

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

Penelitian yang bertujuan untuk mengembangkan dan menguji kualitas aplikasi berbasis *chatbot* sebagai media komunikasi kehumasan dengan menggunakan model pengembangan Agile: Scrum dilaksanakan dalam langkah iteratif terhadap beberapa tahapan. Bagian pertama (*planning*) menyajikan hasil penelitian terhadap kebutuhan pengembangan. Bagian kedua (*analysis & design*) menyajikan hasil penelitian tentang analisis fungsional perangkat lunak, rangkaian cara kerja, serta desain percakapan dan antarmuka *chatbot*. Bagian ketiga (*development*) menyajikan hasil penelitian tentang teknis pengembangan. Bagian keempat (*testing*) menyajikan hasil penelitian terhadap pengujian enam karakteristik menurut ISO/IEC 20510. Bagian kelima (*deployment*) menyajikan hasil penelitian terhadap langkah publikasi dan integrasi *chatbot*.

A. Hasil Penelitian

1. *Planning*

Pada tahap *planning*, penulis melakukan wawancara dengan Bapak Dedy Herdito, M.M. selaku staf humas UNY serta penyebaran kuesioner pelayanan humas dan kebutuhan pengembangan (media) kepada 100 mahasiswa/i seluruh fakultas di UNY. Hasil perhitungan kuesioner pelayanan humas selengkapnya tersaji pada Lampiran 7, sedangkan untuk rangkuman hasilnya tersaji pada Tabel 21 berikut.

Tabel 21. Rangkuman Hasil Perhitungan Kuesioner Pelayanan Humas UNY

No	Aspek	Skor	Persentase
<i>Tangibles</i>			
1	Ketersediaan media informasi yang menarik dan lengkap (web, sosial media, brosur, dan sebagainya)	372	74,40%
2	Saya merasa puas dengan sumber informasi yang disediakan	346	69,20%
3	Saya mendapatkan kemudahan dalam proses pelayanan	355	71,00%
4	Saya sering memanfaatkan layanan informasi yang disediakan	347	69,40%
5	Informasi yang saya butuhkan tersedia setiap saat	309	61,80%
<i>Reliability</i>			
6	Saya mendapati kecermatan petugas dalam melayani pengguna	347	69,40%
7	Konten layanan informasi yang disajikan memenuhi harapan saya	336	67,20%
<i>Responsiveness</i>			
8	Saya mendapatkan respon setiap proses pelayanan	352	70,40%
9	Saya mendapatkan jawaban/informasi dengan tepat waktu.	328	65,60%
10	Hampir semua pertanyaan yang pernah saya ajukan direspon oleh petugas.	337	67,40%
<i>Assurance</i>			
11	Saya mendapati petugas memberikan jaminan tepat waktu dalam pelayanan	335	67,00%
12	Saya mendapatkan legalitas atau bukti valid tentang informasi yang disampaikan.	361	72,20%
<i>Empathy</i>			
13	Saya memperoleh pelayanan dari petugas dengan mendahulukan kepentingan permohonan pengguna	344	68,80%
14	Saya mendapatkan pelayanan dari petugas dengan ramah dan sopan santun	370	74,00%
15	Saya mendapatkan pelayanan dari petugas dengan tidak diskriminatif	375	75,00%
NILAI KESELURUHAN			5214
PERSENTASE NILAI KESELURUHAN			69,52%

Berdasarkan hasil perhitungan pada Tabel 21, pelayanan humas UNY mendapatkan persentase nilai keseluruhan sebesar 69,52% dengan kategori Baik. Namun terdapat beberapa pelayanan humas yang memiliki nilai di bawah rata-rata

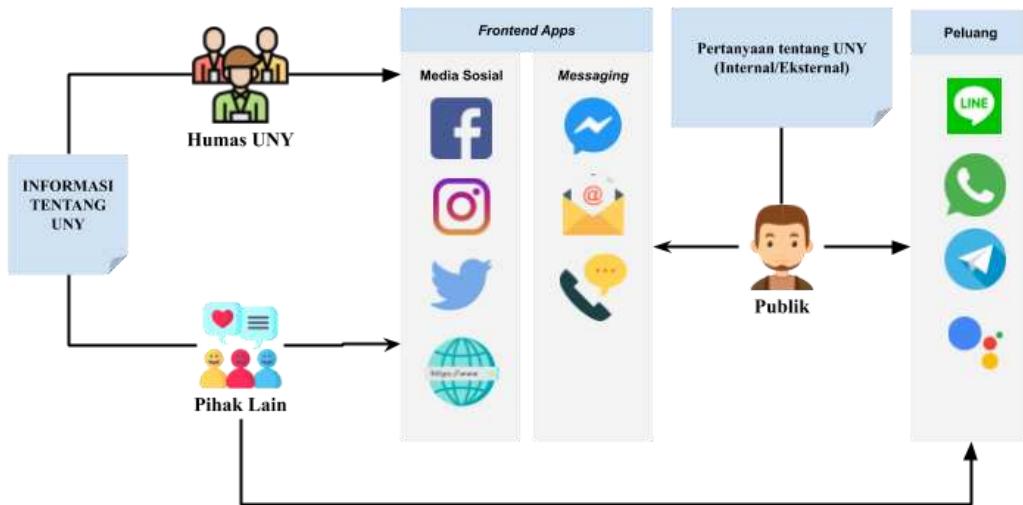
pada aspek tertentu. Selain hasil perhitungan pelayanan Humas UNY, hasil wawancara yang tersaji dalam Lampiran 5 dan hasil kuesioner kebutuhan media dalam Lampiran 9 dideskripsikan pada narasi berikut.

Bapak Dedy Herdito, M.M. memberikan penjelasan bahwa media komunikasi dan informasi kehumasan UNY yang digunakan saat ini cukup beragam, di antaranya adalah Instagram (51,5 ribu pengikut), Facebook (12,6 ribu pengikut), Twitter (5,2 ribu pengikut), dan YouTube (5,8 ribu pengikut) dengan ID @unyofficial per tanggal 29 April 2019. Sedangkan *website* resmi UNY memiliki data kunjungan terbanyak pada bulan januari 2019 dengan 3,07 juta pengunjung. 83,95% *traffic* berasal dari hasil mesin pencarian, sedangkan yang mengunjungi langsung ke alamat *website* hanya sekitar 12,91%, sisanya berasal dari *referrals*, sosial media, *email*, dan iklan (Similarweb, 2019). Data tersebut diperkuat dengan hasil kuesioner kebutuhan media bahwa 89% mahasiswa lebih memilih *instagram* untuk mengetahui informasi terbaru tentang UNY. Namun untuk informasi baku atau statis umumnya tersedia di *website*. Selain itu, pengguna dapat berinteraksi dengan humas UNY melalui *telephone* atau *email*. Beliau menyampaikan bahwa:

Walaupun informasi sudah ada di *website* namun pengguna terkadang masih bertanya melalui *email*, terutama yang berkaitan dengan Penerimaan Mahasiswa Baru (PMB) karena sudah mencakup sebagian informasi penting tentang UNY seperti program studi, jurusan, akreditasi, beasiswa, dan sebagainya.

Selain menggunakan aplikasi tersebut, pengguna juga menggunakan berbagai macam aplikasi lain yang tidak kalah populer, di antaranya adalah WhatsApp (49%), Telegram (5%), dan Line(9%). Informasi didapatkan dengan menghubungi

kerabat, dosen, maupun pihak Humas UNY. Berikut alur interaksi publik dengan Humas UNY yang tersaji pada Gambar 15 berikut.



Gambar 15. Interaksi publik dengan Humas UNY

Hasil kuesioner kebutuhan media menyebutkan bahwa jenis konten yang dibutuhkan di antaranya adalah informasi tentang beasiswa, program studi, kontak penting, prestasi, lingkungan kampus, tentang PMB, jadwal perkuliahan, lokasi bangunan, urusan administrasi tugas akhir, hingga publikasi kegiatan. Namun beliau berpendapat bahwa:

Umumnya pertanyaan datang dari pihak *eksternal* yakni calon mahasiswa, guru sekolah, maupun orang tua calon mahasiswa itu sendiri. Sedangkan yang sudah menjadi mahasiswa (*internal*) umumnya bertanya tentang cara publikasi karya. Namun tidak semua pertanyaan dapat kami jawab, seperti pertanyaan tentang akademik, kemahasiswaan, dan sebagainya yang memang ada bagiannya sendiri.

