

BAB III

ANALISIS DAN PERANCANGAN

3.1 Analisis Sistem

Analisis sistem bertujuan untuk mengidentifikasi permasalahan-permasalahan yang terdapat pada sistem serta menentukan kebutuhan-kebutuhan dari sistem yang akan dibangun. Analisis tersebut meliputi analisis masalah, spesifikasi kebutuhan perangkat lunak, analisis kebutuhan fungsional, analisis kebutuhan non fungsional.

3.1.1 Analisis Masalah

Perguruan Tinggi adalah satuan pendidikan yang menyelenggarakan pendidikan tinggi dan dapat berbentuk Akademi, Politeknik, Sekolah tinggi, Institut, atau Universitas. Perguruan Tinggi di kota Bandung pada tahun 2017 berjumlah 107 untuk Swasta diantaranya 20 Universitas, 2 Institut, 50 Sekolah Tinggi, 25 Akademi, dan 10 Politeknik [1]. Sedangkan untuk Perguruan Tinggi Negeri di Kota Bandung berjumlah 11 [2].

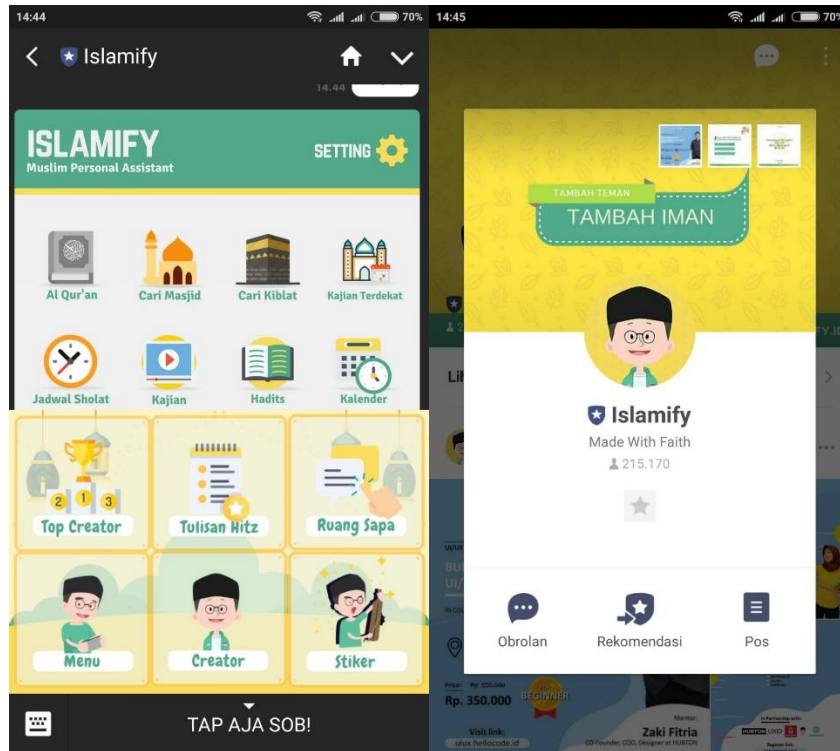
Untuk mendapatkan Informasi Perguruan Tinggi yang diminatinya, saat ini siswa SMA/K dalam mencari informasi Perguruan Tinggi menggunakan cara mengunjungi setiap website Perguruan Tinggi satu persatu yang diinginkan untuk mendapatkan informasi mengenai Perguruan Tinggi. Berdasarkan data kuisioner terhadap 68 responden siswa SMA/K yang akan melanjutkan sekolahnya ke Perguruan Tinggi, terdapat 86% siswa yang masih mengalami kebingungan dalam menentukan Perguruan Tinggi yang diminatinya. Lalu berdasarkan data kuisioner yang sama, faktor peminat siswa SMA/K dalam mencari Perguruan Tinggi diantaranya berdasarkan Akreditasi, Lokasi, Biaya perkuliahan serta Jurusan yang diminatinya .

3.1.2 Analisis Aplikasi Sejenis

Berdasarkan analisis yang dilakukan terhadap beberapa bot yang dilakukan oleh penulis diperoleh beberapa masukan yang digunakan dalam penelitian ini yakni sebagai berikut :

1. Islamify

Islamify merupakan bot pada aplikasi pesan instal Line yang menyediakan informasi mengenai ayat Al Qur'an, lokasi masjid, kiblat, jadwal sholat dan kajian agama. Gambar 3.1 merupakan tampilan bot Islamify :

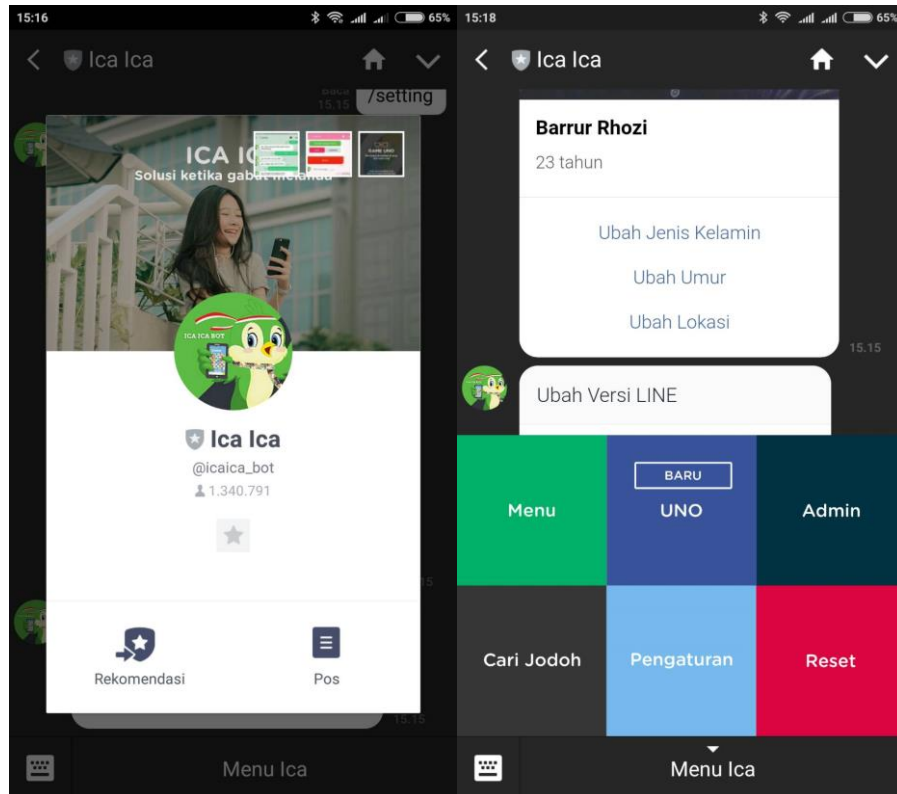


Gambar III.1 Bot Islamify

Islamify dijadikan sebagai contoh acuan pengembangan pada penelitian ini. Beberapa acuan yang diterapkan diantaranya yakni konsep konten yang digunakan di dalam bot (mengenai pencarian lokasi).

2. Ica Ica

Ica Ica merupakan bot pada pesan instan Line yang menyediakan layanan bermain game dan melakukan pencarian jodoh. Gambar III.2 merupakan tampilan bot Ica Ica.

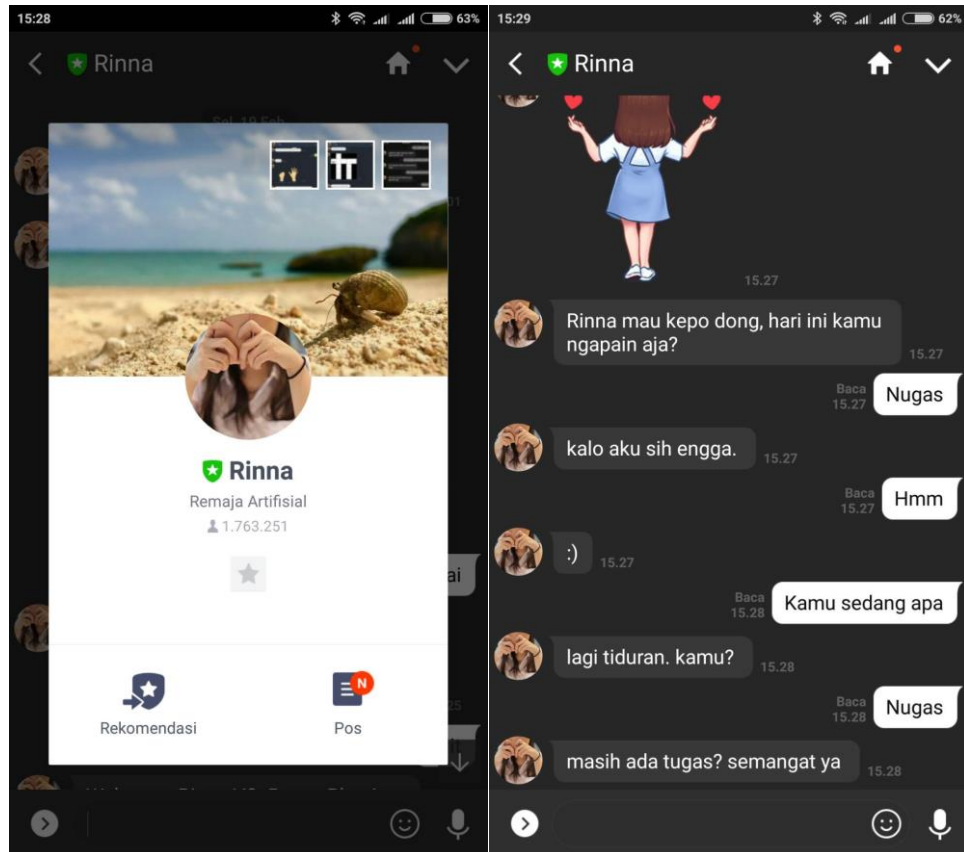


Gambar III.2 Ica Ica Bot

Ica Ica dijadikan acuan pengembangan dalam penelitian ini dalam bentuk fitur yang tersedia didalamnya (menu profil untuk melihat informasi umum mengenai akun Line yang digunakan oleh user dan tampilan menu dengan ukuran 2500x1686px yang didalamnya terdapat 6 menu).

3. Rinna

Rinna diciptakan oleh beberapa teknisi dari tim Microsoft Artificial Intelligence & Research dengan merancanganya sebagai Bot AI yang peka terhadap para pengguna. Rinna menggunakan bahasa seperti remaja pada umumnya, dan ia dapat membicarakan topik tentang wanita seperti gosip, artis, kadang-kadang film Korea, bahkan animasi atau fesyen. Gambar III.3 merupakan tampilan mengenai bot Rinna.

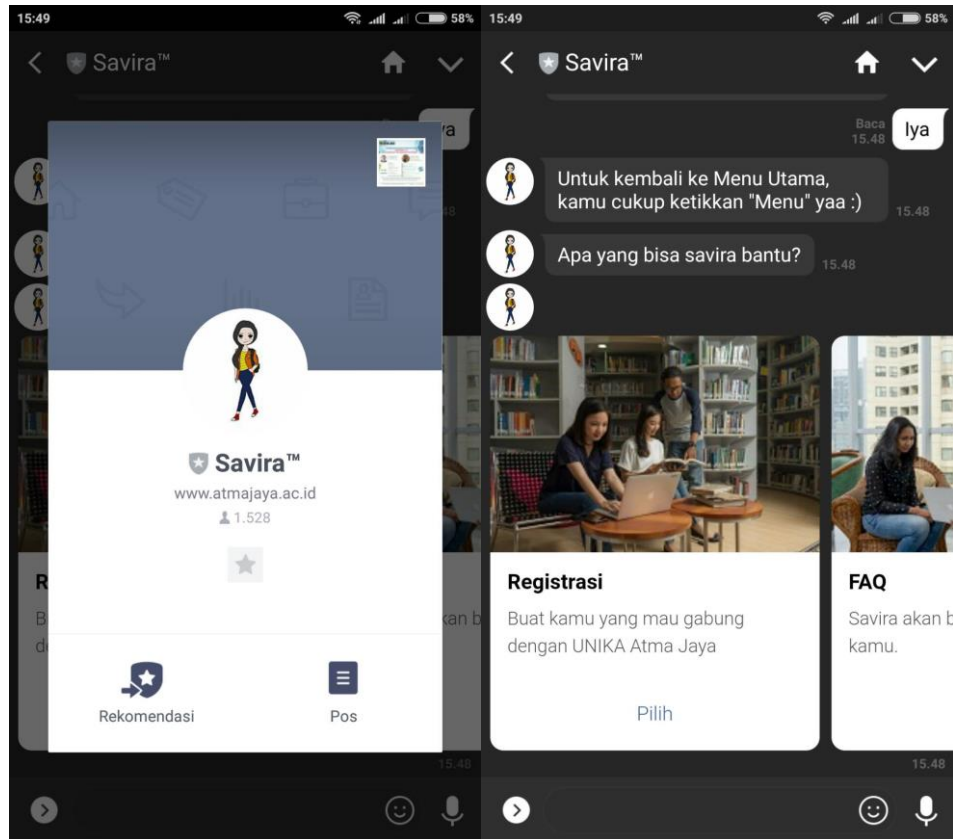


Gambar III.3 Bot Rinna

Bot Rinna dijadikan acuan pengembangan dalam penelitian ini dalam bentuk AI (Artificial Intelligence) atau kecerdasan buatan dalam pembuatan aplikasi chatbot yang akan diteliti.

4. SaviraTM

Savira atau Sahabat Virtual Atma Jaya adalah chatbot LINE yang dapat memberikan informasi seputar perkuliahan di Unika Atma Jaya. Calon mahasiswa dapat bertanya tentang pendaftaran mahasiswa baru, fasilitas, jurusan, prestasi, hingga terkait administrasi. Gambar III.4 merupakan tampilan mengenai Bot Savira.



Gambar III.4 Bot Savira

Bot Savira dijadikan sebagai contoh acuan pengembangan pada penelitian ini. Beberapa acuan yang diterapkan diantaranya yakni konsep penyediaan informasi mengenai kampus diantaranya informasi akreditasi, program studi, lokasi, dan biaya perkuliahan.

