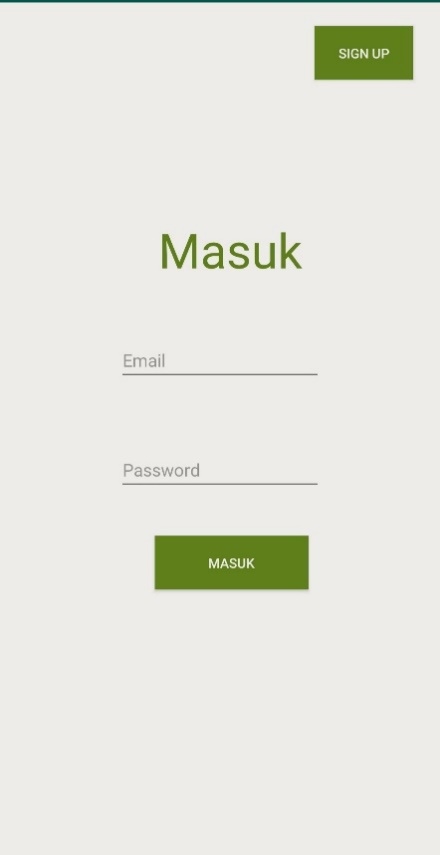
## Pendaftaran Akun Sebagai Murid



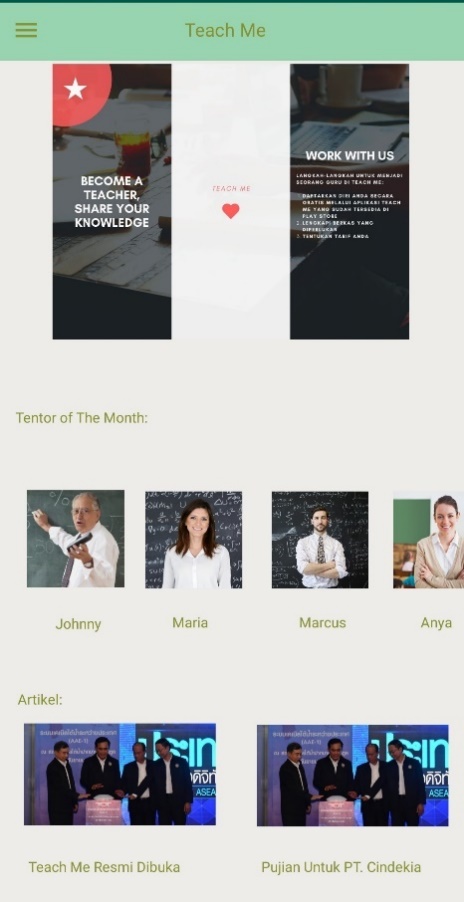




## Halaman Login



## Halaman Beranda



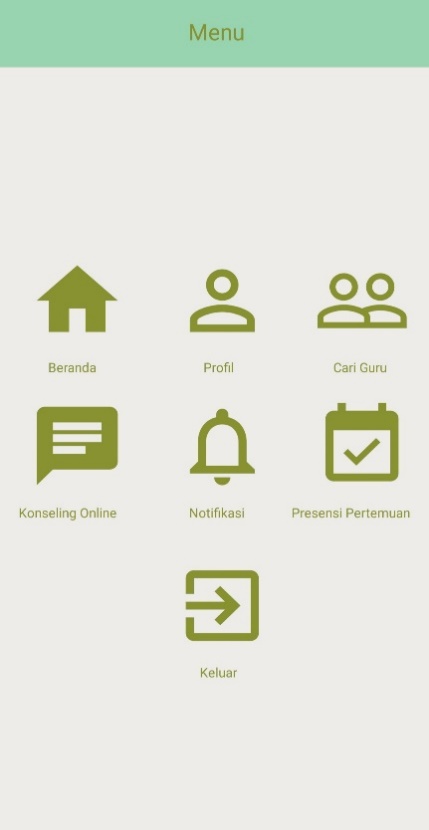
## Profil Guru



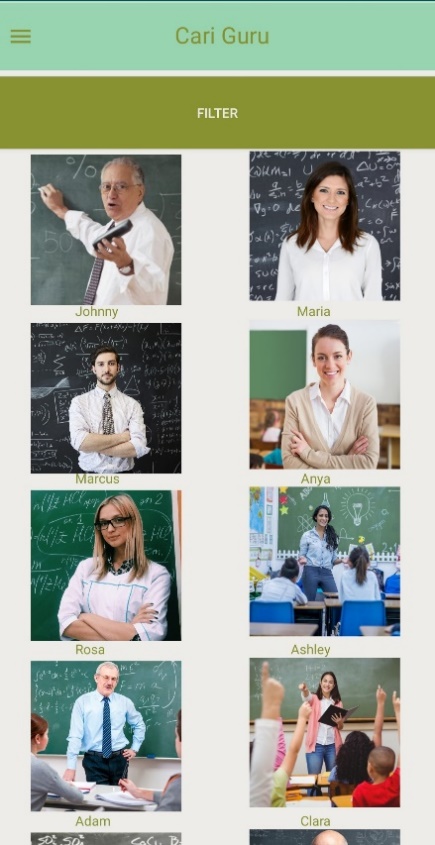
## Profil Siswa



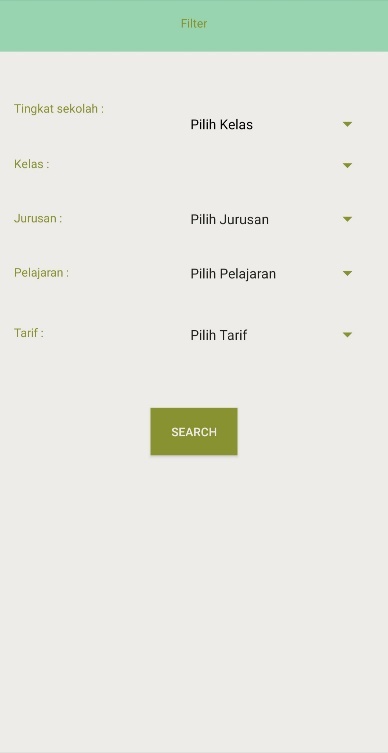
## Halaman Menu



## Cari Guru



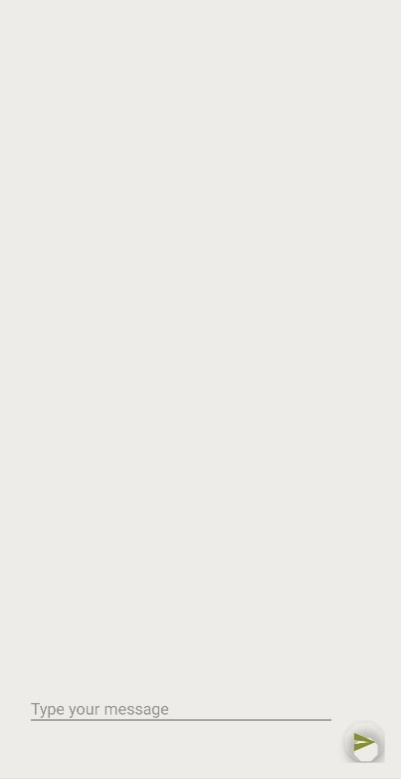
## Filterisasi Cari Guru



## Konfirmasi Pemesanan



## Konseling Online



## Notifikasi



## Presensi Online



# BAB 5

# KESIMPULAN DAN SARAN

## 5.1 Kesimpulan

Aplikasi Teach Me adalah aplikasi berbasis android yang dikembangkan oleh PT. Cendikia dengan tujuan untuk menyediakan tenaga pengajar bagi orang tua untuk memberikan bimbingan belajar langsung ke rumah untuk anak-anak mereka. Teach Me menyediakan beberapa fitur, antara lain: Cari Guru, Konseling Online, Jadwal Saya, Presensi Online, Guru Favorit Bulan Ini, dan Artikel.

Aplikasi berbasis android ini dikembangkan dengan bahasa pemrograman Java melalui IDE Android Studio. Diharapkan dengan dikembangkannya aplikasi Teach Me ke dalam android dapat memberikan akses kemudahan kepada penggunanya dalam mengakses aplikasi Teach Me dan lebih cepat menerima manfaat yang diberikan oleh aplikasi Teach Me.

# 5.2 Saran

Adapun saran yang dapat diberikan untuk pengembangan lebih lanjut dari aplikasi Teach Me, antara lain:

1. Aplikasi Teach Me dapat dikembangkan ke dalam iOS sehingga memperluas sasaran pengguna.
2. Pengembangan lebih lanjut aplikasi Teach Me dapat berupa rekomendasi tenaga pengajar yang sesuai dengan kriteria pengguna yang terdapat pada pendaftaran.
3. Pada fitur Artikel, konten yang disediakan tidak hanya berbentuk teks dan gambar. Tapi juga dapat berupa video dan audio mengenai pembelajaran dan berita-berita menarik di bidang pendidikan, sehingga dapat memberikan bentuk baru dalam memberikan pembelajaran kepada para pengguna yang masing-masing memiliki cara memahami sesuatu yang berbeda.