Pendapat tersebut menegaskan bahwa tidak semua informasi yang berkaitan dengan UNY dapat dijawab oleh staf humas UNY atau satu lembaga saja. Permasalahan tentang pemetaan peran memang umum terjadi, karena pengguna umumnya merasa bahwa mereka dapat bertanya tentang apapun kepada humas. Selain masalah generalisasi peran, humas UNY memiliki *rating* yang rendah

terhadap ketersediaan informasi dan ketepatan waktu menjawab pertanyaan dibandingkan dengan rata-rata seluruh aspek pelayanan humas. Hasil kuesioner pelayanan humas menyatakan bahwa sebesar 61,8% mahasiswa merasa cukup untuk ketersediaan informasi. Sedangkan 65,6% mahasiswa menyatakan bahwa ketepatan waktu dalam menerima jawaban juga tergolong cukup. Hal tersebut diakui oleh staf UNY bahwa “Karena jam kerja staf humas terbatas (07.30–16.00 WIB) menyebabkan pelayanan belum sepenuhnya optimal, akhirnya banyak pertanyaan masuk diluar jam kerja yang harus dijawab tetapi tidak tepat pada waktunya.”

Kendala tersebut dapat menjadi salah satu faktor 63% mahasiswa jarang mengakses informasi tentang UNY. Dan sebagai alternatif, akhirnya 68% mahasiswa lebih menyukai bertanya dengan teman atau kerabat dibandingkan dengan kanal resmi yang dikelola oleh UNY (62%) terutama pada informasi baku atau statis. Rencana pengembangan *chatbot* atau *virtual assistant* untuk membantu pekerjaan humas sangat didukung. Dalam pandangan Humas UNY, pengembangan media merupakan salah satu prioritas utama sebagai wujud inovasi pelayanan humas. Beliau berpendapat bahwa:

Saya rasa konsep *chatbot* atau *virtual assistant* sudah banyak yang mengetahui, namun saya pribadi masih jarang menggunakannya. Karena belum ada kepentingan yang berarti disana. Namun teknologi canggih seperti *chatbot* atau *virtual assistant* berpotensi dapat membantu pekerjaan humas UNY terutama sebagai *customer service* yang dapat tersedia setiap saat atau *realtime*.

Pendapat tersebut diperkuat oleh hasil kuesioner kebutuhan pengembangan bahwa sebesar 65% mahasiswa merasa *chatbot* atau *virtual assistant* dapat membantu pekerjaan humas UNY. Didukung juga dengan pengetahuan calon

pengguna bahwa sebesar 65% mahasiswa sudah mengetahui tentang *chatbot* atau *virtual assistant* seperti google assistant, cortana, dan siri. Namun Google Assistant lebih populer karena umumnya sudah tertanam pada *smartphone* android (94% mahasiswa UNY menggunakan android).

Beliau berharap *chatbot* atau *virtual assistant* dapat berinteraksi dengan santai menyesuaikan pengguna yang pada umumnya adalah generasi milenial. Hal tersebut senada dengan hasil kuesioner kebutuhan pengembangan oleh mahasiswa bahwa karakter yang lebih santai sangat disukai dibandingkan dengan *chatbot* berkarakter formal.

Akan tetapi, baik staf humas UNY dan sebagian calon pengguna masih memiliki kekhawatiran pada perilaku *chatbot* kedepannya, yakni masalah miskomunikasi atau kesalahan konten informasi yang disampaikan. Dengan harapan supaya kesalahan tersebut dapat diminimalisir untuk menjaga citra positif perguruan tinggi.

2. *Analysis & Design*

a. **Analisis Fungsionalitas**

Peneliti mengidentifikasi kebutuhan fungsionalitas *chatbot* berdasarkan rangkuman *user stories* pada tahap *planning*. Hasil dari proses identifikasi kebutuhan fungsionalitas yang juga merupakan *product backlog* dari *chatbot* UNYSA tersaji pada Tabel 22 berikut.

Tabel 22. Kebutuhan Fungsional

No	Aspek	Kebutuhan Fungsional
1	Performa	<i>Chatbot</i> dapat beroperasi secara <i>realtime</i> untuk melayani pengguna.
2	Integrasi	<i>Chatbot</i> dapat diakses dengan mudah melalui berbagai macam <i>frontend apps</i> yakni <i>WhatsApp</i> , <i>Telegram</i> , <i>Line</i> , <i>Messenger</i> , dan <i>Google Assistant</i> .
3	Personalisasi	<i>Chatbot</i> direpresentasikan sebagai karakter perempuan.
4	Personalisasi	<i>Chatbot</i> diharapkan ramah terhadap pengguna dengan bahasa yang santai dan mudah dipahami.
5	Personalisasi	Staf Humas dapat melatih dan menyesuaikan jawaban <i>chatbot</i> .
6	Konten	Konten disajikan berupa teks, audio, gambar, dan video.
7	Konten	<i>Chatbot</i> dapat mengambil konten dari <i>website resmi UNY</i> .
5	Konten	Konten yang disajikan berupa informasi umum seputar perguruan tinggi (UNY) seperti tentang pendaftaran, biaya, akademik, beasiswa, profil, hingga tentang program studi.
6	Pemahaman	<i>Chatbot</i> mampu memahami pertanyaan yang bervariasi namun dalam konteks yang sama.
7	Navigasi	<i>Chatbot</i> mampu memberikan rekomendasi topik pertanyaan terkait.
8	Navigasi	<i>Chatbot</i> mampu mengakhiri percakapan dengan baik.
9	Error Management	<i>Chatbot</i> mampu menangani <i>error</i> jika pertanyaan tidak dapat dipahami atau jawaban tidak tersedia.

b. *Project Management*

Mengelola suatu proyek supaya berjalan efektif dan efisien dibutuhkan sebuah perencanaan waktu dan target dengan merinci seluruh aktivitas pada setiap fase pengembangan. Dengan menggunakan model *scrum*, suatu aktivitas dapat dilakukan perulangan pada suatu tahapan, bergantung pada perubahan yang dilakukan. Rincian perencanaan dan hasil penjadwalan *project management* tersaji pada Lampiran 10.

c. Persona

Dalam rangka meningkatkan personalitas *chatbot*, maka diperlukan karakter khusus yang merepresentasikan baik dari segi fungsi dan sifat. Persona didesain berdasarkan hasil kuesioner kebutuhan pengembangan media dengan mengikuti panduan desain persona Google yang diimplementasikan pada Tabel 23 berikut.

Tabel 23. Hasil Desain Persona

	Nama Persona	UNYSA (UNY Smart Assistant)
	Kata sifat utama	Lugas, Teknis, Ramah
	Karakter yang mewujudkan	Staf Humas UNY (Ahli), Dosen & Pegawai, serta Mahasiswa.
	Deskripsi	Staf Humas UNY adalah mereka yang mengetahui mayoritas informasi seputar UNY, yang merupakan bagian dari tugas humas untuk menjembatani informasi dari lembaga ke publik. Selain staf humas UNY, informasi diketahui sebagian pegawai, dosen, dan mahasiswa yang merupakan bagian dari civitas akademika UNY.