Berdasarkan berbagai aplikasi sejenis mengenai chatbot pada gambar diatas, maka dapat hasil analisis aplikasi bot dapat dilihat dibawah ini

Tabel III.1 Analisis Aplikasi Sejensi

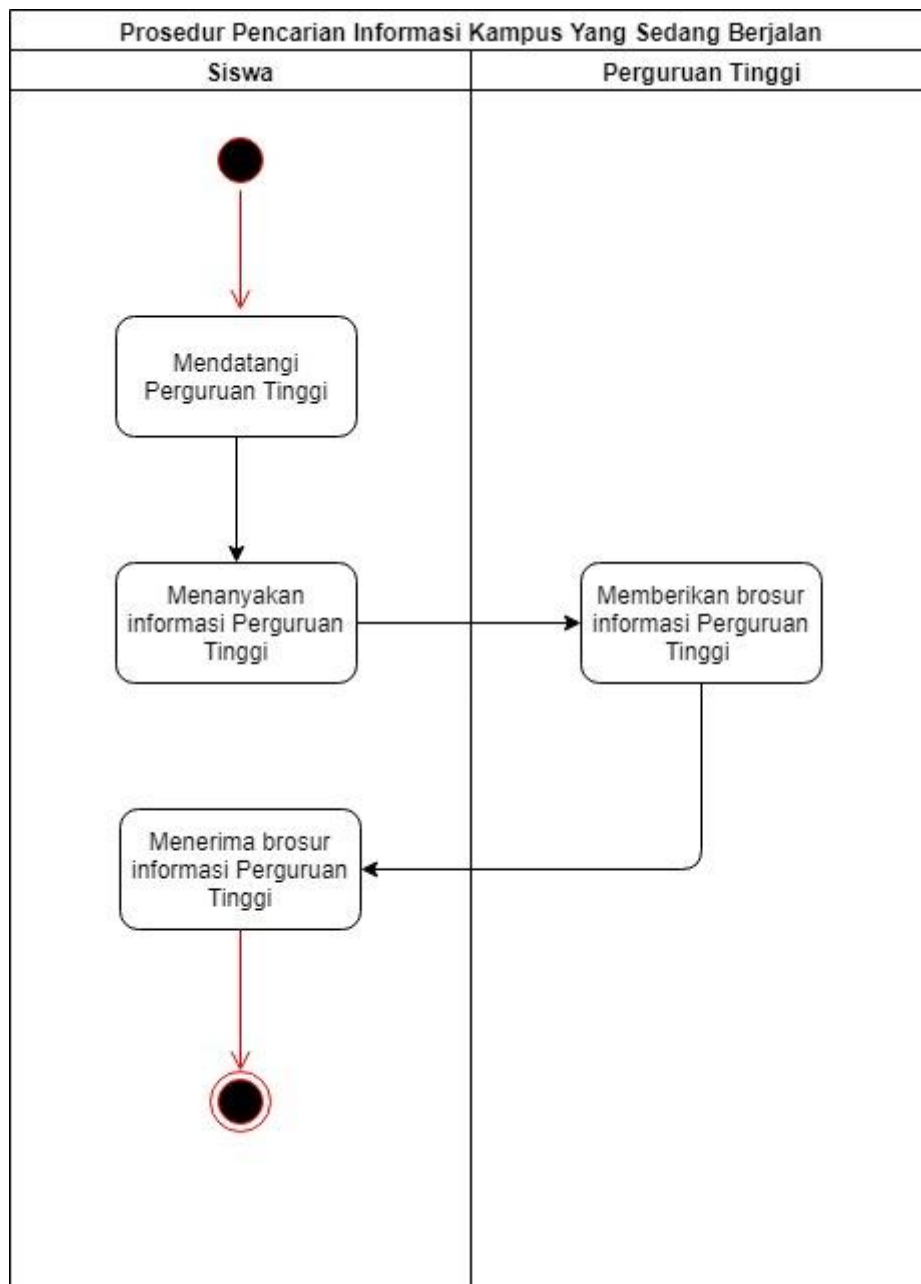
Parameter	Konten/Fitur	Menyimpan Informasi Pengguna	Menggunakan Rich Menu
Islamify	a. Al-Qur'an b. Lokasi Masjid c. Cari Kiblat	a. Nama Pengguna b. Jenis Kelamin	Ya

	d. Jadwal Sholat e. Kajian f. Hadits g. Kalender h. Stiker		
Icaica	a. Permainan (Cari kata, tebak kata, adu pinalti, dsb) b. Cari jodoh	a. Jenis Kelamin b. Tanggal Lahir c. Lokasi d. Versi LINE	Ya
Rinna	a. Permainan (Gunting, Batu, Kertas, Escape Room) b. Menggunakan Teknologi AI c. Dapat bercakap dalam bahasa alami	a. Nama pengguna b. Jenis Kelamin	Tidak
Savira	a. Registrasi kuliah b. FAQ c. Pindah prodi d. Pindah kampus e. Salah transfer f. Payment Plan	a. Nama pengguna b. Asal sekolah c. Email	Tidak
SkyBot	a. Cari Kampus b. Cari Jurusan c. Cari Biaya d. Cari Lokasi e. Profile f. Panduan	a. Nama Pengguna b. User Id LINE c. Foto pengguna	Ya

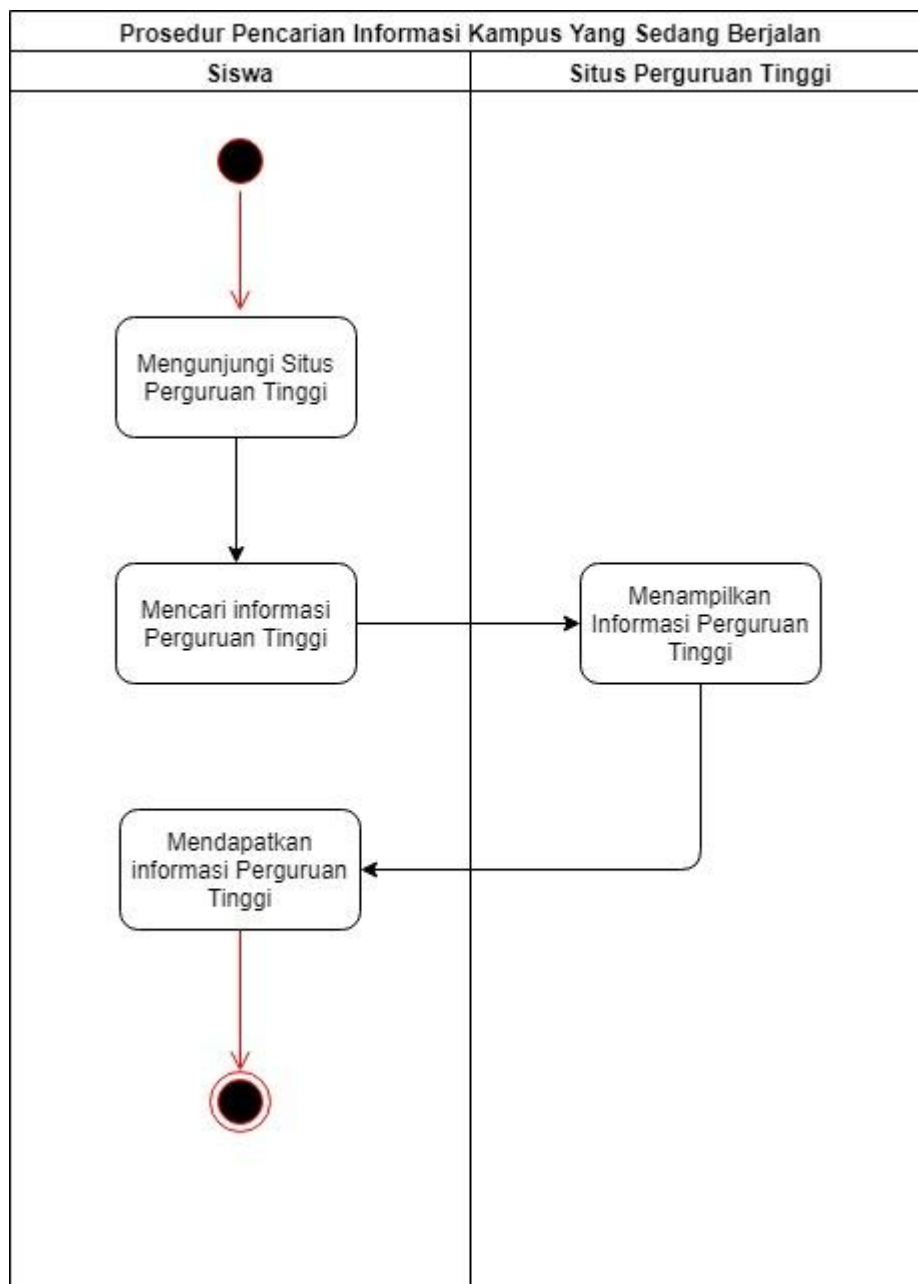
3.2 Analisis Prosedur Yang Berjalan

Prosedur pencarian informasi Perguruan Tinggi yang terjadi pada siswa sekolah SMA/K, biasanya harus mencari melalui brosur perguruan tinggi atau

mendatangi Perguruan Tinggi secara langsung untuk mendapatkan informasinya. Sehingga, hal ini sangat menyulitkan siswa sekolah SMA/K dalam mencari informasi Perguruan Tinggi yang diminatinya . Tentu saja hal ini sangat kurang efektif dan efisien dalam mencari informasi Perguruan Tinggi. Sedangkan untuk melakukan pencarian informasi lokasi Perguruan Tinggi, siswa sekolah SMA/SMK harus mencari secara manual baik melalui internet ataupun brosur yang didalamnya tercantum lokasi tempat Perguruan Tinggi.



Gambar III.5 Prosedur sistem yang berjalan



Gambar III.6 Prosedur yang berjalan 1

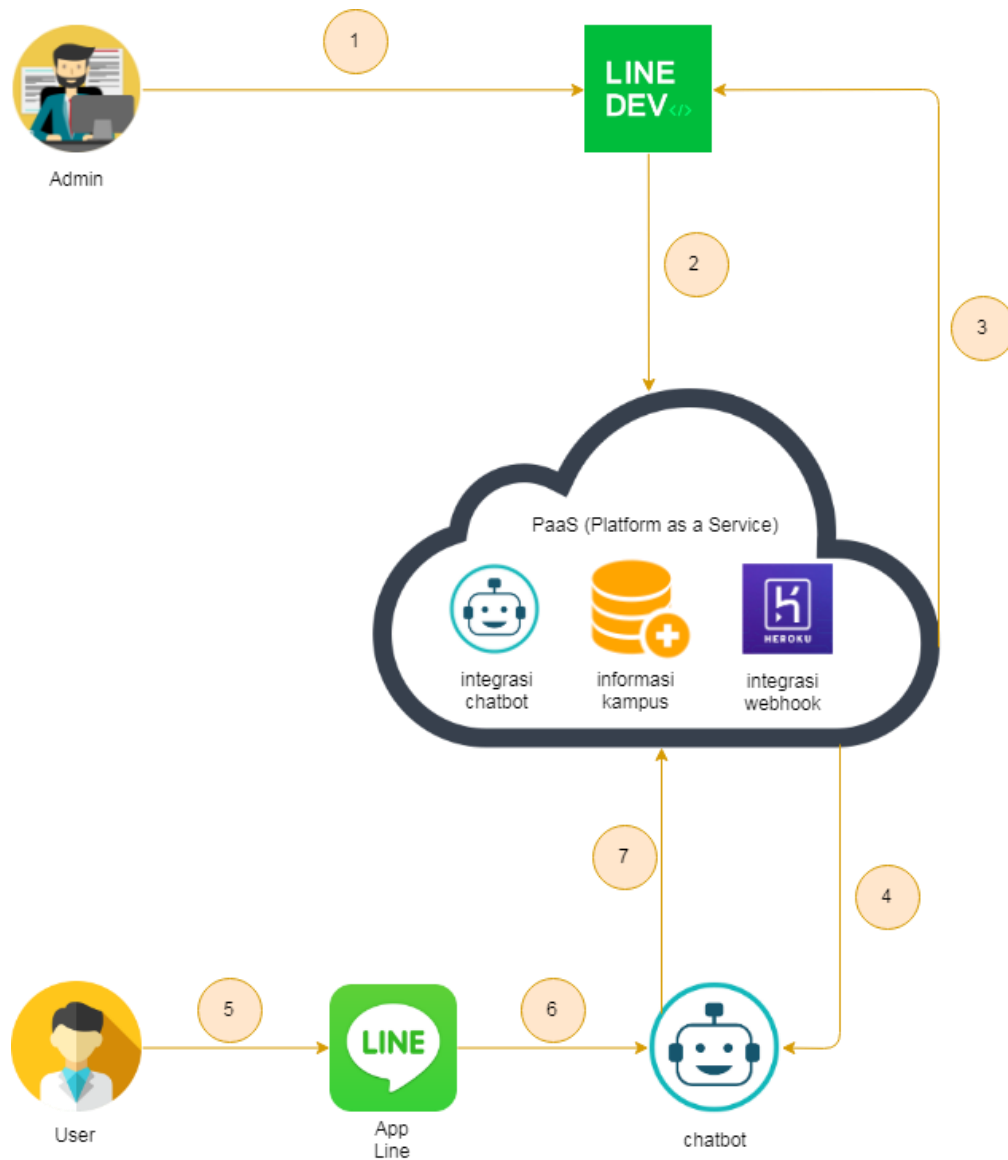
3.3 Analisis Solusi Yang Ditawarkan

Berdasarkan analisis terhadap masalah yang ada, maka diperlukan aplikasi untuk menyediakan pencarian informasi Perguruan Tinggi di Kota Bandung dengan menggunakan LINE Chatbot. Teknologi LINE Chatbot dipilih berdasarkan data kuisioner terhadap pengguna yang sering menggunakan layanan chat, serta karakteristik kemudahan penggunaan dan pembangunan . Adapun karakter chatbot yang dibangun adalah sebagai berikut :

1. Pengguna dapat berinteraksi dengan bot untuk melakukan pencarian informasi Perguruan Tinggi mengenai akreditasi, biaya perkuliahan, jurusan, dan lokasi kampus yang ada di Kota Bandung.
2. Pengguna dapat berinteraksi dengan bot dengan cara melakukan pencarian melalui menu yang sudah disiapkan, serta melakukan pencarian dengan mengetikkan pesan berupa pertanyaan kepada bot dalam bahasa Indonesia. Selain itu, pengguna dapat mencari lokasi tempat Perguruan Tinggi di Kota Bandung dengan cara mengirimkan lokasi tempat pengguna berada.

3.4 Analisis Deskripsi Aplikasi

Aplikasi yang dibangun berupa chatbot dimana pengguna dapat melakukan pencarian informasi dengan cara melalui menu yang sudah disediakan dan dengan cara mengetikkan pesan berupa pertanyaan. Adapun cara kerja aplikasi secara umum dapat dilihat pada gambar dibawah ini.



Gambar III.7 Analisis Deskripsi Aplikasi

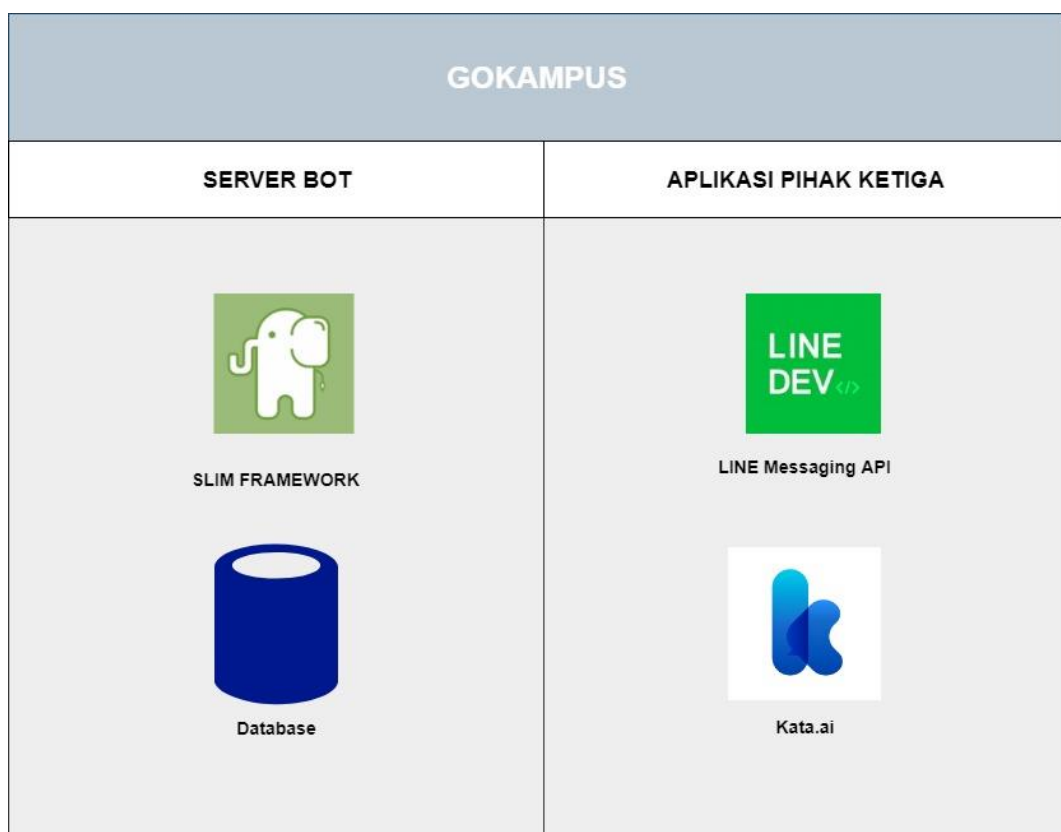
Secara umum aplikasi yang dibangun akan menggunakan platform Line sebagai media antara pengguna dengan aplikasi. Adapun deskripsi aplikasi adalah sebagai berikut :

1. Admin atau Developer melakukan pendaftaran atau login di website LINE Developers (<https://developers.line.biz>), setelah dapat mengakses selanjutnya Developer membuat chatbot LINE dan menyalin data.
2. Channel Secret dan Channel Access Token kemudian diintegrasikan dengan layanan Platform as a Service

3. Developer akan diberi subdomain webhook Heroku untuk dapat terintegrasi dengan Chatbot LINE, url subdomain kemudian di atur pada LINE Developers.
4. Setelah terintegrasi chatbot dapat digunakan pada aplikasi pesan instan LINE.
5. User membuka aplikasi LINE untuk dapat berkomunikasi dengan chatbot.
6. User melakukan komunikasi dengan chatbot untuk mencari informasi perguruan tinggi yang ada di Kota Bandung
7. User dapat melihat informasi perguruan tinggi sesuai percakapan kepada chatbot.