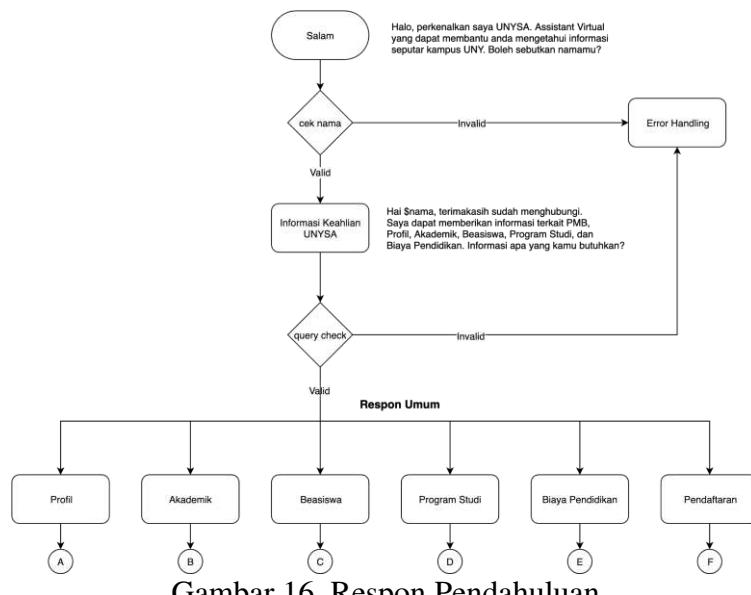
d. *Conversational Design*

Conversational Design merupakan desain aliran percakapan didalam *chatbot*. Hal ini bertujuan agar percakapan antara pengguna dan *bot* lebih terarah dan meminimalisir *error*. Ketika pengguna bertanya tentang sesuatu, sebelum keinginan pengguna terpenuhi maka *bot* akan menyediakan keterangan atau langkah-langkah kepada pengguna supaya pertanyaan tersebut dapat dimengerti dan diberikan jawaban yang tepat. Namun tidak menutup kemungkinan jika pengguna bertanya langsung secara spesifik terhadap informasi yang diinginkan, maka seharusnya *chatbot* juga tetap dapat melayani sejauh pertanyaan masuk dalam konteks dan terdapat jawaban yang relevan.

Berdasarkan hasil wawancara dan kuesioner didapatkan data tentang topik yang sering ditanyakan oleh pengguna, di antaranya adalah tentang profil, akademik, beasiswa, program studi, biaya pendidikan, dan pendaftaran. Kemudian penulis melakukan analisa untuk menentukan aliran percakapan. Hasil desain aliran percakapan selengkapnya tersaji dalam Lampiran 11. Berikut adalah contoh desain aliran percakapan untuk topik pendahuluan dan profil.

1) Respon Pendahuluan

Respon Pendahuluan merupakan tahap memulai percakapan, dimulai dari salam yang dilakukan oleh *chatbot* ketika ia dipanggil oleh pengguna. Kemudian *bot* menawarkan lingkup informasi yang bisa dilayani seperti yang tersaji pada Gambar 16 berikut.

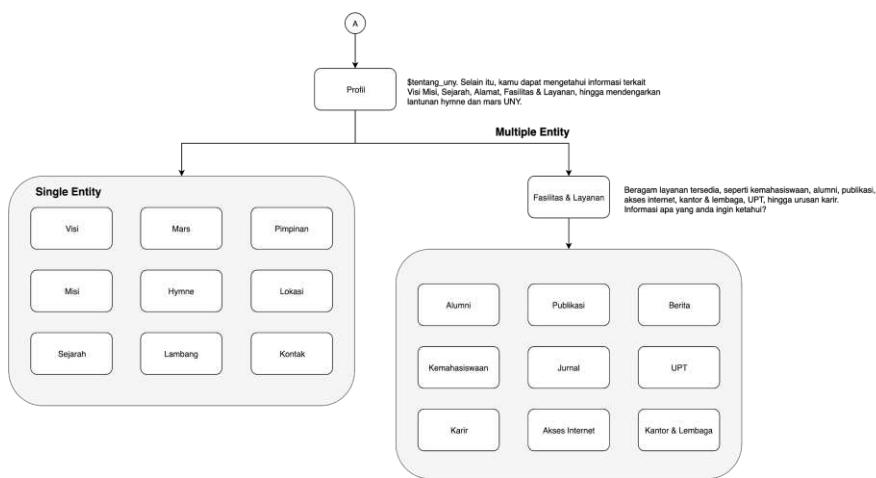


Gambar 16. Respon Pendahuluan

2) Topik tentang Profil UNY

Profil UNY terdiri dari berbagai macam topik yang secara umum hanya terdiri dari entitas tunggal misalkan visi, mars, pimpinan, sejarah, dan berbagai macam entitas lainnya. Sedangkan untuk entitas jamak (bercabang) misalkan UPT

Perpustakaan. Jika pengguna bertanya dan menghasilkan nilai entitas hanya UPT (tunggal) maka akan dijawab tentang UPT secara umum. Namun jika nilai entitas UPT Perpustakaan, maka akan dijawab secara spesifik tentang perpustakaan tersebut. Lingkup kemampuan *chatbot* pada topik profil UNY tersaji pada Gambar 17 berikut.



Gambar 17. Topik tentang Profil UNY

e. Sumber Data

Berdasarkan hasil penelitian kebutuhan, UNYSA menggunakan *database* google sheet untuk menyimpan informasi jawaban atas pertanyaan yang bersifat umum. Sedangkan untuk informasi dinamis dengan banyak nilai berbeda seperti data akreditasi program studi, jenjang, fakultas, berita, pengumuman, dan Uang Kuliah Tunggal (UKT) menggunakan teknik *web scraping* terhadap informasi yang sudah tersedia di *website* UNY. Sumber data respon *chatbot* melalui *web scraping* tersaji pada Tabel 24 berikut.

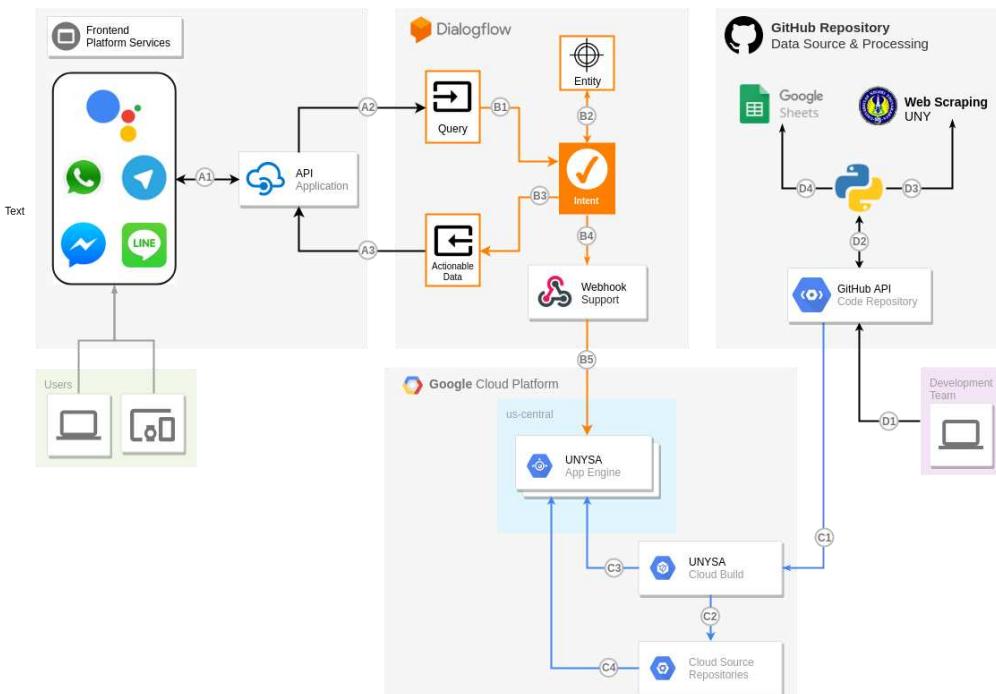
Tabel 24. Sumber Data Respon *Chatbot*

UKT	http://pmb.uny.ac.id/biaya-pendidikan/uang-kualiah-tunggal-ukt-program-studi-sarjana-terapan-ster
	http://pmb.uny.ac.id/biaya-pendidikan/uang-kuliah-tunggal-ukt-program-studi-s1
Akkreditasi	http://pdpt.uny.ac.id/dtakreditasi

Tampilan situs pada Tabel 24 saat dilakukan *scraping* tersaji dalam Lampiran 12. Data tersebut mungkin tidak lengkap untuk penelitian saat ini, dan lebih banyak situs yang perlu ditambahkan di kemudian hari. Selain itu, respon dapat ditambahkan ke daftar secara manual pada google sheet jika jenis informasi sulit dilakukan *scraping*.

f. Desain Arsitektur

Desain arsitektur menggambarkan proses secara keseluruhan bagaimana *chatbot* dapat bekerja. Pengguna dari beragam aplikasi *frontend apps* yang digunakan untuk antarmuka percakapan. Kemudian dilakukan pemrosesan NLU pada dialogflow untuk memahami konteks pertanyaan dan memberikan respon yang tepat. Hingga bagaimana menjalankan *fulfillment code* pada Google Cloud Platform dengan fitur *Continuous Integration/Continuous Delivery* (CI/CD) seperti yang tersaji pada Gambar 18 berikut.