3.5 Analisis Arsitektur Aplikasi Yang Dibangun

Aplikasi terdiri dari dua bagian utama yakni Server Bot dan Aplikasi Pihak Ketiga (Line Messaging API dan Kata.AI). Gambar III.2 dibawah ini merupakan gambaran mengenai arsitektur aplikasi yang dibangun.



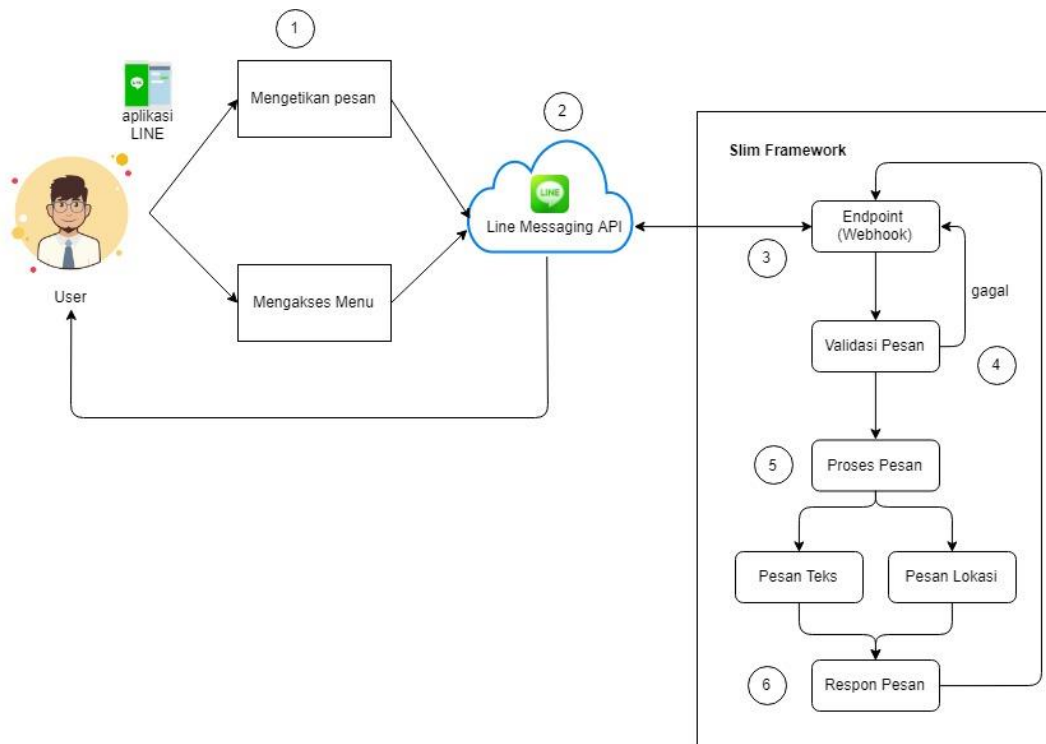
Gambar III.8 Analisis Arsitektur Aplikasi Yang Dibangun

Penjelasan untuk setiap bagian aplikasi adalah sebagai berikut:

1. Server Bot

a. Slim Framework

Slim framework digunakan untuk menangani request dan response yang dilakukan oleh user melalui Line Messaging API. User akan melakukan request berupa akses menu atau mengetik masukan berupa pesan teks pada aplikasi Line. Lalu request tersebut diteruskan melalui Line Messaging API untuk kemudian diteruskan kembali ke server bot dan kemudian diterima oleh Slim Framework melalui satu endpoint yang dinamakan dengan webhook yang sebelumnya sudah diatur pada Line Developer Console.



Gambar III.9 Alur Slim Framework

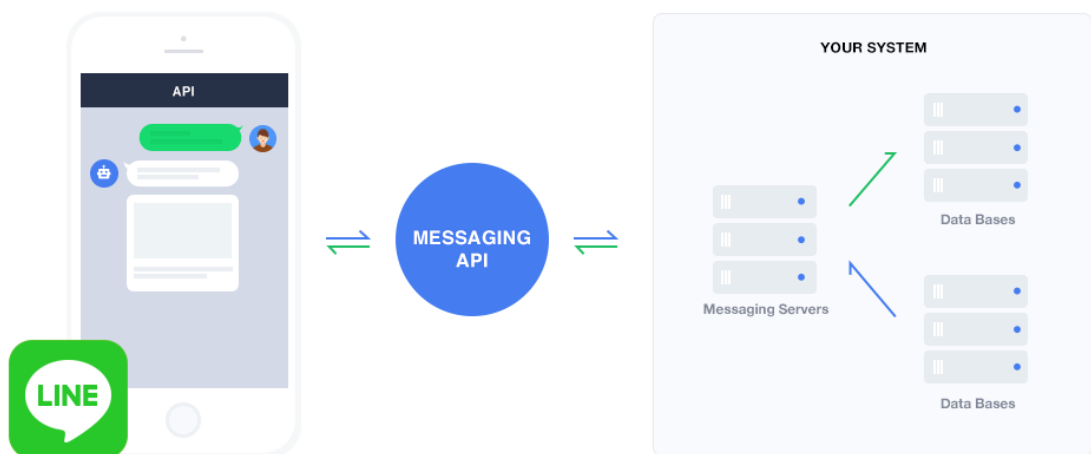
b. Database

Aplikasi membutuhkan akses terhadap data baik untuk melakukan pembacaan data ataupun penulisan data. Pada beberapa aksi yang perlu untuk melakukan pembacaan atau penulisan data, aplikasi akan melakukan akses ke database lalu kemudian data tersebut diolah pada Slim Framework untuk kemudian diteruskan kembali kepada user melalui Line Messaging API.

2. Aplikasi Pihak Ketiga

a. Line Messaging API

Line Messaging API digunakan untuk bisa terhubung dengan user melalui aplikasi Line. Untuk bisa terhubung melalui line messaging api, perlu dilakukan pengaturan webhook pada Line Developer Console dan pengaturan channel secret dan channel token pada aplikasi bot. Hal ini dilakukan untuk bisa mengatur setiap pesan yang dikirimkan oleh user akan diteruskan kepada bot yang sesuai.



Gambar III.10 Line Messaging API

b. Kata.ai

Google Distance Matrix API digunakan untuk bisa mengetahui lokasi terdekat antara lokasi pengguna dengan lokasi tempat Perguruan Tinggi. Pengguna akan mengirimkan lokasinya kepada bot, lokasi tersebut berupa lokasi saat ini berada ataupun lokasi yang dipilih oleh pengguna.

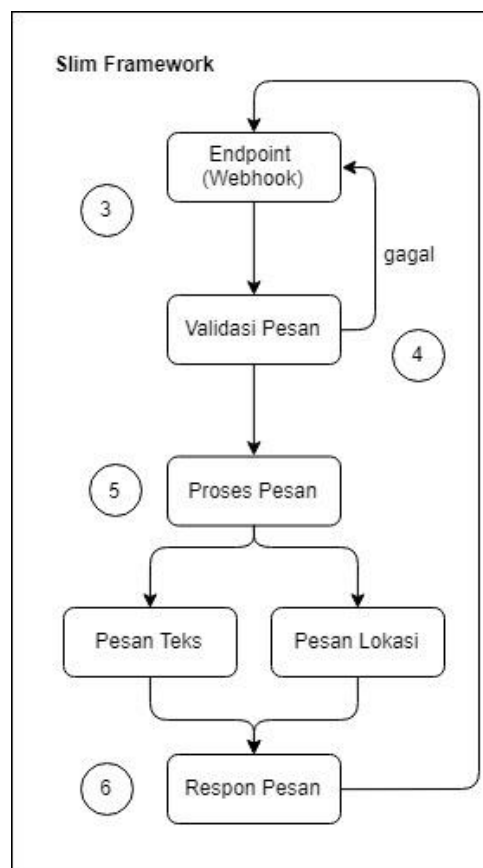
Bot kemudian akan memproses permintaan tersebut lalu melakukan request kepada Google Distance Matrix API untuk menghitung lokasi terdekat antara lokasi pengguna dengan seluruh data lokasi tempat perguruan tinggi yang sebelumnya sudah ada di database.

3.6 Analisis Teknologi Yang Digunakan

Dalam pembangunan aplikasi chatbot ini, penulis memanfaatkan teknologi Slim Framework, Deploy Heroku, aplikasi pihak ketiga yaitu kata.ai dan Line Messaging API.

1. Slim Framework

Slim framework digunakan untuk menangani request dan response yang dilakukan oleh user melalui Line Messaging API. Line messaging api akan mengirimkan request kepada aplikasi slim melalui route (webhook) yang sebelumnya sudah diatur.



III.11 Alur Penggunaan Slim Framework

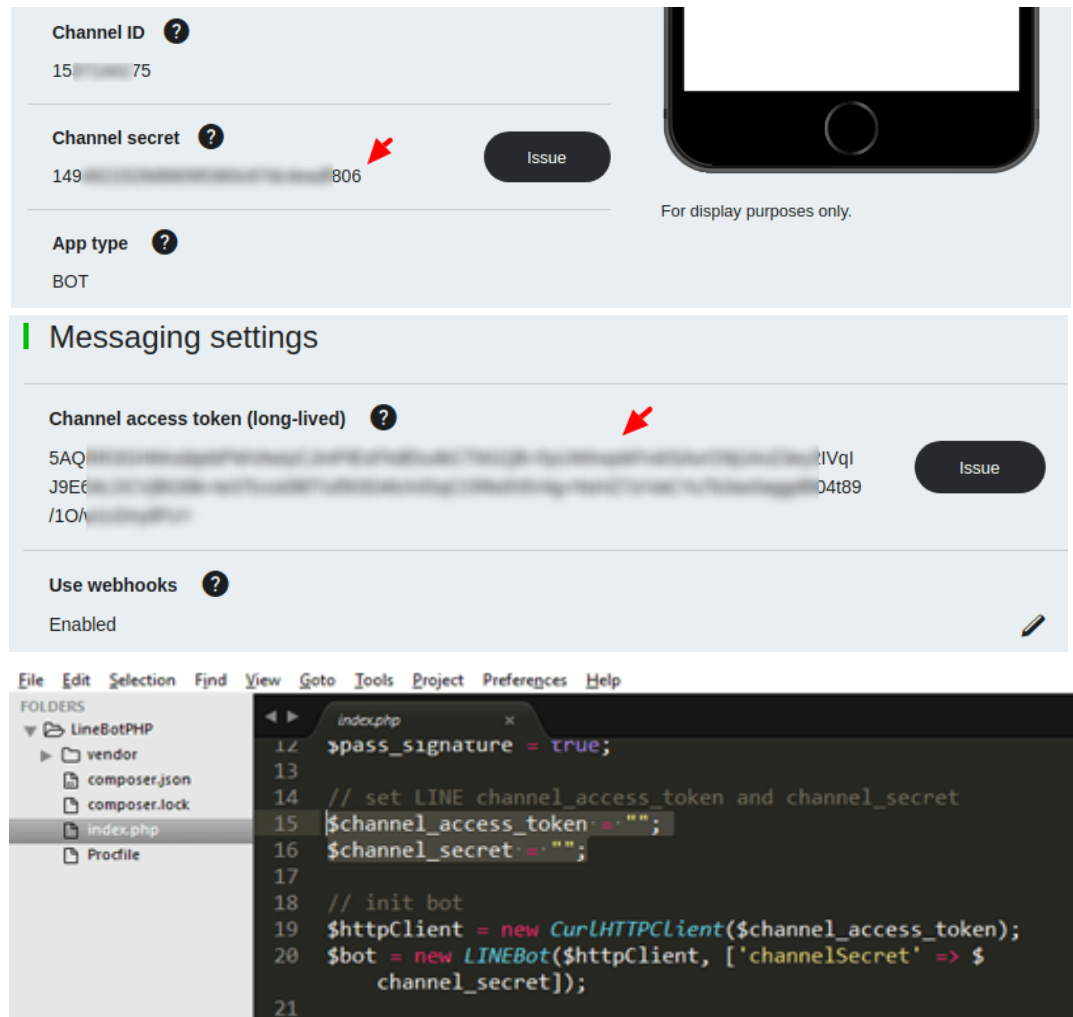
2. Deploy Heroku

Heroku adalah sebuah layanan *PaaS (Platform as a Service)* yang menyediakan *web service* bagi para *developer* aplikasi. Dengan menempatkan aplikasi mereka di ruang publik, diharapkan semua orang dapat mengaksesnya.

Setelah program sudah siap, maka selanjutnya mendeploy program ke heroku. Berikut adalah cara mendeploy program kita ke heroku.

```
1. $ heroku login // jika belum login
2. $ heroku create namaprojekanda // membuat project baru di heroku
3. $ git init
4. $ heroku git:remote -a namaprojekanda
5. $ git add .
6. $ git commit -m "initial commit"
7. $ git push heroku master
```

Setelah proses push berhasil, cek aplikasi yang dibuat di browser dengan alamat URL <https://namaprojekanda.herokuapp.com>. Langkah selanjutnya menintegrasikan Line Channel terhadap webhook yang sudah dibuat sebelumnya.



Gambar III.12 Integrasi LINE Channel API

3. Line Messaging API

Untuk dapat memanfaatkan fitur line messaging api, developer harus membuat channel messaging api terlebih dahulu. Channel ini merupakan representasi bot yang akan dibuat pada aplikasi Line. Untuk membuat channel ini, developer harus mempunyai akun Line Developer terlebih dahulu dengan cara mendaftar pada situs line developer. Gambar III.4 merupakan halaman masuk line developer.

LINE Business ID

Log in with LINE account

or

Log in with business account

[Create an account](#)

By logging in to LINE Business ID, you agree to the [Terms of Use](#).

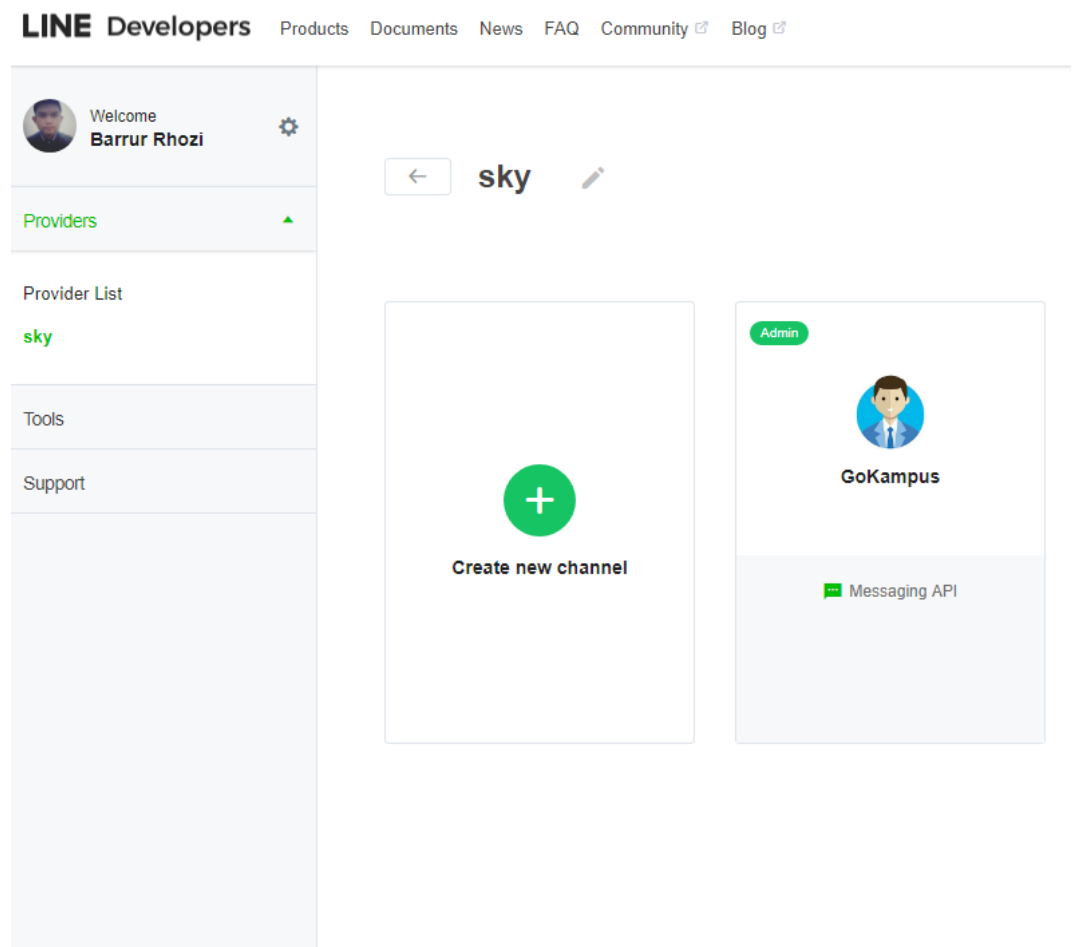
[? About LINE Business ID](#)

English ▼

[Help](#) [Terms of Use](#) © LINE Corporation

Gambar III.13 Halaman masuk line developer

Setelah masuk, developer dapat membuat channel messaging api untuk membuat bot. Gambar III.14 merupakan channel line messaging api.