Gambar 18. Desain Arsitektur Sistem

Dengan desain arsitektur tersebut, diharapkan layanan dapat berjalan dengan baik dan pengembangan tetap dapat berjalan secara agile. Berikut penjelasan dari desain arsitektur *chatbot*.

1) *Frontend Apps*

Adalah aplikasi yang digunakan sebagai antarmuka *chatbot*. Sesuai dengan hasil analisis kebutuhan fungsional, aplikasi yang digunakan untuk menjalankan *chatbot* terdiri dari berbagai macam platform (*multi-platform*) seperti WhatsApp, Telegram, Messenger, Line, dan Google Assistant. Untuk mengintegrasikan seluruh aplikasi tersebut dibutuhkan *API Service* (A1) dari masing-masing aplikasi untuk kemudian dihubungkan dengan dialogflow. API digunakan untuk mengirimkan data dari pengguna (A2) dan menampilkan respon dari *chatbot* (A3).

2) Dialogflow

Dialogflow menerima *query* dari pengguna dan dilakukan identifikasi *intent* yang cocok berdasarkan *query* tersebut (B1) dengan mengklasifikasikan *entity* (B2) agar respon yang dikirimkan ke pengguna tepat. Jika respon terdapat dalam *intent* maka data dikirimkan secara langsung ke pengguna (B3), namun jika teridentifikasi pada *external resources* (B4) maka data respon dipanggil melalui *webhook* (B5).

3) Google Cloud Platform (GCP)

Berfungsi sebagai layanan *deployment script* dengan menggunakan modul app engine paket standar F1 (*default*) yang secara otomatis dapat melakukan *scale up* sesuai kondisi penggunaan, selain itu pada layanan GCP diaktifkan juga fitur *CI/CD* untuk menunjang implementasi model Agile: Scrum. CI/CD diimplementasikan menggunakan modul cloud build bersumber pada repository

github (C1) dengan melakukan *trigger* setiap perubahan yang ada pada *repository*. Peran cloud build selanjutnya adalah melakukan *automatic deployment* pada app engine (C2) dan melakukan *repository mirroring* (C3) dari github. Gambaran paket app engine pada GCP tersaji pada Gambar 19 berikut.

Instance Class	Memory Limit	CPU Limit	Supported Scaling Types
F1 (default)	256 MB	600 MHz	automatic
F2	512 MB	1.2 GHz	automatic
F4	1024 MB	2.4 GHz	automatic
F4_HIGHMEM	2048 MB	2.4 GHz	automatic
B1	256 MB	600 MHz	manual, basic
B2 (default)	512 MB	1.2 GHz	manual, basic

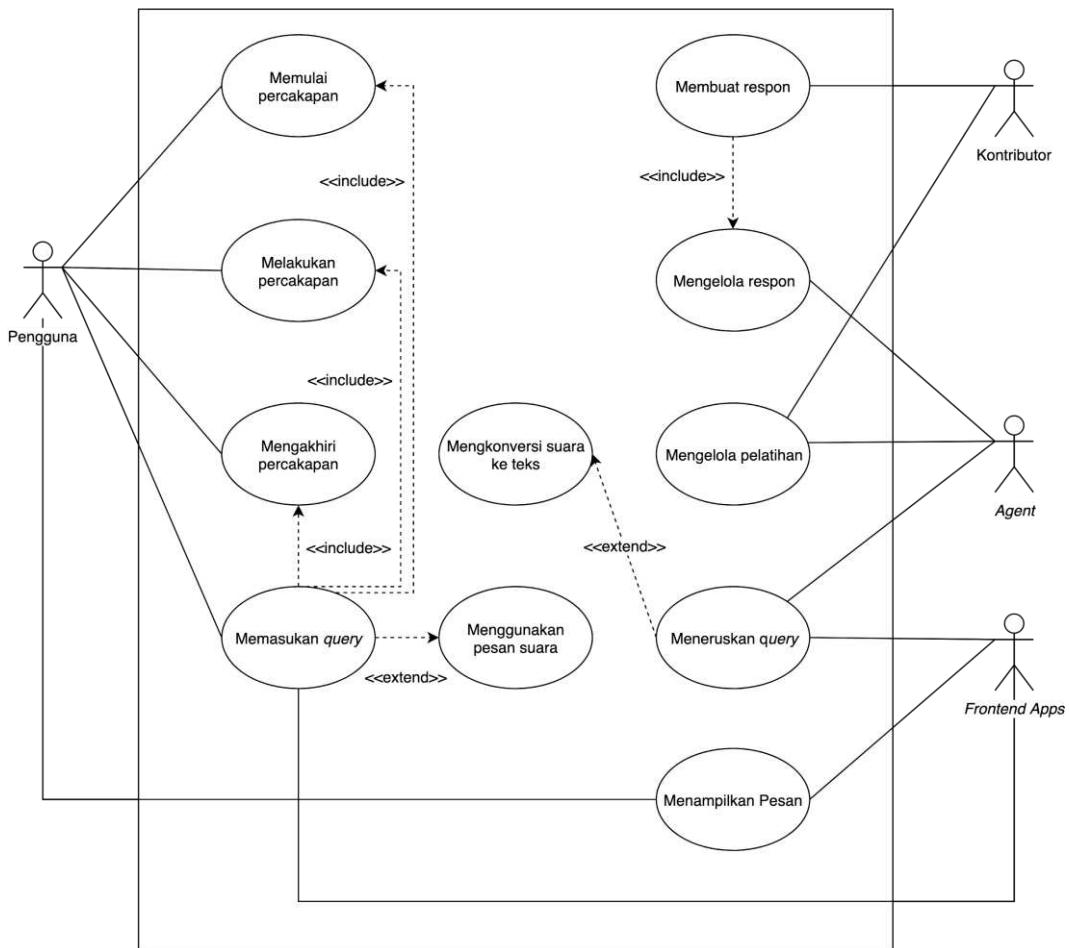
Gambar 19. Paket App Engine GCP

4) Github

Digunakan untuk *hosting script* secara *public*. Dengan menggunakan github memungkinkan untuk berkolaborasi dan *source control* lebih efektif. Pengembangan tetap dilakukan di dalam komputer lokal, saat selesai penulisan kode program dapat dilakukan *push* ke dalam github (D1). Bahasa pemrograman yang digunakan adalah python untuk memproses logika (D2) dan melakukan pengambilan data dengan teknik *web scraping* di *official website* UNY (D3) serta pada Google Sheet (D4).

g. *Use Case Diagram*

Pemodelan *use case* untuk *chatbot* menggambarkan proses aliran data interaksi antara pengguna dengan *bot* atau *agent*. *Use case diagram* untuk seluruh aktor yang terlibat tersaji pada Gambar 20 berikut.



Gambar 20. Use Case Diagram

Penjelasan aktor yang terlibat dalam aplikasi tersaji pada Tabel 25 berikut.

Tabel 25. Deskripsi Aktor

No	Aktor	Deskripsi
1	Pengguna	Pengguna adalah pengguna <i>chatbot</i> dari berbagai macam <i>platform</i> yang akan melakukan percakapan dengan <i>bot</i> .
2	Kontributor	Kontributor dapat diberikan kepada staf humas atau pengguna khusus untuk memberikan pelatihan maupun kontribusi jawaban kepada <i>chatbot</i> .
3	Frontend Apps	<i>Frontend Apps</i> adalah aplikasi yang digunakan sebagai antarmuka <i>chatbot</i> serta meneruskan pesan ke <i>bot</i> untuk diproses.
4	Agent	<i>Agent</i> melakukan pengolahan respon dengan menggunakan <i>NLU</i> pada <i>dialogflow</i> .

Sedangkan deskripsi *use case* tersaji pada Tabel 26 hingga Tabel 29 berikut.

Tabel 26. Deskripsi *Use Case Diagram*: Pengguna

No	Nama <i>Use Case</i>	Deskripsi
1	Memasukan <i>query</i>	<i>Use case</i> untuk memasukan <i>query</i> atau pesan
a	Menggunakan pesan suara	<i>Use case</i> untuk memasukan pesan suara
2	Memulai percakapan	<i>Use case</i> untuk memulai percakapan dengan <i>bot</i> .
3	Melakukan percakapan	<i>Use case</i> untuk melakukan percakapan dengan <i>bot</i> .
4	Mengakhiri percakapan	<i>Use case</i> untuk mengakhiri percakapan dengan <i>bot</i>

Tabel 27. Deskripsi *Use Case Diagram*: Kontributor

No	Nama <i>Use Case</i>	Deskripsi
1	Membuat respon	<i>Use case</i> untuk membuat respon yang akan digunakan <i>bot</i> .
2	Mengelola pelatihan	<i>Use case</i> untuk memberikan pelatihan pada <i>bot</i> .

Tabel 28. Deskripsi *Use Case Diagram*: Frontend Apps

No	Nama <i>Use Case</i>	Deskripsi
1	Menampilkan pesan	<i>Use case</i> untuk menampilkan respon <i>bot</i> .
2	Meneruskan <i>query</i>	<i>Use case</i> untuk meneruskan <i>query</i> dari pengguna ke <i>bot</i> .
a	Mengkonversi suara ke teks	<i>Use case</i> untuk melakukan konversi data dari suara ke teks.

Tabel 29. Deskripsi *Use Case Diagram*: Agent

No	Nama <i>Use Case</i>	Deskripsi
1	Mengelola respon	<i>Use case</i> untuk melakukan pengolahan respon <i>bot</i> terhadap pengguna.
2	Mengelola Pelatihan	<i>Use case</i> untuk melakukan pelatihan pada <i>bot</i> .
3	Meneruskan <i>query</i>	<i>Use case</i> untuk meneruskan pesan dari pengguna ke <i>bot</i> untuk dilakukan pemrosesan.

Setelah *use case* dibuat, langkah selanjutnya adalah menjabarkan kegiatan yang akan dilakukan pada *chatbot* dalam skenario *use case* yang secara umum menggambarkan proses percakapan antara pengguna dengan *agent*. Terdapat skenario eksekusi *main* dan *alternate flow* yang terdiri dari satu atau lebih aliran.

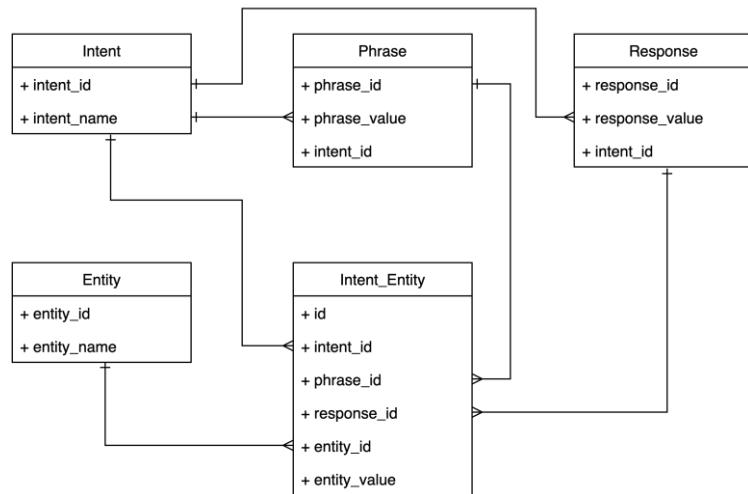
Use case umumnya menggambarkan aliran percakapan antara pengguna dengan *agent* sehingga memiliki skenario yang sama dan saling terkait antar struktur pada *use case*. Sehingga peneliti hanya membuat satu skenario yang menggambarkan seluruh struktur proses pada *use case* seperti yang tersaji pada Tabel 30 berikut.

Tabel 30. Skenario *Use Case*

Use Case	Memasukan <i>query</i>
Aktor	Pengguna, <i>Frontend Apps</i> , <i>Agent</i> , Kontributor
Deskripsi	Mendeskripsikan proses yang digunakan untuk memberikan <i>query</i> atau pesan kepada <i>agent</i> .
Successful completion	<ol style="list-style-type: none"> 1. Pengguna memasukan <i>query</i> atau pesan pada <i>frontend apps</i> berupa suara atau teks. 2. <i>Frontend apps</i> meneruskan <i>query</i> dalam bentuk teks (suara akan dikonversi ke teks) dari pengguna ke <i>agent</i>. 3. <i>Agent</i> melakukan pengolahan respon yang relevan dengan <i>query</i> pengguna. 4. Kontributor memberikan pelatihan <i>agent</i> untuk <i>query</i> yang belum dipahami. 5. Kontributor membuat respon untuk <i>query</i> baru atau yang belum dipahami. 6. <i>Agent</i> akan meneruskan respon yang dihasilkan ke <i>frontend apps</i>. 7. <i>Frontend apps</i> menampilkan respon yang dihasilkan <i>agent</i>.
Alternative	
Exception	<i>Query</i> tidak valid <ul style="list-style-type: none"> - <i>Agent</i> menginformasikan kepada pengguna bahwa <i>query</i> tidak dimengerti.
Precondition	Pengguna menggunakan salah satu <i>frontend apps</i> yang didukung.
Postcondition	Pengguna berhasil mendapatkan respon <i>agent</i> dari <i>query</i> yang dikirimkan.
Assumptions	Tidak ada

h. Class Diagram

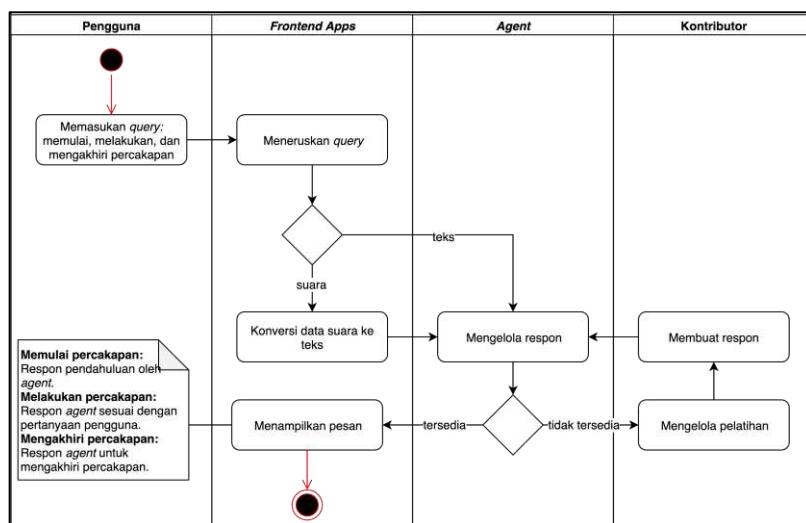
Pembuatan *class diagram* pada *chatbot* mengacu pada arsitektur dialogflow untuk melakukan pengolahan respon yang digambarkan pada *use case*. Hasil pemodelan *class diagram* tersaji dalam Gambar 21.