Gambar III.14 Channel Line Messaging API

Untuk dapat terhubung ke server bot, developer harus mengatur webhook (endpoint) pada line developer. Setiap request yang dikirimkan melalui bot akan dikirimkan ke alamat tersebut. Gambar III.7 menunjukkan pengaturan webhook pada situs line developer.

Messaging settings

Channel access token (long-lived) ⓘ

IsZGQ7C12z25uUQbarFI2sR6UpWbB+K7doX+Ms3vjov6S.JnZcLtsS8TVpsrTh04/rfSyJp.JjzzcvLZU1W593FXptA0ntu5sPFU3geXYKpBZtq1JrUeMVMCPkZ5AXoURnehMa+01OglUNLTBUeWAdB04t89/1O/w1cDnyilFU=

Issue

Use webhooks ⓘ

Enabled

Edit

Webhook URL Requires SSL ⓘ

—

Edit

Allow bot to join group chats ⓘ

Disabled

Edit

Using LINE@ features

Message text for LINE@ features are set on the LINE@ Manager.

Auto-reply messages ⓘ

Disabled

Edit

Greeting messages ⓘ

Disabled

Edit


Gambar III.15 Pengaturan Webhook pada Line Developer

Developer harus mengatur token dan channel pada aplikasi bot untuk dapat mengembalikan respon berupa pesan kepada user. Gambar III.8 menunjukkan pengaturan umum yang dapat diatur oleh developer.

Basic information

App icon and App name can be changed at LINE official account manager. [Change informations](#) ⓘ

App icon



App name

GoKampus

App description

aplikasi ini berguna bagi siswa SMA/K yang akan melanjutkan sekolah ke jenjang yang lebih tinggi untuk mencari informasi Perguruan Tinggi yang diminatinya.

Channel ID ⓘ

1649189666

Channel secret ⓘ

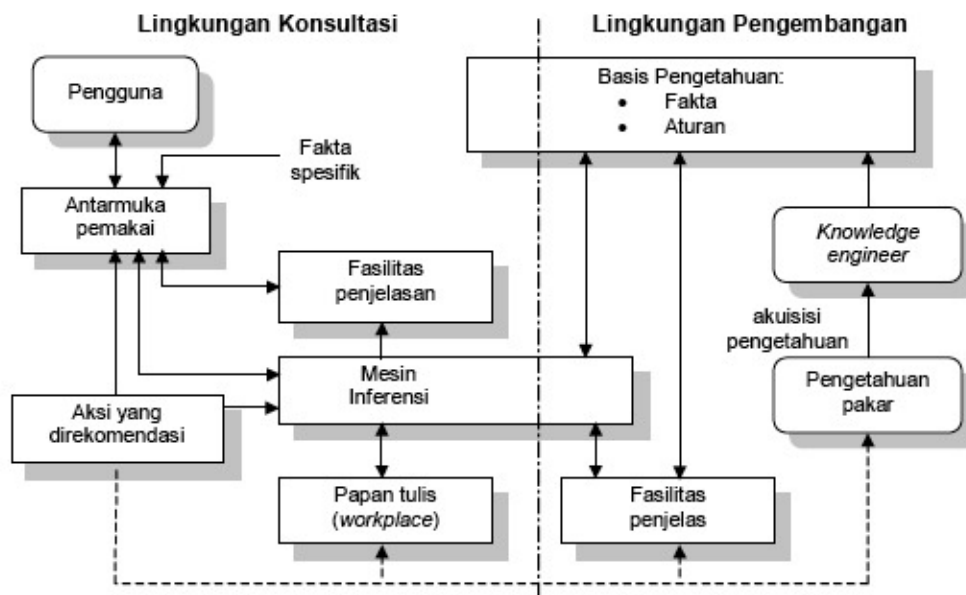
8d5e13dab45263b6dc6c7c53362a29a1

Gambar III.16 Pengaturan umum bot pada line developer

3.7 Analisis Metode

Analisis metode adalah analisis metode-metode yang akan digunakan. Penelitian ini dibangun menggunakan sistem pakar dengan metode yang digunakan forward chaining .

3.7.1 Diagram Struktur Sistem Pakar



Struktur sistem pakar bisa dijelaskan seperti berikut :

1. **Basis pengetahuan :** Berupa pengetahuan-pengetahuan yang dibutuhkan untuk memahami, memformulasi, dan memecahkan masalah. Diaman basis pengetahuan tersusun atas 2 elemen dasar:
 - **Fakta**, misalnya: situasi, kondisi, dan kenyataan dari permasalahan yang ada, serta teori dalam bidang itu
 - **Aturan**, yang mengarahkan penggunaan pengetahuan untuk memecahkan masalah yang spesifik dalam bidang yang khusus

Mengembangkan Sistem Pakar dapat dilakukan dengan 2 cara:

- a. Membangun sendiri semua komponen di atas, atau
- b. Memakai semua komponen yang sudah ada kecuali isi basis pengetahuan.

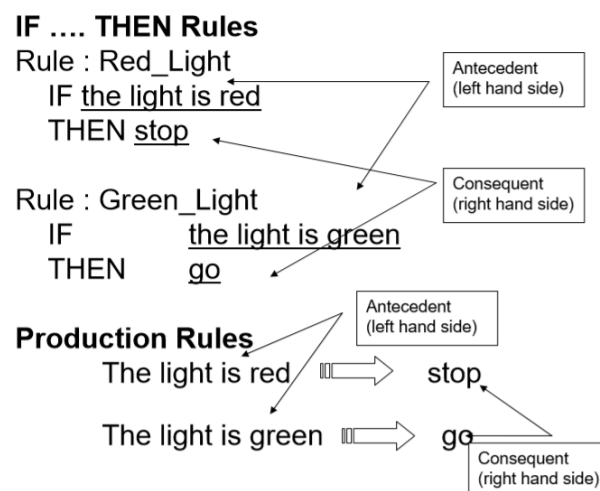
Tahap-tahap pembangunnan yaitu:

1. Pemilihan Masalah
2. Rekayasa Pengetahuan (Knowledge Engineering)
3. Partisipasi Dalam Proses Pengembangan
4. Akuisisi Pengetahuan

Inferensi digunakan dalam sistem pakar untuk memperoleh informasi terbaru dari informasi yang sudah ada yaitu salah satunya forward chaining.

3.7.2 Forward Chaining

Forward chaining disebut juga penalaran dari bawah ke atas karena penalaran dari fakta pada level bawah menuju konklusi pada level atas didasarkan pada fakta. Penalaran dari bawah ke atas dalam suatu sistem pakar dapat disamakan untuk pemrograman konvensional dari bawah ke atas. Fakta merupakan satuan dasar dari paradigma berbasis pengetahuan karena mereka tidak dapat diuraikan ke dalam satuan paling kecil yang mempunyai makna.



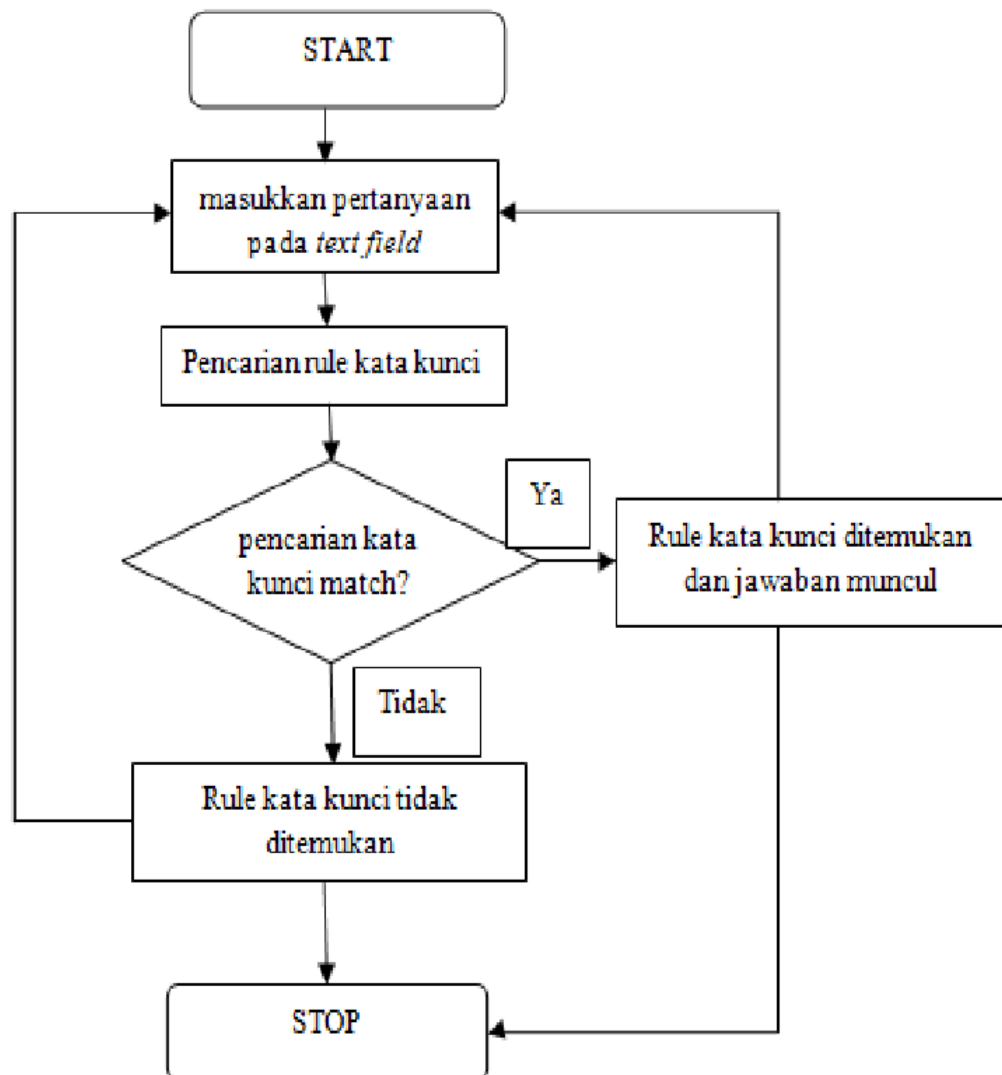
3.7.3 Algoritma Forward Chaining

Untuk mempresentasikan aplikasi berikut kedalam algoritma , dapat dituliskan menggunakan metode JIKA MAKA (IF THEN), berikut role algoritmanya :

Contoh : Cari Kampus

- Rule 1
Jika pengguna memilih menu cari kampus, **Maka** bot akan menampilkan pilihan kampus berdasarkan akreditasi, dan status.
- Rule 2
Jika pengguna memilih pilihan berdasarkan akreditasi, **Maka** bot akan menampilkan pilihan akreditasi A,B, dan C.
- Rule 3
Jika pengguna memilih pilihan berdasarkan status, **Maka** bot akan menampilkan pilihan status kampus Negeri atau Swasta.
- Rule 4
Jika pengguna memilih pilihan akreditasi A,B, atau C, **Maka** bot akan menampilkan informasi kampus berdasarkan akreditasi
- Rule 5
Jika pengguna memilih pilihan status kampus Negeri atau Swasta, **Maka** bot akan menampilkan informasi kampus Negeri atau Swasta yang dipilih.

3.7.4 Proses Forward Chaining



Gambar III.17 Proses Forward Chaining

Proses forward chaining dimulai dengan user memasukkan informasi atau pertanyaan ke dalam text field yang sudah disediakan oleh aplikasi. Kemudian pertanyaan tersebut dipisah menjadi beberapa kata dan dicari kata kunci yang sesuai dengan yang ada pada aplikasi setelah kata kunci ditemukan maka aplikasi menjawab pertanyaan user dengan jawaban yang benar sedangkan jika aplikasi tidak dapat menemukan kata kunci maka jawaban muncul tetapi tidak berhubungan dengan pertanyaan dari user.

Berikut adalah tabel hasil implementasi algoritma sistem pakar pada aplikasi chatbot.

Tabel. III.1 Implementasi implementasi algoritma sistem pakar

No	Masukan Pertanyaan	Kata Kunci (Keyword)	Hasil
1.	Tolong carikan kampus negeri di Bandung	Cari kampus Negeri	Menampilkan list kampus negeri
2.	Tolong carikan alamat kampus di Bandung	Cari lokasi kampus	Menampilkan list lokasi kampus
3.	Carikan akreditasi kampus	Cari akreditasi kampus	Menampilkan akreditasi kampus
4.	Carikan akreditasi jurusan	Cari akreditasi jurusan	Menampilkan akreditasi jurusan
5.	Cari biaya jurusan teknik informatika	Biaya jurusan teknik informatika	Menampilkan kampus berdasarkan akreditasi jurusan

3.8 Analisis Spesifikasi Kebutuhan Perangkat Lunak

Berdasarkan hasil analisis masalah, analisis sistem aplikasi dan analisis arsitektur sistem didapatkan spesifikasi kebutuhan perangkat lunak yang dibangun dibagi menjadi dua kebutuhan yaitu spesifikasi kebutuhan nonfungsional dan spesifikasi kebutuhan fungsional.