Gambar 21. *Class Diagram*

i. *Activity Diagram*

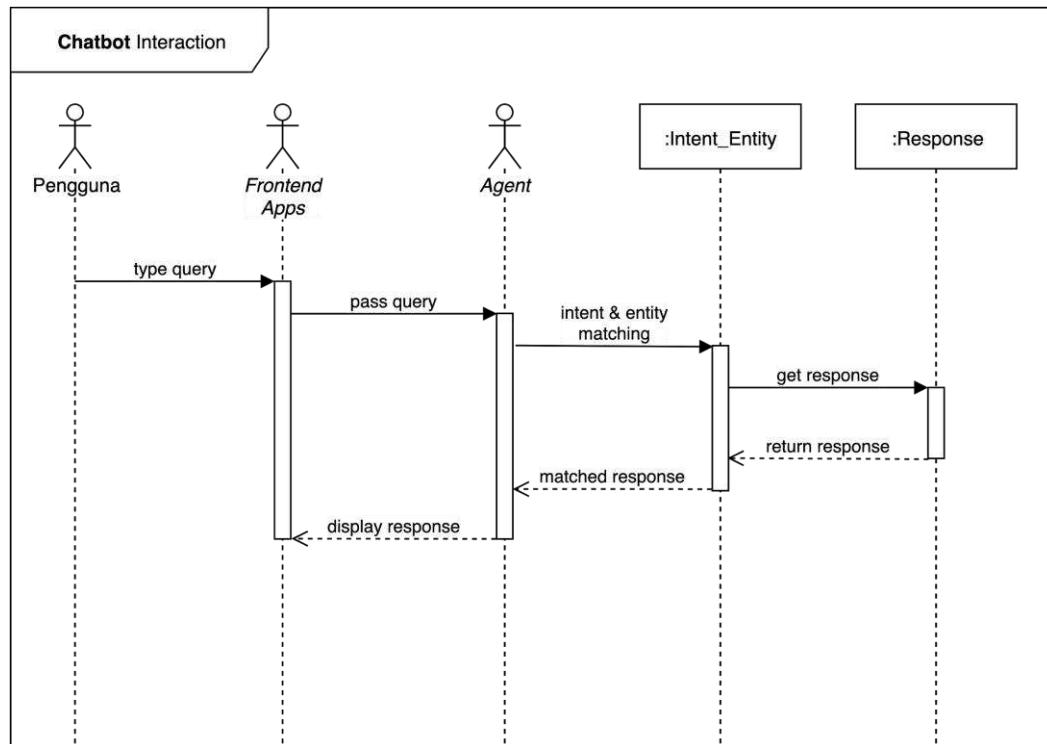
Activity diagram berikut merepresentasikan seluruh aktivitas percakapan antara pengguna dengan *agent (chatbot)* yang melibatkan aktor *frontend apps*, maupun kontributor. Hasil pemodelan *activity diagram* tersaji dalam Gambar 22.



Gambar 22. Activity Diagram

j. Sequence Diagram

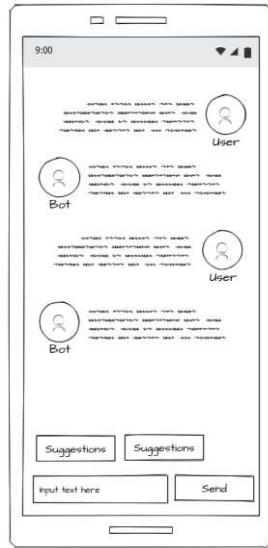
Sequence diagram menunjukkan proses transaksi data percakapan yang terjadi selama eksekusi antara aktor dan objek. Hasil pemodelan *sequence diagram* disajikan pada Gambar 23 berikut.



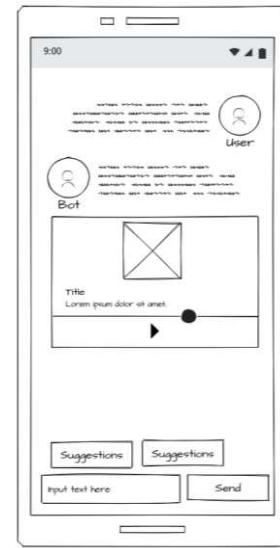
Gambar 23. *Sequence Diagram*

k. Desain Antarmuka

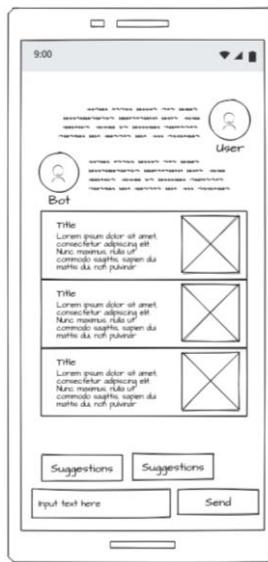
Desain antarmuka untuk pengguna pada dasarnya mengikuti desain antarmuka percakapan *frontend apps*. Hasil desain antarmuka berupa *wireframe* berturut-turut tersaji pada Gambar 24 hingga Gambar 27 menggambarkan jenis respon yang akan ditampilkan pada pengguna seperti *simple*, *media*, *list*, dan *basic responses*. Pada penelitian seluruh jenis tampilan respon dapat digunakan di Google Assistant, sedangkan untuk aplikasi lain hanya mendukung *simple responses*.



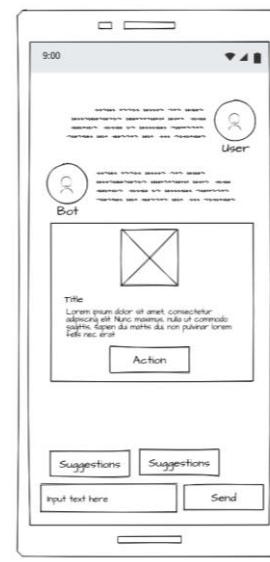
Gambar 24. Konsep *Simple Responses*



Gambar 25. Konsep *Media Responses*



Gambar 26. Konsep *List Responses*



Gambar 27. Konsep *Basic Card*

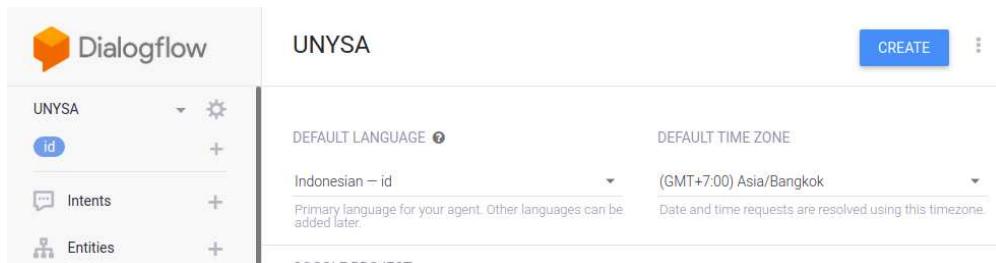
3. *Development*

Tahap pengembangan *chatbot* UNYSA menggunakan dialogflow sebagai komponen NLU. Selain itu, digunakan flask microframework (python) untuk mengelola data respon dengan yang menggunakan metode *web scraping* dan integrasi google sheet. Berikut penjabaran fase pengembangan *chatbot*.

a. Dialogflow

1) Pembuatan *Agent*

Agent pada dialogflow merupakan sebuah modul tempat seluruh pemrosesan bahasa alami dilakukan, selain itu *agent* juga dapat digunakan untuk mengelola alur percakapan berdasarkan hasil desain yang sudah dibuat. *Agent* melakukan pemrosesan *request* yang datang dari pengguna dan melakukan pemetaan ke dalam *intent*. Pembuatan *agent* pada dialogflow tersaji pada Gambar 28 berikut.



Gambar 28. Pembuatan *Agent*

2) Pembuatan *Entities*

Entities digunakan untuk mengidentifikasi makna dari pertanyaan yang dikirim pengguna. Sebagian besar jenis *entities* yang peneliti gunakan adalah *developer entities*. *Entities* akan dicocokan dengan *query* yang dikirim oleh pengguna, apakah teridentifikasi kesesuaian dengan kata kunci (*entity value*) pada *entities* yang dibuat seperti tentang akademik, pendaftaran, beasiswa, prodil UNY, dan sebagainya. Gambaran pembuatan *entities* tersaji pada Gambar 29 berikut.