3.8.1 Analisis Kebutuhan Non Fungsional

Analisis kebutuhan non fungsional yang dilakukan meliputi analisis perangkat keras, analisis perangkat lunak. Analisis yang dilakukan yakni sebagai berikut :

1. Analisis Perangkat Keras

Perangkat keras minimum yang harus diperlukan untuk mendukung kinerja aplikasi dapat dilihat pada tabel berikut :

Tabel III.2 Analisis Perangkat Keras

No	Perangkat	Spesifikasi
1	Prosesor	1 CPU
2	Hardisk	16 GB
3	RAM	1 GB

2. Analisis Perangkat Lunak

Berikut adalah spesifikasi perangkat lunak yang digunakan dalam membangun dan mengimplementasikan aplikasi chatbot sebagai berikut :

a. User (Client)

1. Aplikasi Pesan Instan (LINE v9.2.2)

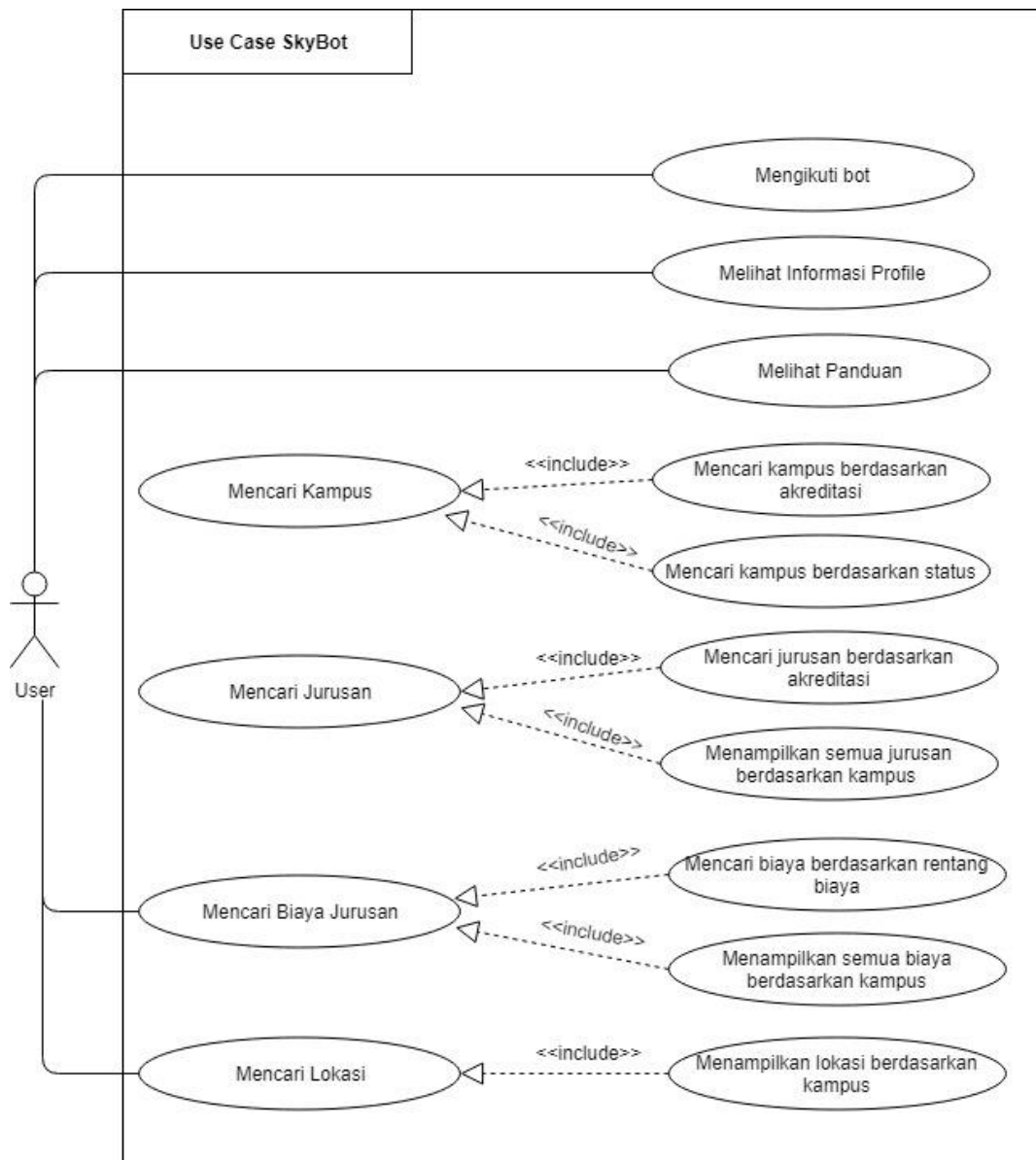
b. Bot (Server)

1. Sistem Operasi Windows (7,8,10)
2. Database Server (MySQL 5.6.1)
3. Slim Framework
4. PHP (PHP v7.1.3)
5. Composer

3.8.2 Analisis Kebutuhan Fungsional

Identifikasi aktor dapat dilakukan dalam analisis berorientasi objek dengan menggunakan UML yaitu menentukan aktor atau user sistem. Aktor dalam konteks UML menampilkan peran pemain atau sesuatu diluar sistem yang dikembangkan dapat berupa perangkat keras, user, sistem yang lain dan sebagainya. Diagram yang akan digambarkan pada bagian ini adalah use case diagram, class diagram, dan sequence diagram.

1. Use Case Diagram



Gambar III.17 Use Case Diagram

Definisi aktor dari pembangunan perangkat lunak ini dapat dilihat pada tabel berikut:

Tabel III.3 Definisi Actor

No	Aktor	Deskripsi
1	User	<i>User</i> hanya dapat mengakses fitur yang tersedia

1.2 Definisi Use Case

Definisi *Use case* dari pembangunan perangkat lunak ini dapat dilihat pada tabel berikut:

Tabel III.4 Definisi Use Case

No.	Use Case	Deskripsi
1	Mengikuti Bot	Mengikuti bot di aplikasi LINE
2	Melihat Informasi Profile	Melihat informasi profile pengguna LINE
3	Melihat Panduan	Melihat panduan pencarian informasi kampus
4	Mencari kampus berdasarkan akreditasi	Menampilkan kampus berdasarkan akreditasi
5	Mencari kampus berdasarkan status	Menampilkan kampus berdasarkan status Negeri atau Swasta
6	Mencari jurusan berdasarkan akreditasi	Menampilkan jurusan berdasarkan akreditasi
7	Menampilkan semua jurusan berdasarkan kampus	Menampilkan semua jurusan berdasarkan kampus yang dipilih
8	Mencari biaya berdasarkan rentang harga	Menampilkan biaya semester jurusan berdasarkan rentang harga
9	Menampilkan semua biaya berdasarkan kampus	Menampilkan semua biaya semester jurusan berdasarkan kampus yang dipilih
10	Menampilkan lokasi berdasarkan kampus	Menampilkan lokasi berdasarkan kampus yang dipilih

1.3 Use Case Description

Keterangan lebih lengkap mengenai masing-masing usecase yang terdapat pada use case diagram ditulis dalam use case description yang terdapat pada Tabel.

Tabel III.5 Use Case Description Mengikuti Bot

Use Case Name	Mengikuti Bot	
Related Requirements		
Goal in Context	Mengikuti akun bot	
Preconditions	User sudah mempunyai dan masuk ke dalam aplikasi Line	
Successful end Condition	User berhasil mengikuti akun bot	
Failed end Condition	User gagal mengikuti akun bot	
Primary Actors	User	
Secondary Actors		
Trigger	User meminta mengikuti bot	
Main Flow	Step	Action
	1	User meminta untuk mengikuti bot
	2	Bot memeriksa apakah pengguna sudah mengikuti bot
	3	User berhasil mengikuti bot
Extensions	Step	Branching Action
	2.1	Bot gagal memeriksa data pengguna
	2.2	User gagal mengikuti bot

Tabel III.6 Use Case Description Melihat Informasi Profile

Use Case Name	Melihat Informasi Profile	
Related Requirements		
Goal in Context	Melihat informasi profile pengguna LINE	
Preconditions	User sudah mengikuti bot	
Successful end Condition	User berhasil melihat informasi profile pengguna LINE	
Failed end Condition	User gagal melihat informasi profile pengguna LINE	
Primary Actors	User	
Secondary Actors		
Trigger	User meminta informasi profile pengguna LINE	
Main Flow	Step	Action
	1	User meminta untuk melihat informasi profile pengguna LINE
	2	Bot memeriksa apakah pengguna sudah mengikuti bot
	3	User berhasil melihat informasi profile pengguna LINE
Extensions	Step	Branching Action
	2.1	Bot gagal memeriksa informasi profile
	2.2	User gagal melihat informasi profile

Tabel III.7 Use Case Description Melihat Panduan

Use Case Name	Melihat Panduan	
Related Requirements		
Goal in Context	Melihat Panduan aplikasi	
Preconditions	User sudah mengikuti bot	
Successful end Condition	User berhasil melihat panduan aplikasi	
Failed end Condition	User gagal melihat panduan aplikasi	
Primary Actors	User	
Secondary Actors		
Trigger	User meminta melihat panduan aplikasi	
Main Flow	Step	Action
	1	User meminta untuk melihat panduan aplikasi
	2	Bot memeriksa apakah pengguna sudah mengikuti bot
	3	User berhasil melihat panduan aplikasi
Extensions	Step	Branching Action
	2.1	Bot gagal memeriksa panduan aplikasi
	2.2	User gagal melihat panduan aplikasi

Tabel III.8 Use Case Description Mencari Kampus berdasarkan Akreditasi

Use Case Name	Mencari kampus berdasarkan akreditasi	
Related Requirements		
Goal in Context	Mencari kampus berdasarkan akreditasi di Kota Bandung	
Preconditions	User sudah mengikuti bot	
Successful end Condition	User berhasil mencari kampus berdasarkan akreditasi di Kota Bandung	
Failed end Condition	User gagal mencari kampus berdasarkan akreditasi di Kota Bandung	
Primary Actors	User	
Secondary Actors		
Trigger	User meminta mencari kampus berdasarkan akreditasi di Kota Bandung	
Main Flow	Step	Action
	1	User meminta untuk mencari kampus berdasarkan akreditasi di Kota Bandung
	2	Bot memeriksa apakah pengguna sudah mengikuti bo
	3	User berhasil mencari kampus berdasarkan akreditasi di Kota Bandung
Extensions	Step	Branching Action
	2.1	Bot gagal memeriksa informasi kampus berdasarkan akreditasi
	2.2	User gagal melihat informasi kampus berdasarkan akreditasi

Tabel III.9 Use Case Description Mencari Info Kampus berdasarkan status

Use Case Name	Mencari info kampus berdasarkan status	
Related Requirements		
Goal in Context	Mencari informasi kampus berdasarkan status	
Preconditions	User sudah mengikuti bot	
Successful end Condition	User berhasil mencari informasi kampus berdasarkan status	
Failed end Condition	User gagal mencari informasi kampus berdasarkan status	
Primary Actors	User	
Secondary Actors		
Trigger	User meminta mencari informasi kampus berdasarkan status	
Main Flow	Step	Action

	1	User meminta untuk mencari informasi kampus berdasarkan status
	2	Bot memeriksa informasi status yang diakses user
	3	User berhasil melihat informasi status
Extensions	Step	Branching Action
	2.1	Bot gagal memeriksa informasi status yang diminta
	2.2	User gagal mencari informasi status kampus

Tabel III.10 Use Case Description Mencari Informasi Jurusan Berdasarkan Akreditasi

Use Case Name	Mencari info jurusan berdasarkan akreditasi	
Related Requirements		
Goal in Context	Mencari informasi jurusan berdasarkan akreditasi	
Preconditions	User sudah mengikuti bot	
Successful end Condition	User berhasil mencari informasi jurusan berdasarkan akreditasi	
Failed end Condition	User gagal mencari informasi jurusan berdasarkan akreditasi	
Primary Actors	User	
Secondary Actors		
Trigger	User meminta mencari informasi jurusan berdasarkan akreditasi	
Main Flow	Step	Action
	1	User meminta untuk mencari informasi jurusan berdasarkan akreditasi
	2	Bot memeriksa informasi akreditasi jurusan yang diakses user
	3	User berhasil melihat informasi akreditasi jurusan
Extensions	Step	Branching Action
	2.1	Bot gagal memeriksa informasi akreditasi jurusan yang diminta
	2.2	User gagal mencari informasi akreditasi jurusan

Tabel III.11 Use Case Description menampilkan Info semua jurusan Berdasarkan kampus

Use Case Name	Menampilkan info semua jurusan berdasarkan kampus	
Related Requirements		
Goal in Context	Menampilkan informasi semua jurusan berdasarkan kampus	
Preconditions	User sudah mengikuti bot	
Successful end Condition	User berhasil menampilkan informasi semua jurusan berdasarkan kampus	
Failed end Condition	User gagal menampilkan informasi semua jurusan berdasarkan kampus	
Primary Actors	User	
Secondary Actors		
Trigger	User meminta menampilkan informasi semua jurusan berdasarkan kampus	
Main Flow	Step	Action
	1	User meminta untuk menampilkan informasi semua jurusan berdasarkan kampus
	2	Bot memeriksa informasi semua jurusan yang diakses user
	3	User berhasil melihat informasi semua jurusan berdasarkan kampus dipilih
Extensions	Step	Branching Action
	2.1	Bot gagal memeriksa informasi semua jurusan yang diminta
	2.2	User gagal mencari informasi semua jurusan berdasarkan kampus dipilih

Tabel III.12 UseCase Description Mencari Info Biaya Berdasarkan rentang harga

Use Case Name	Mencari info biaya jurusan berdasarkan rentang harga
Related Requirements	
Goal in Context	Mencari informasi biaya jurusan berdasarkan rentang harga
Preconditions	User sudah mengikuti bot
Successful end Condition	User berhasil mencari informasi biaya jurusan berdasarkan rentang harga
Failed end Condition	User gagal mencari informasi semua jurusan berdasarkan rentang harga

Primary Actors	User	
Secondary Actors		
Trigger	User meminta mencari informasi biaya jurusan berdasarkan rentang harga	
Main Flow	Step	Action
	1	User meminta untuk mencari informasi biaya jurusan berdasarkan rentang harga
	2	Bot memeriksa informasi biaya jurusan yang diakses user
	3	User berhasil melihat informasi biaya jurusan berdasarkan rentang harga
Extensions	Step	Branching Action
	2.1	Bot gagal memeriksa informasi biaya jurusan yang diminta
	2.2	User gagal mencari informasi biaya jurusan berdasarkan rentang harga

Tabel III.13 Use Case Description menampilkan Info semua biaya Berdasarkan kampus

Use Case Name	Menampilkan info semua biaya berdasarkan kampus	
Related Requirements		
Goal in Context	Menampilkan informasi semua biaya berdasarkan kampus	
Preconditions	User sudah mengikuti bot	
Successful end Condition	User berhasil menampilkan informasi semua biaya berdasarkan kampus	
Failed end Condition	User gagal menampilkan informasi semua biaya berdasarkan kampus	
Primary Actors	User	
Secondary Actors		
Trigger	User meminta menampilkan informasi semua biaya berdasarkan kampus	
Main Flow	Step	Action
	1	User meminta untuk menampilkan informasi semua biaya berdasarkan kampus
	2	Bot memeriksa informasi semua biaya yang diakses user
	3	User berhasil melihat informasi semua biaya berdasarkan kampus dipilih

Extensions	Step	Branching Action
	2.1	Bot gagal memeriksa informasi semua biaya yang diminta
	2.2	User gagal mencarti informasi semua biaya berdasarkan kampus dipilih

Tabel III.14 Use Case Description menampilkan Info lokasi Berdasarkan kampus

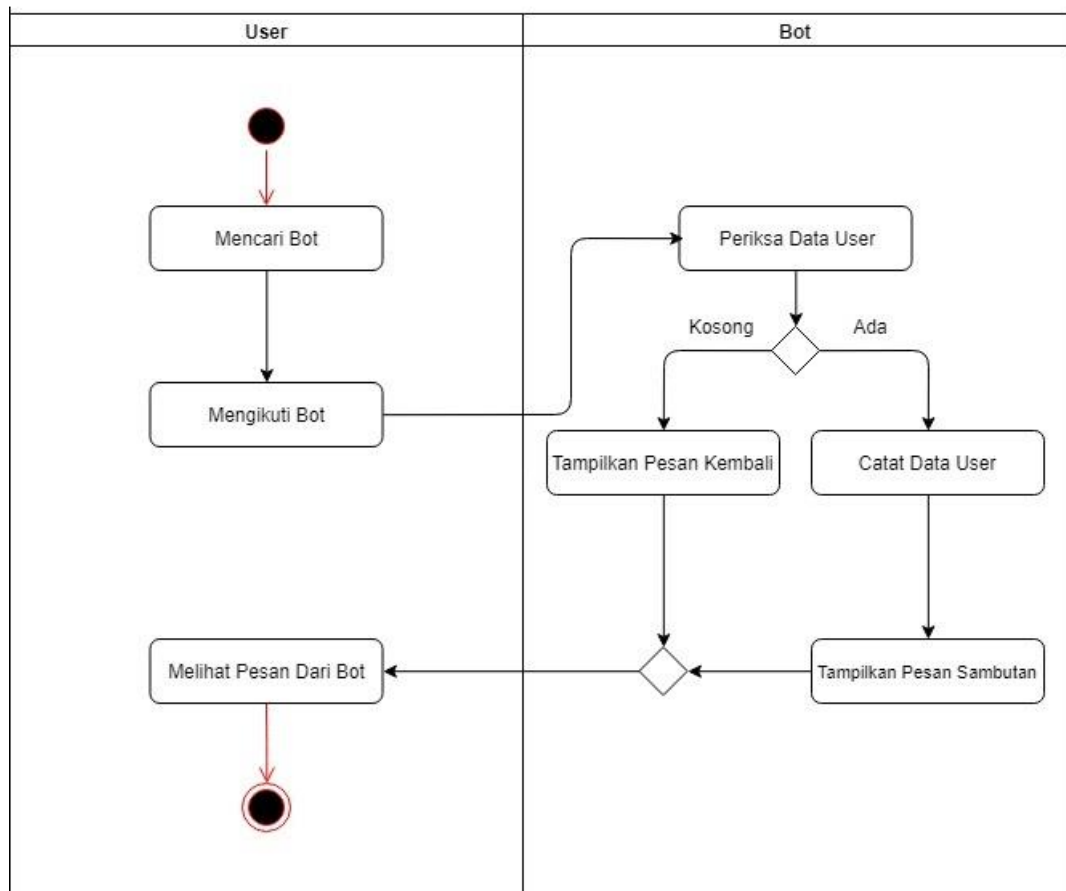
Use Case Name	Menampilkan info lokasi berdasarkan kampus	
Related Requirements		
Goal in Context	Menampilkan informasi lokasi berdasarkan kampus	
Preconditions	User sudah mengikuti bot	
Successful end Condition	User berhasil menampilkan informasi lokasi berdasarkan kampus	
Failed end Condition	User gagal menampilkan informasi lokasi berdasarkan kampus	
Primary Actors	User	
Secondary Actors		
Trigger	User meminta menampilkan informasi lokasi berdasarkan kampus	
Main Flow	Step	Action
	1	User meminta untuk menampilkan informasi lokasi berdasarkan kampus
	2	Bot memeriksa informasi lokasi yang diakses user
	3	User berhasil melihat informasi lokasi berdasarkan kampus dipilih
Extensions	Step	Branching Action
	2.1	Bot gagal memeriksa informasi lokasi yang diminta
	2.2	User gagal mencarti informasi lokasi berdasarkan kampus dipilih

2. Activity Diagram

Activity Diagram adalah diagram yang menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Aktivitas pengguna di dalam aplikasi pada setiap use case dimodelkan melalui activity diagram .