Gambar 29. Pembuatan *Entities*

Di dalam *entities*, selain kata kunci (*entity value*) terdapat daftar sinonim supaya suatu kata kunci memiliki banyak variasi kata. Contoh pembuatan *entities* beserta sinonim yang menggunakan format *JSON (JavaScript Object Notation)* tersaji pada Gambar 30 berikut.

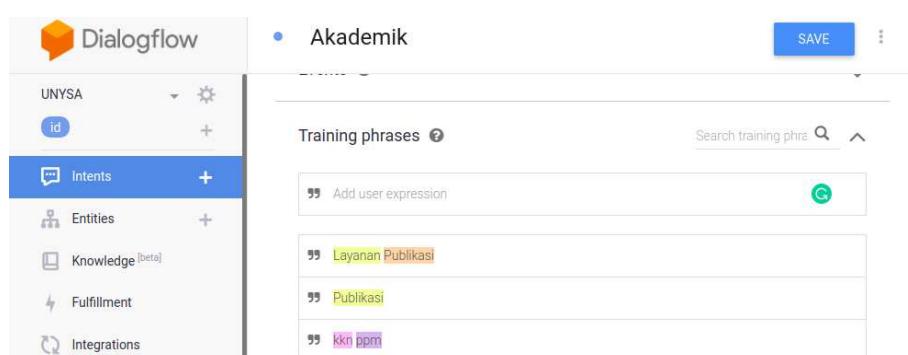
```
[  
  {  
    "value": "Akreditasi",  
    "synonyms": [  
      "Akreditasi"  
    ]  
  },  
  {  
    "value": "Animo",  
    "synonyms": [  
      "Animo"  
    ]  
  },  
]
```

Gambar 30. Contoh *Entities*

3) Pembuatan *Intent*

Intent digunakan untuk memetakan *query* dari pengguna supaya mendapatkan respon yang relevan, penulis membuat *intent* berdasarkan topik seperti akademik, program studi, dan sebagainya. Di dalam *intent* diberikan masukan *training phrases* yang merupakan perkiraan pertanyaan dari pengguna, hal ini juga berfungsi sebagai kamus atau inisialisasi *entities* apa saja yang akan digunakan dalam *intent* tersebut.

Contoh pembuatan *intent* pada dialogflow tersaji pada Gambar 31 berikut.



Gambar 31. Contoh Pembuatan *Intent*

4) Parameters

Setelah melakukan masukan terhadap *training phrases*, maka kebutuhan *parameters* akan disesuaikan dengan *entities value* apa saja yang terkandung dalam *training phrases*. Hal ini berarti pengguna harus memasukan beberapa kata kunci sesuai dengan kriteria *intent* tersebut supaya mendapatkan jawaban yang relevan seperti yang tersaji pada Gambar 32 berikut.

The screenshot shows the Dialogflow interface with the 'Akademik' intent selected. On the left sidebar, 'Intents' is highlighted. In the main area, the 'Action and parameters' section is displayed. It includes a table for defining parameters:

REQUIRED	PARAMETER NAME	ENTITY	VALUE	IS LIST
<input type="checkbox"/>	Akademik	@Akademik	\$Akademik	<input type="checkbox"/>
<input type="checkbox"/>	biaya	@biaya	\$biaya	<input type="checkbox"/>
<input type="checkbox"/>	pendaftaran	@pendaftaran	\$pendaftaran	<input type="checkbox"/>

Gambar 32. Contoh Pembuatan *Parameters*

5) Mengaktifkan *fulfillment responses*

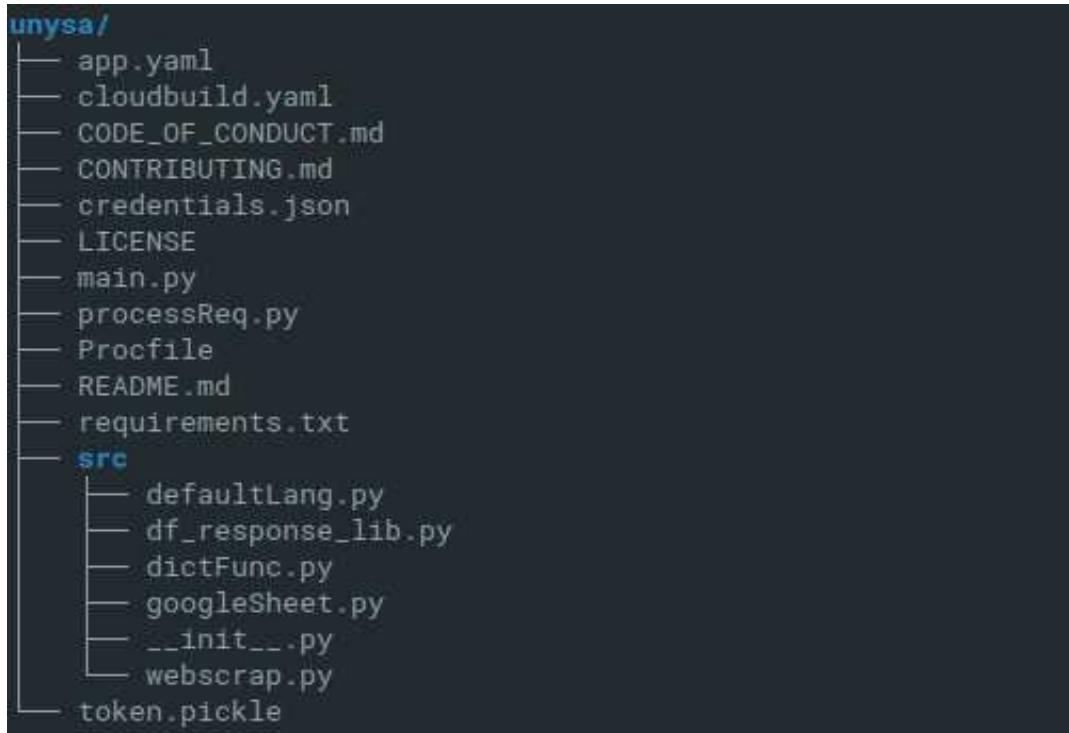
Fulfillment responses digunakan untuk melakukan panggilan terhadap program yang dibuat *developer* untuk mengolah logika dan mendapatkan data sebagai respon dari *agent*. Untuk mengaktifkan *fulfilment responses* pada *intents* dapat dilakukan seperti pada Gambar 33 berikut.

The screenshot shows the Dialogflow interface with the 'Akademik' intent selected. On the left sidebar, 'Fulfillment' is highlighted. In the main area, the 'Responses' section is shown. Under the 'DEFAULT' tab, there is a 'Text or SSML response' input field containing '1 Enter a text or SSML response'. Below it is a 'Fulfillment' section with a red box around the 'Enable webhook call for this intent' toggle switch, which is turned on.

Gambar 33. Mengaktifkan *Fulfillment*

b. *Fulfillment*

Penulis menggunakan flask *microframework* (python) untuk mengelola respon dari *agent* dengan melakukan pengambilan data melalui *web scraping* dan integrasi google sheet. Struktur kode disajikan pada Gambar 34 berikut.



```
unysa/
├── app.yaml
├── cloudbuild.yaml
├── CODE_OF_CONDUCT.md
├── CONTRIBUTING.md
├── credentials.json
├── LICENSE
├── main.py
├── processReq.py
├── Procfile
├── README.md
├── requirements.txt
└── src
    ├── defaultLang.py
    ├── df_response_lib.py
    ├── dictFunc.py
    ├── googleSheet.py
    ├── __init__.py
    └── webscrap.py
token.pickle
```

Gambar 34. Struktur Kode *Fulfillment Responses*

1) Potongan kode *webscrap.py*

Dalam *webscrap.py* terdapat modul *beautifulsoup* yang digunakan untuk pengambilan data dari *web* resmi UNY (*web scraping*). *Web scraping* ditujukan untuk data yang berbentuk *list* seperti daftar akreditasi program studi dan UKT.