2.1 Activity Diagram Mengikuti Bot

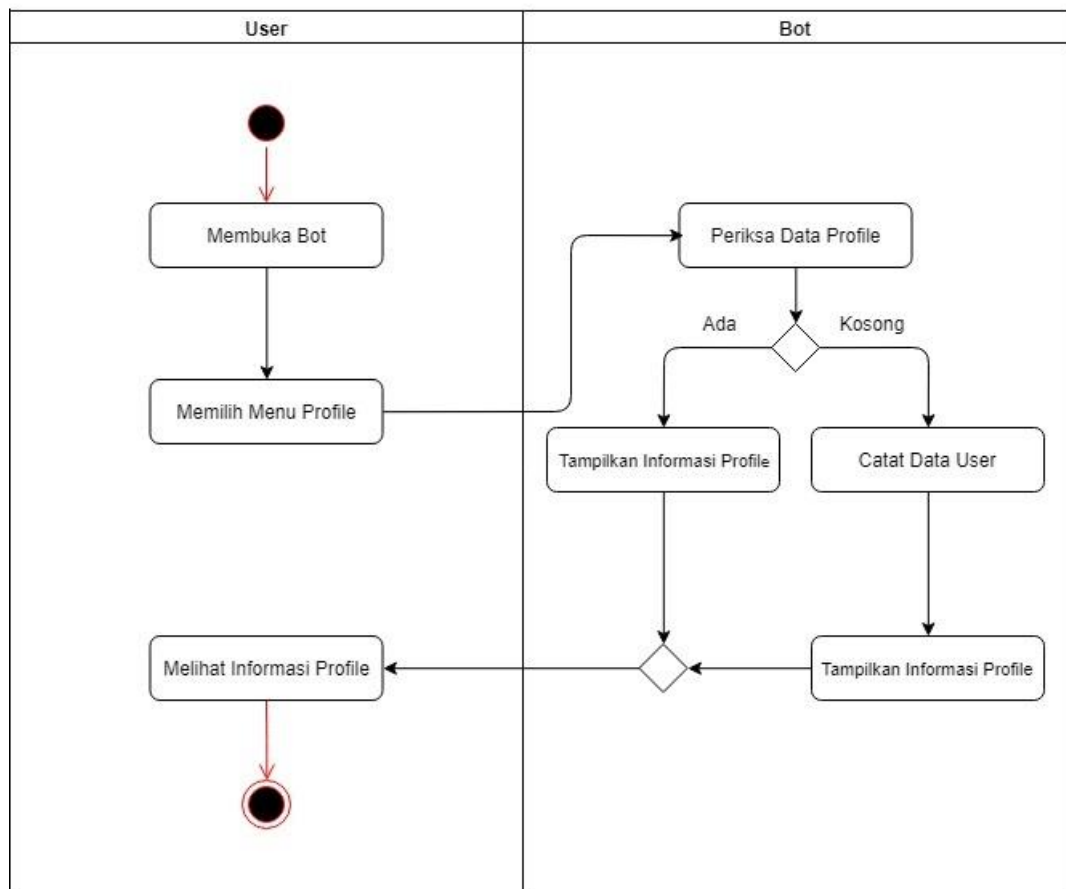
Activity diagram Mengikuti Bot menggambarkan kegiatan-kegiatan yang terjadi pada saat pengguna mengikuti bot. Gambar III.10 menunjukkan activity diagram mengikuti bot :



Gambar III.18 Mengikuti Bot

2.2 Activity Diagram Melihat Informasi Profile

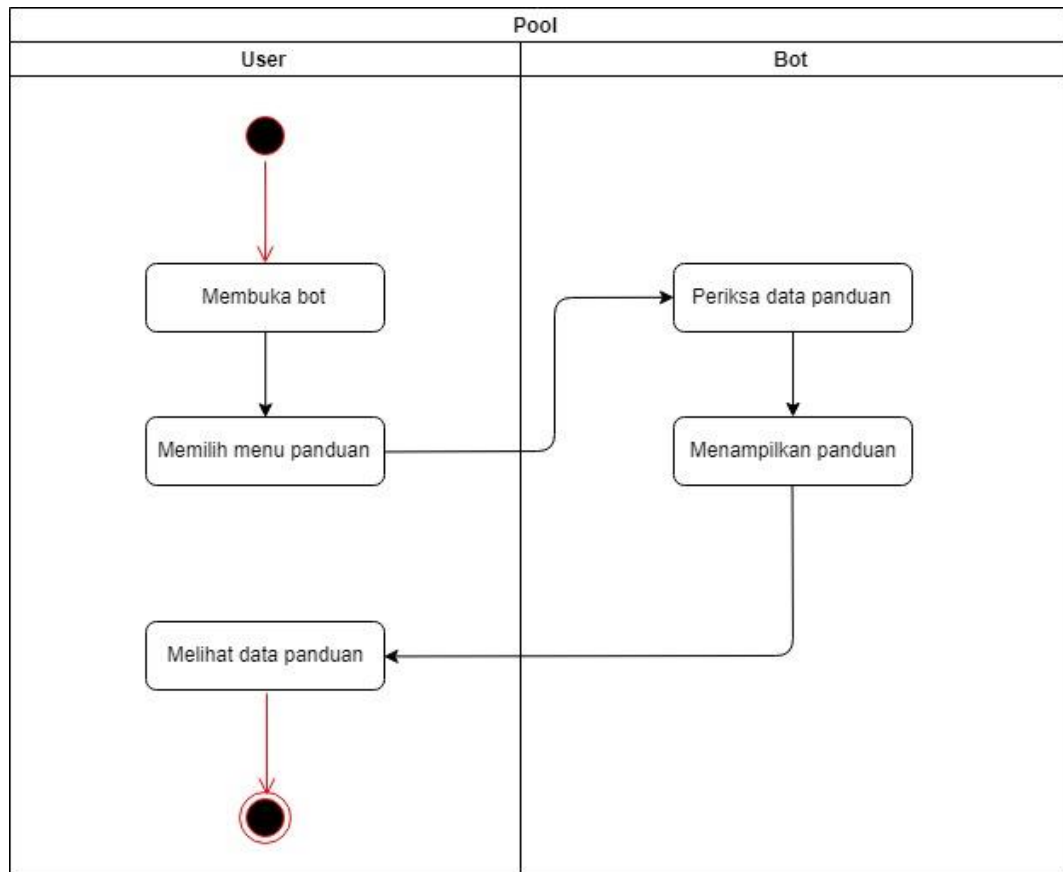
Activity diagram Melihat Informasi Profil menggambarkan kegiatan-kegiatan yang terjadi pada saat pengguna melihat informasi profil. Gambar III.11 menunjukkan activity diagram melihat informasi profil:



Gambar III.19 Activity Diagram Melihat informasi profil

2.3 Activity Diagram Melihat Panduan

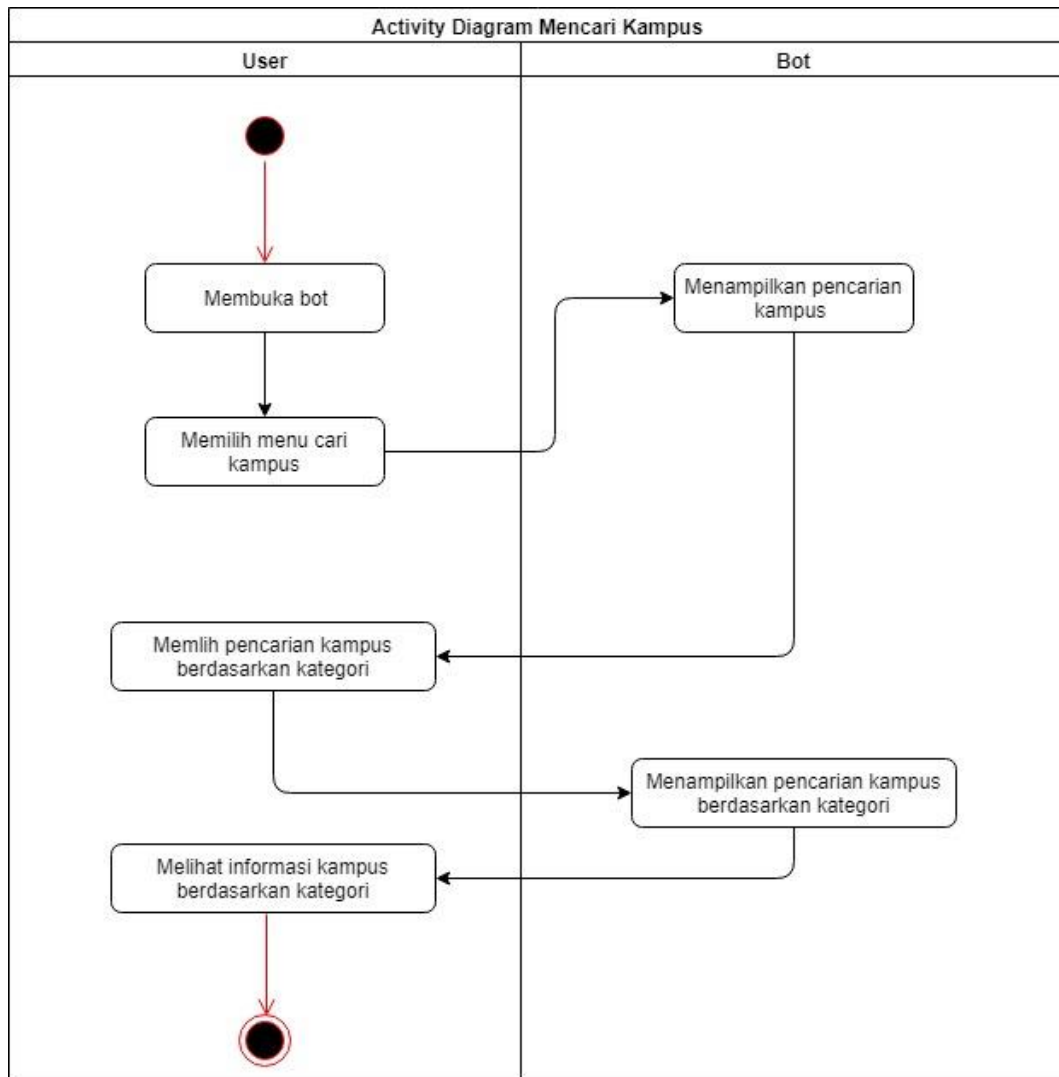
Activity diagram Melihat Stiker menggambarkan kegiatan-kegiatan yang terjadi pada saat pengguna melihat panduan. Gambar III.12 menunjukkan activity diagram melihat informasi panduan :



Gambar III.20 Activity Diagram Menu Melihat Panduan

2.4 Activity Diagram Menu Mencari Kampus

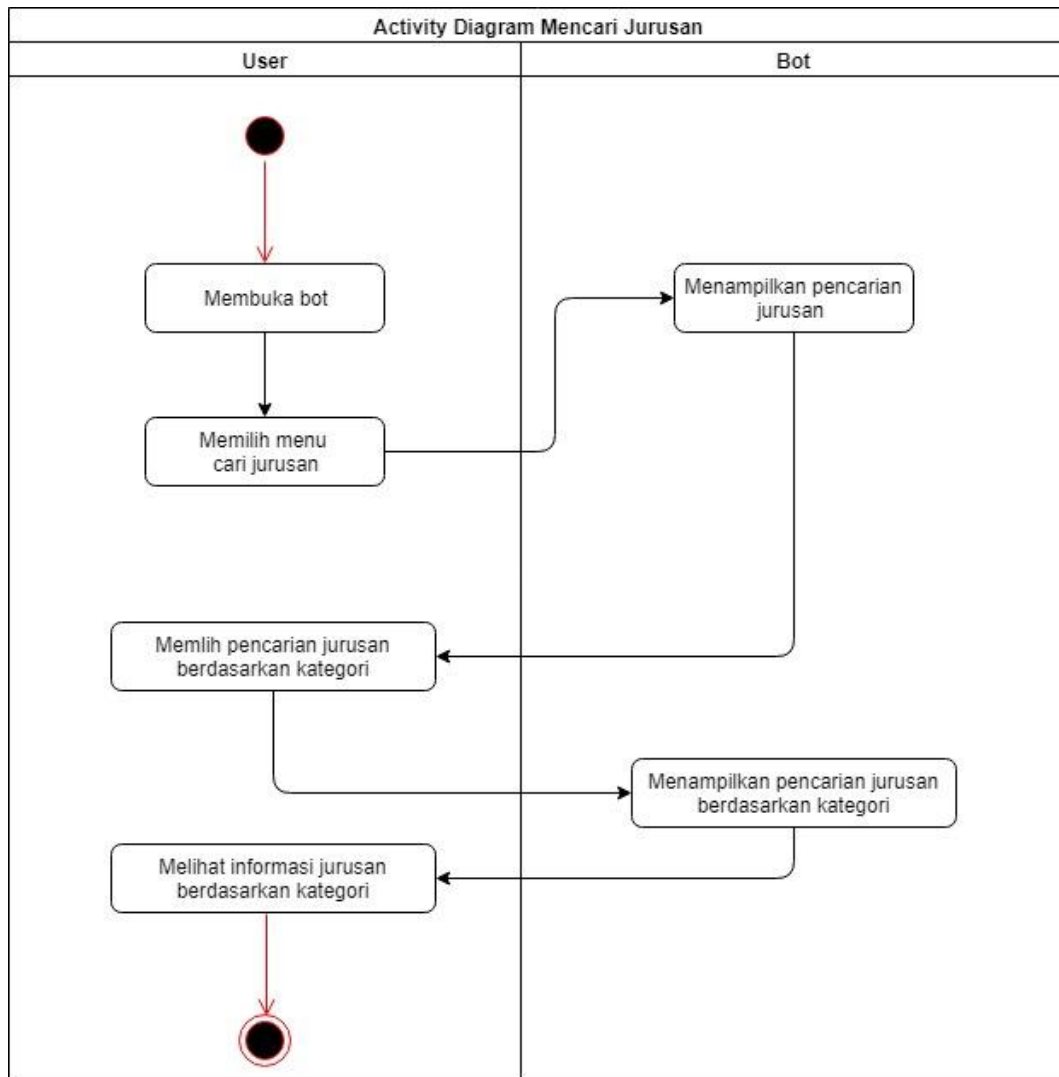
Activity diagram mencari kampus di Kota Bandung menggambarkan kegiatan-kegiatan yang terjadi pada saat pengguna mencari kampus di Kota Bandung. Gambar III.13 menunjukkan activity diagram mencari kampus di Kota Bandung:



Gambar III.21 Activity diagram menu cari kampus

2.5 Activity Diagram menu cari jurusan

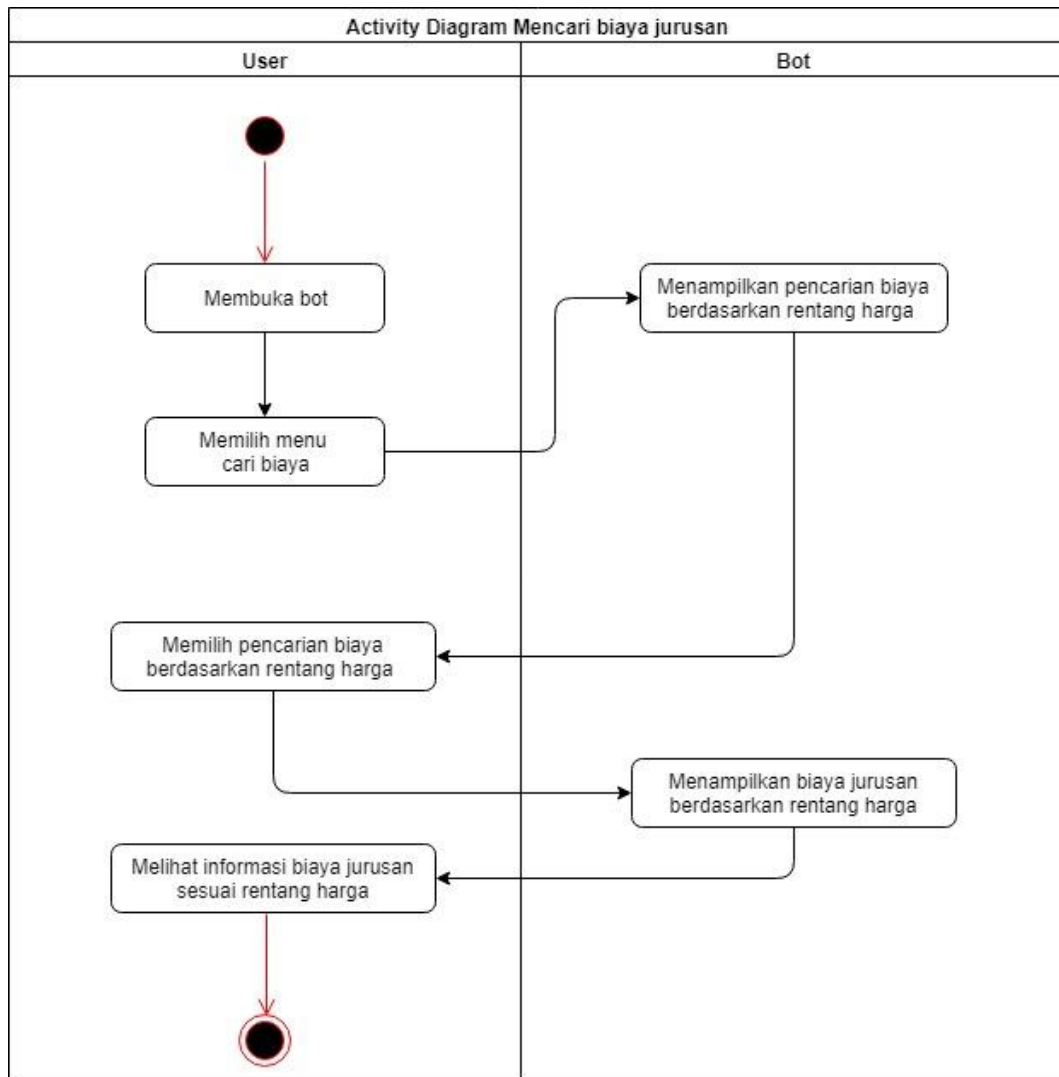
Activity diagram cari jurusan di Kota Bandung menggambarkan kegiatan-kegiatan yang terjadi pada saat pengguna mencari jurusan di Kota Bandung. Gambar III.13 menunjukkan activity diagram mencari jurusan di Kota Bandung:



Gambar III.22 Activity diagram menu cari jurusan

2.6 Activity Diagram menu cari biaya

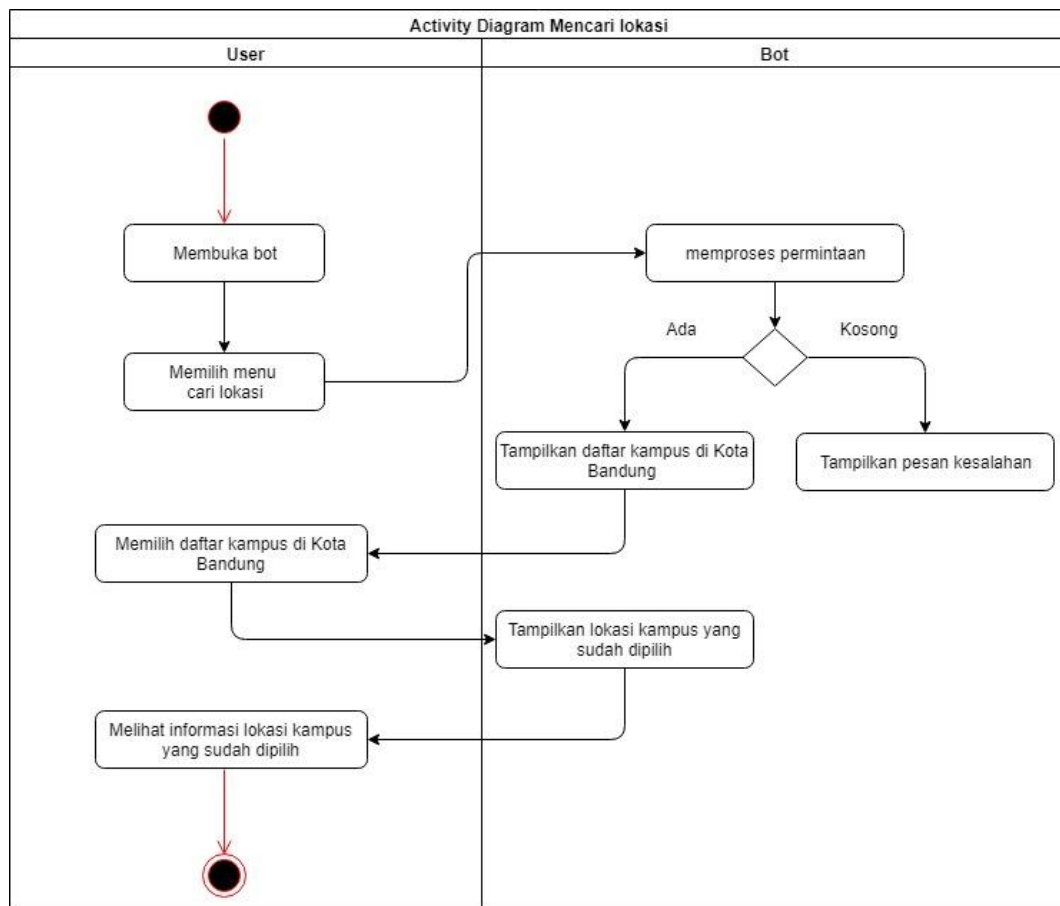
Activity diagram Mencari informasi pencarian biaya jurusan yang menggambarkan kegiatan-kegiatan yang terjadi pada saat pengguna mencari informasi cari biaya jurusan berdasarkan rentang harga. Gambar III.14 menunjukkan activity diagram mencari informasi biaya jurusan berdasarkan rentang harga:



Gambar III.23 activity diagram menu cari biaya

2.7 Activity Diagram menu cari lokasi

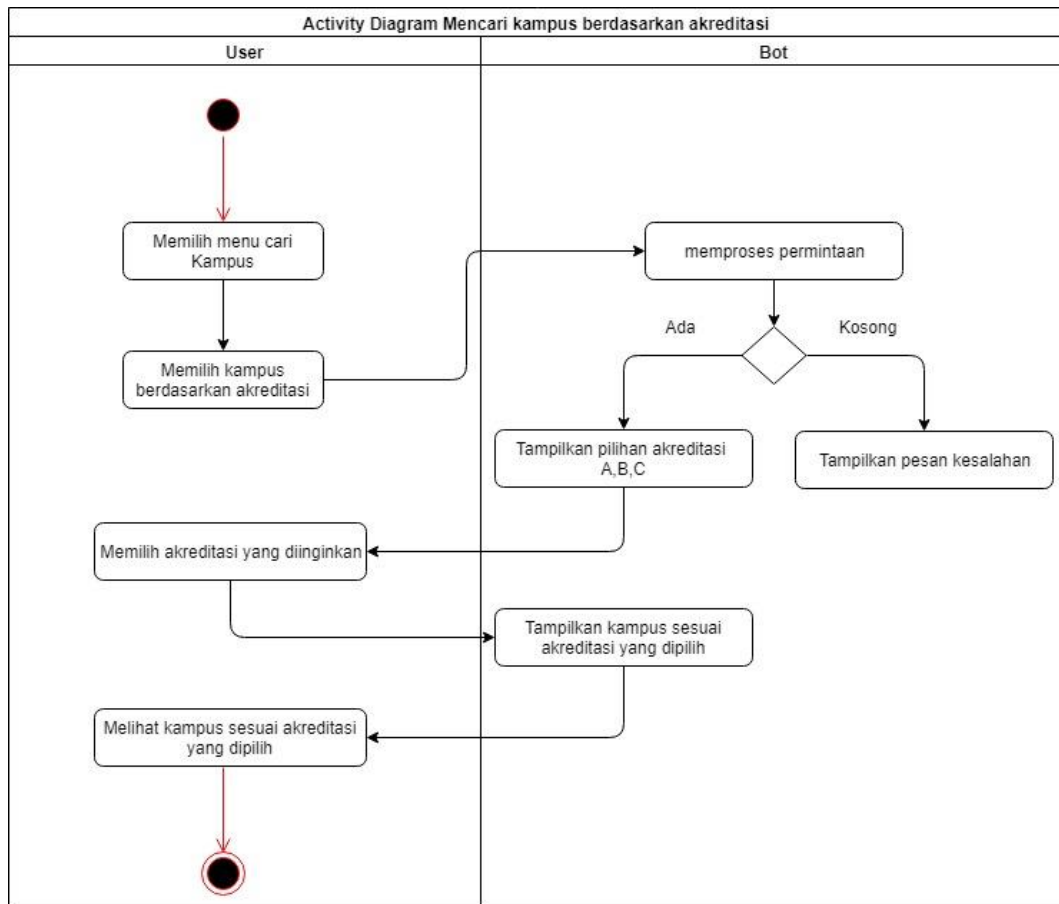
Activity diagram Mencari informasi pencari lokasi kampus yang menggambarkan kegiatan-kegiatan yang terjadi pada saat pengguna mencari informasi lokasi kampus. Gambar III.15 menunjukkan activity diagram mencari informasi lokasi kampus :



Gambar III.24 activity diagram menu cari lokasi

2.8 Activity Diagram Mencari kampus berdasarkan akreditasi

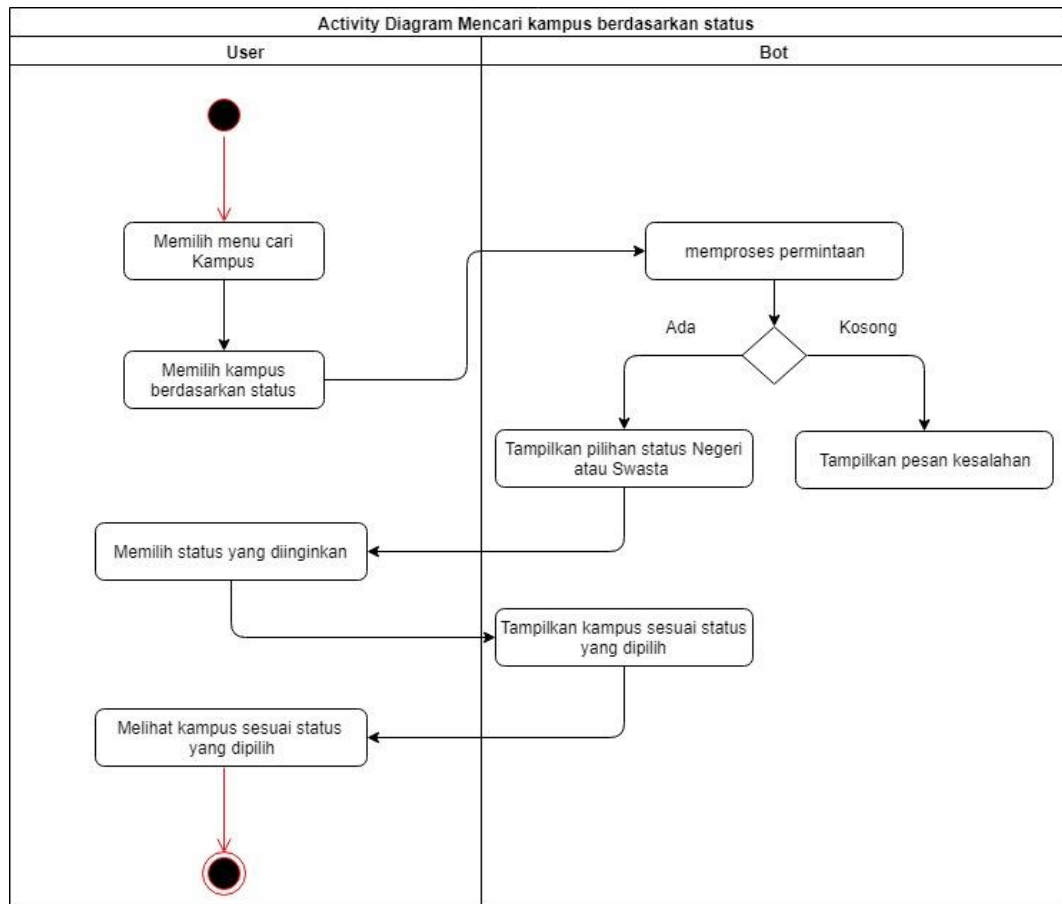
Activity diagram Mencari informasi kampus berdasarkan akreditasi yang menggambarkan kegiatan-kegiatan yang terjadi pada saat pengguna mencari informasi kampus berdasarkan akreditasi. Gambar III.16 menunjukkan activity diagram mencari informasi kampus berdasarkan akreditasi :



Gambar III.25 activity diagram mencari kampus berdasarkan akreditasi

2.9 Activity Diagram Mencari kampus berdasarkan status

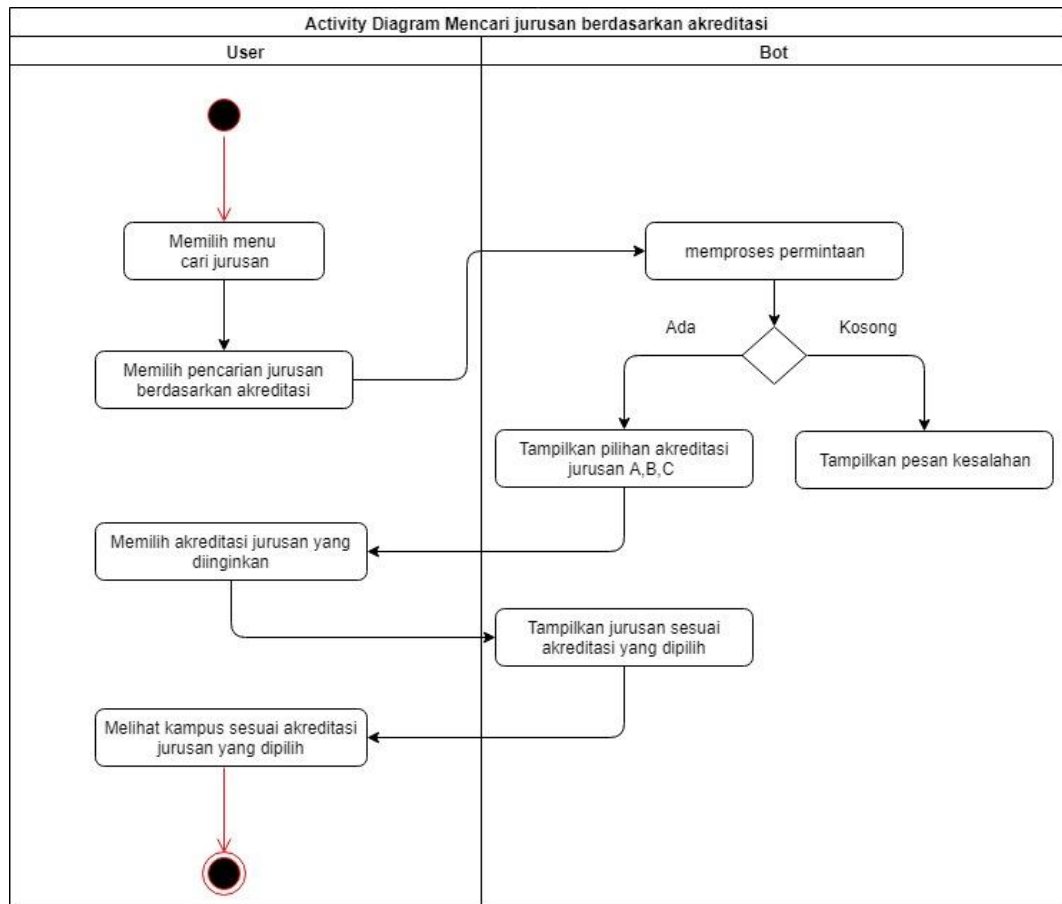
Activity diagram Mencari informasi kampus berdasarkan status yang menggambarkan kegiatan-kegiatan yang terjadi pada saat pengguna mencari informasi kampus berdasarkan status. Gambar III.17 menunjukkan activity diagram mencari informasi kampus berdasarkan status :



Gambar III.26 activity diagram mencari kampus berdasarkan status

2.10 Activity Diagram Mencari jurusan berdasarkan akreditasi

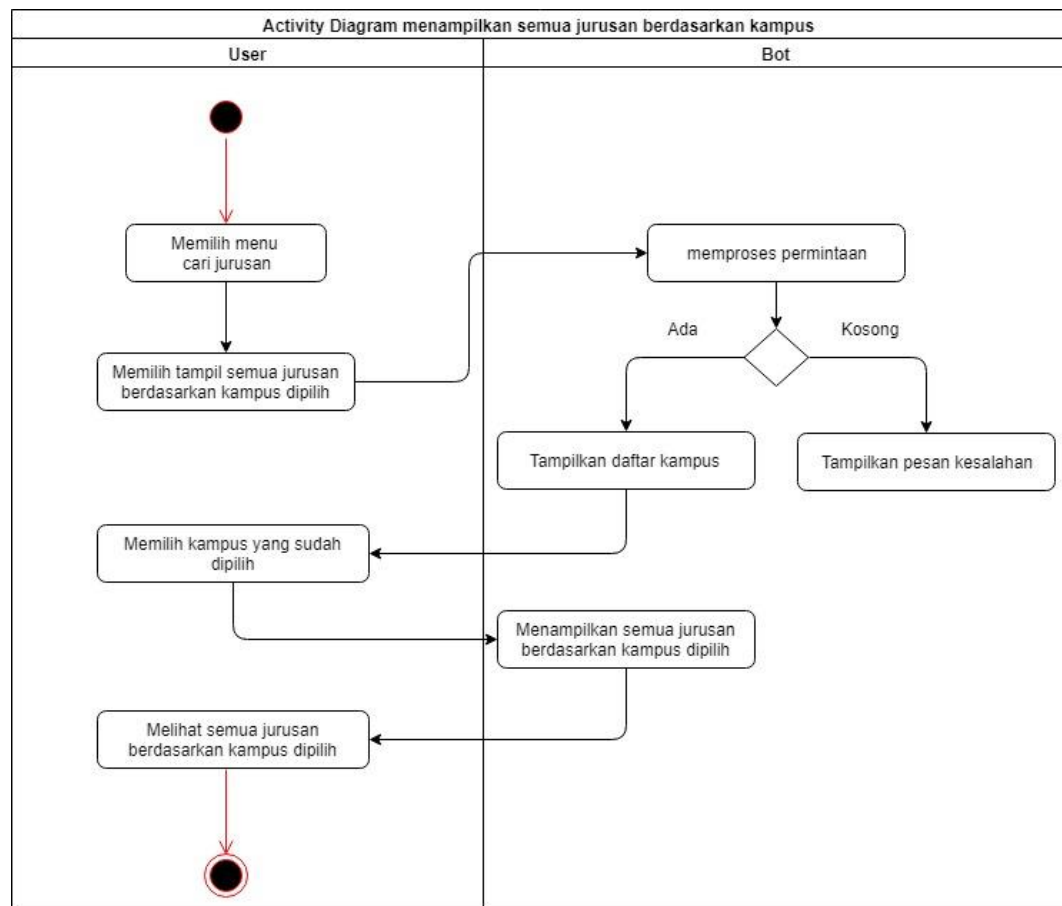
Activity diagram Mencari informasi jurusan berdasarkan akreditasi yang menggambarkan kegiatan-kegiatan yang terjadi pada saat pengguna mencari informasi jurusan berdasarkan akreditasi. Gambar III.17 menunjukkan activity diagram mencari informasi jurusan berdasarkan akreditasi :



Gambar III.27 activity diagram mencari jurusan berdasarkan akreditasi

2.11 Activity Diagram menampilkan semua jurusan berdasarkan kampus

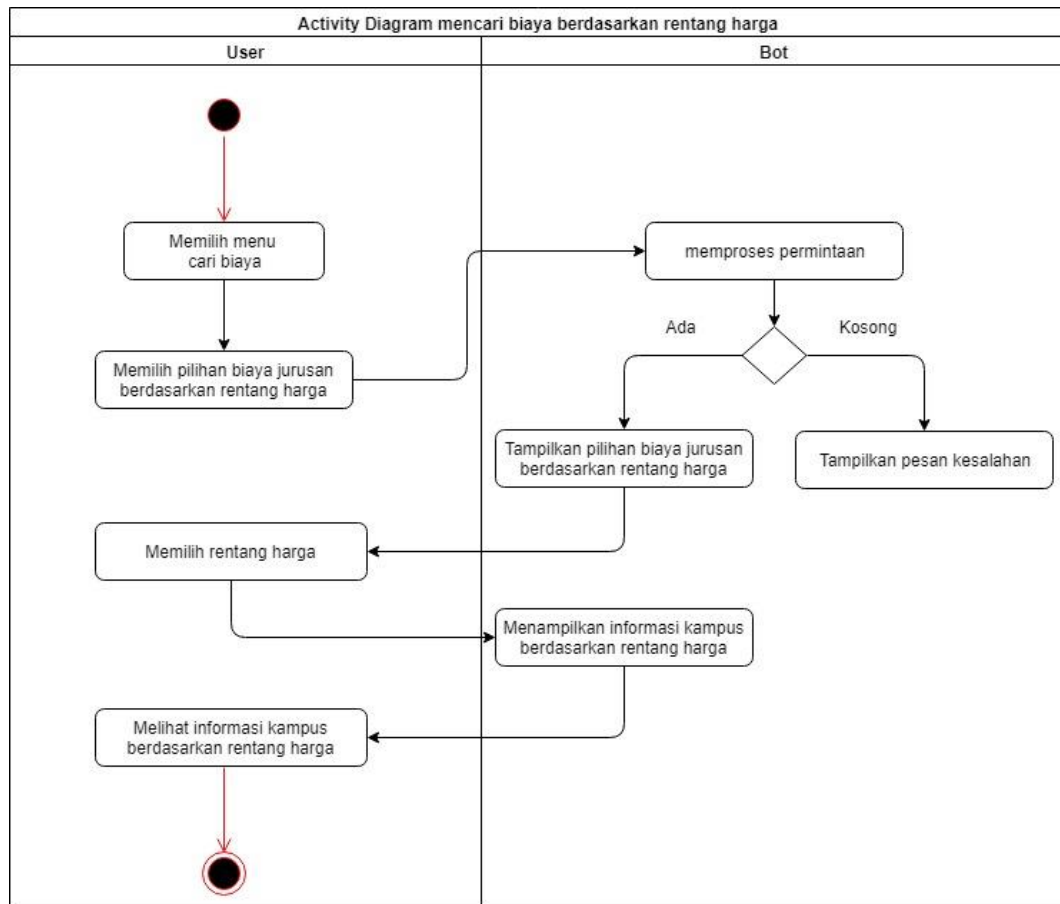
Activity diagram menampilkan semua informasi jurusan berdasarkan kampus yang menggambarkan kegiatan-kegiatan yang terjadi pada saat pengguna mencari informasi semua jurusan berdasarkan kampus. Gambar III.17 menunjukkan activity diagram menampilkan semua informasi jurusan berdasarkan kampus :



Gambar III.28 activity diagram menampilkan semua jurusan berdasarkan kampus

2.12 Activity Diagram menampilkan biaya berdasarkan rentang harga

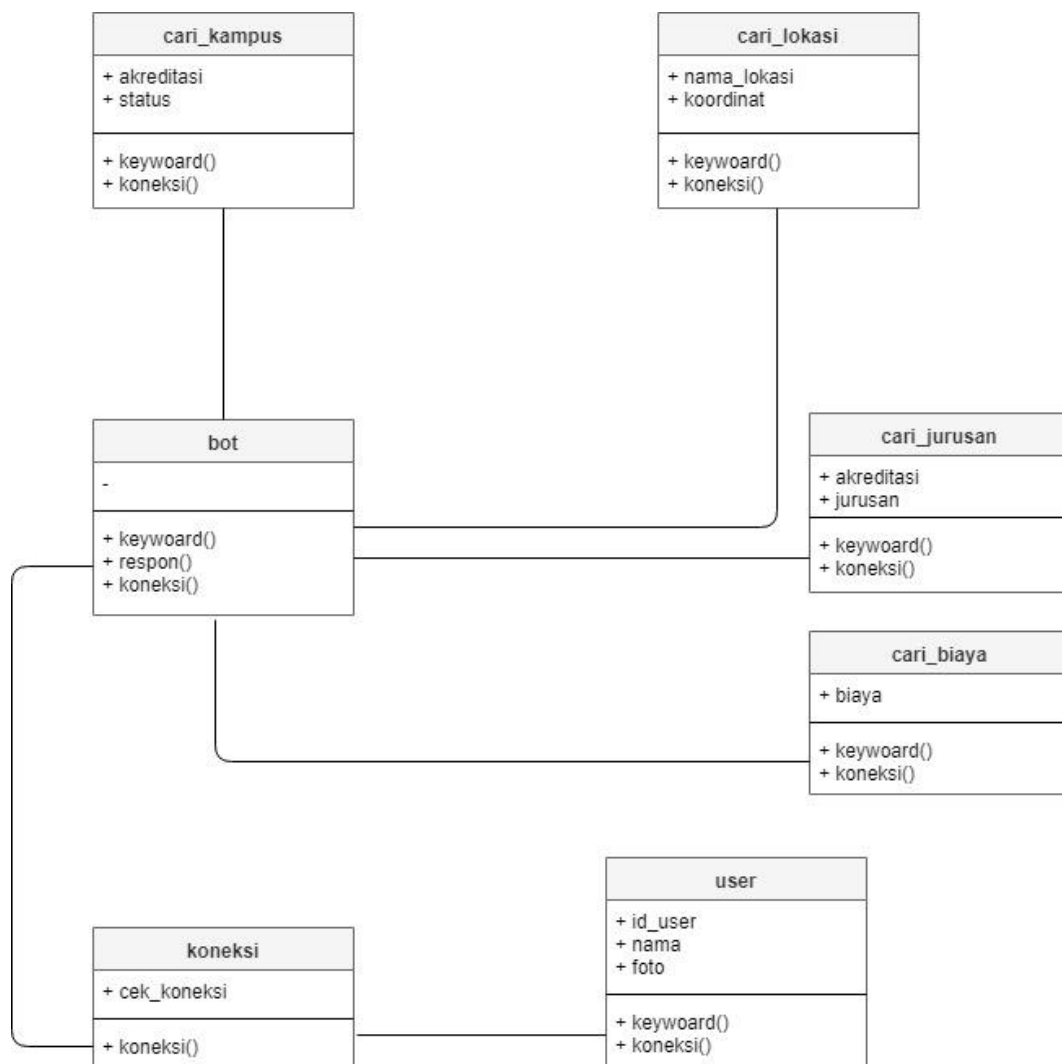
Activity diagram menampilkan biaya berdasarkan rentang harga yang menggambarkan kegiatan-kegiatan yang terjadi pada saat pengguna mencari informasi biaya berdasarkan rentang harga. Gambar III.17 menunjukkan activity diagram menampilkan biaya berdasarkan rentang harga :



Gambar III.29 activity diagram mencari biaya berdasarkan rentang harga

3. Class Diagram

Class diagram keseluruhan digunakan untuk menggambarkan struktur dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem serta hubungannya antara kelas. Class diagram keseluruhan dapat dilihat pada gambar III.18.

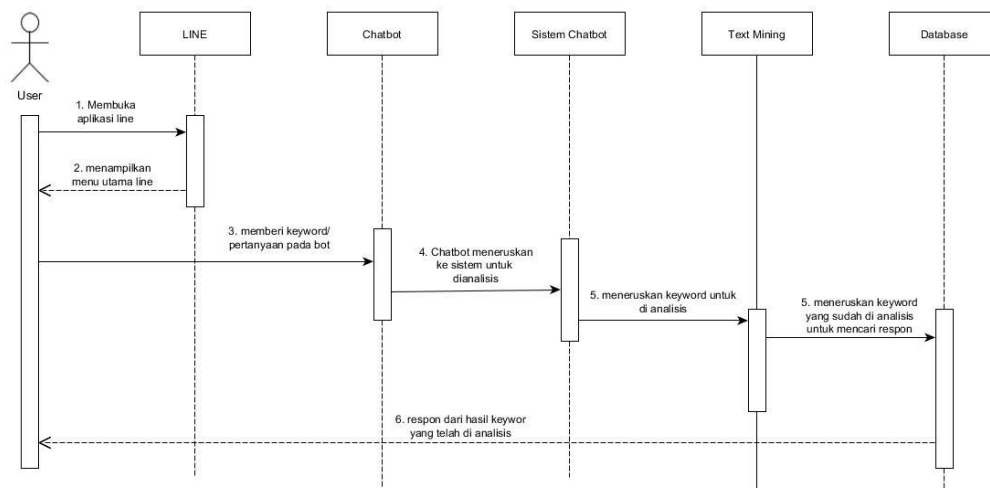


Gambar III.30 Class Diagram

4. Sequence Diagram

Proses ini menggambarkan interaksi antara sejumlah objek dalam urutan waktu. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara objek juga interaksi antara objek yang terjadi pada titik tertentu dalam eksekusi sistem. Sequence diagram system klasifikasi ini yaitu Sequence Diagram Pertanyaan. Sequence Diagram Pertanyaan merupakan diagram yang menyajikan proses menu pertanyaan dan interaksi dengan sistem chatbot dapat dilihat pada gambar 3.15. Penjelasan langkah-langkahnya adalah sebagai berikut :

1. User (siswa SMA/K) membuka aplikasi LINE.
2. LINE akan menampilkan menu utama LINE.
3. User memberika keyword/pertanyaan pada chatbot.
4. Chatbot akan meneruskannya ke sistem untuk dianalisa.
5. Sistem chatbot menganalisis dengan menggunakan metode forward chaining
6. Setelah dianalisis maka diteruskan ke database untuk mencari respon
7. User menerima respon jawaban dari hasil keyword yang sudah diolah



Gambar Error! No text of specified style in document..31 Sequence Diagram Chatbot

5. Perancangan Basis Data

Perancangan basis data yaitu perancangan yang bisa menciptakan atau merancang kumpulan data yang terhubung dan tersimpan secara bersamaan.

5.1 Skema Relasi

Pada tahap ini akan digambarkan struktur database yang digunakan. Struktur relasi database ini dapat dilihat pada gambar

6. Perancangan Struktur Menu
7. Perancangan Antarmuka
8. Perancangan Pesan Antarmuka
9. Perancangan Jaringan Semantik Antarmuka