Potongan kode *webscrap.py* tersaji pada Gambar 35 berikut.

```

for i in range(1,9):
    # URL of PDPT UNY
    urls = f'http://pdpt.uny.ac.id/dtakreditasi?page={i}'.split(" ")

    for i in urls:
        # Get content from url
        url = requests.get(i)

        # Parsing content
        soup = BeautifulSoup(url.content, 'html.parser')

        # Find all tag table in web
        table = soup.find_all('table')

        # Create table based on content in table
        df = pd.read_html(str(table))[0]

        # Get list in the table (output > multiple list)
        df_fakultas = df['Fakultas'].tolist()
        df_prodi = df['Prodi'].tolist()
        df_akreditasi = df['Akreditasi'].tolist()

```

Gambar 35. Potongan Kode *webscrap.py*

2) Potongan kode *googleSheet.py*

Berfungsi untuk integrasi data respon *manual* yang dibuat pada google sheet.

Pembuatan respon *manual* ditujukan untuk data yang sifatnya deskriptif, data tersebut membutuhkan rangkuman yang lebih singkat sesuai dengan aliran percakapan agar *chatbot* lebih humanis. Potongan program *googleSheet.py* tersaji pada Gambar 36 berikut.

```

worksheet = sh.worksheet("normal_respons")

# Create converter to lower text
def convert_low(list_data):
    result = []
    for element in list_data:
        result.append(element.lower())
    return result

# Take and convert value in google spreadsheet
gsheet_entity_intent = convert_low(worksheet.col_values(1))
gsheet_entity_type = convert_low(worksheet.col_values(2))
gsheet_entity_value = convert_low(worksheet.col_values(3))

```

Gambar 36. Potongan Kode *googleSheet.py*

3) Potongan kode *processReq.py*

Setelah diperoleh data dari berbagai sumber, kemudian dilakukan pengolahan data pada *processReq.py* dengan melakukan pemetaan terhadap *entity value* yang terkandung dalam *parameters* atau *query* dari *user*, kemudian dicocokan dengan respon pada sumber data yang tersedia. Potongan dari kode program *processReq.py* tersaji pada Gambar 37 berikut.

```
if 'Akreditasi' in entity_value:
    # Get index from PDPT
    index = scrap_pdpt.prodi.index(entity_value_prodi_jenjang)

    # Get respons based on value of web scrapping
    text = random.choice(AKREDITASI).format(
        prodi = scrap_pdpt.prodi[index],
        akreditasi = scrap_pdpt.akreditasi[index]
    )

    # Generate response
    speech = ff_response.fulfillment_text(text)
```

Gambar 37. Potongan Kode *processReq.py*

4) Potongan kode *Format Responses*

df_responses_lib.py digunakan mengimplementasikan desain respon *chatbot* pada *wireframing* yang sudah dibuat. Potongan kode *df_responses_lib.py* tersaji pada Gambar 38 berikut.

```
def fulfillment_text(self, fulfillmentText):
    if fulfillmentText == "":
        raise Exception("Fulfillment text should not be empty.")
    else:
        return {
            "fulfillment_text": str(fulfillmentText)
        }
```

Gambar 38. Potongan Kode *Format Responses*

5) Potongan kode *webhook*

Pada *main.py* digunakan untuk membuat proses pengiriman data yang sudah diolah untuk dikirimkan ke dialogflow melalui *webhook* dengan menggunakan format JSON. Potongan kode *main.py* tersaji pada Gambar 39 berikut.

```
def webhook():
    req = request.get_json(silent=True, force=True)
    res = processRequest(req)
    res = json.dumps(res, indent=4)
    r = make_response(res)
    r.headers['Content-Type'] = 'application/json'
    return r
```

Gambar 39. Potongan Kode *main.py*

6) Pengelolaan Materi Respon di Google Sheet

Materi untuk jawaban *chatbot* dapat dimasukan secara manual oleh kontributor melalui google sheet yang sudah disediakan dengan format kolom *intent*, *entity_type*, *entity_value*, dan *answer*. Beberapa contoh materi respon *chatbot* tersaji dalam Lampiran 13 yang digambarkan pada Gambar 40 berikut.

unsa_respons_database						
	A	B	C	D	E	F
	intent	entity_type	entity_value	answer1	answer2	answer3
2	Pendaftaran	pendaftaran	pendaftaran	Banyak cara mendaftar atau jalur seleksi di UNY, untuk diploma dan sarjana diantaranya ada SNMPTN, SBMPTN, dan Seleksi Mandiri. Sedangkan untuk pascasarjana ada SM Pendidikan jalin kualif dan penelitian, UTBK atau Ujian Tulis, dan Kerjasama.	Banyak cara mendaftar atau jalur seleksi di UNY, untuk diploma dan sarjana diantaranya ada SNMPTN, SBMPTN, dan Seleksi Mandiri. Sedangkan untuk pascasarjana ada SM Pendidikan jalin kualif dan penelitian, UTBK atau Ujian Tulis, dan Kerjasama.	Banyak cara mendaftar atau jalur seleksi di UNY, untuk diploma dan sarjana diantaranya ada SNMPTN, SBMPTN, dan Seleksi Mandiri. Sedangkan untuk pascasarjana ada SM Pendidikan jalin kualif dan penelitian, UTBK atau Ujian Tulis, dan Kerjasama.
3	Pendaftaran	jalur_seleksi	SNMPTN	SNMPTN (Seleksi Nasional Masuk Perguruan Tinggi Negeri) merupakan Penjaringan siswa unggul dari masing-masing sekolah di seluruh Indonesia yang memiliki kemampuan akademik tinggi. Dikoordinasikan oleh Lembaga Tes Masuk Perguruan Tinggi (LTMPT) 2019. Jadwal pelaksanaan dan proses seleksi mengikuti ketentuan nasional. Cek selengkapnya tentang SNMPTN di website berikut:	SNMPTN (Seleksi Nasional Masuk Perguruan Tinggi Negeri) merupakan Penjaringan siswa unggul dari masing-masing sekolah di seluruh Indonesia yang memiliki kemampuan akademik tinggi. Dikoordinasikan oleh Lembaga Tes Masuk Perguruan Tinggi (LTMPT) 2019. Jadwal pelaksanaan dan proses seleksi mengikuti ketentuan nasional. Cek selengkapnya tentang SNMPTN di website berikut:	SNMPTN (Seleksi Nasional Masuk Perguruan Tinggi Negeri) merupakan Penjaringan siswa unggul dari masing-masing sekolah di seluruh Indonesia yang memiliki kemampuan akademik tinggi. Dikoordinasikan oleh Lembaga Tes Masuk Perguruan Tinggi (LTMPT) 2019. Jadwal pelaksanaan dan proses seleksi mengikuti ketentuan nasional. Cek selengkapnya tentang SNMPTN di website berikut:

Gambar 40. Contoh Materi Respon *Chatbot*

4. Testing

Pada saat proses pengembangan perangkat lunak, peneliti melakukan pengecekan tentang penulisan kode program, fungsi, tampilan, dan respon *chatbot*.

Pengecekan kesalahan penulisan kode memanfaatkan fitur *debug* pada aplikasi visual studio code (vscode) yang akan menampilkan pesan kesalahan penulisan kode beserta nomor barisnya seperti yang tersaji dalam Gambar 41 berikut.

```

File Edit Selection View Go Debug Terminal Help
DEBUG No Corr processReq.py main.py
VARIABLES
Locals
AKREDITASI: ['Untuk p...']
BeautifulSoup: <class '...>
EXCEPTION_DEFAULT: ['M...']
FAKULTAS: ['Untuk pro...']
Flask: <class 'flask.a...>
GAGAL: ['Mohon maaf, m...']
RES_JENJANG: ['Baik, i...']
RES_PRODI: ['Baik, sil...']
ServiceAccountCredenti...
WATCH
main.py ...
1 # Import Flask module for request and respons to dialogflow
2 from flask import Flask, make_response, request
3
4 # Import local module processReq with all function
5 from processReq import *
6
7 # Define the place of webhook (module name)
8 # app = Flask(__name__)
9 @app.route('/webhook', methods=['GET','POST'])

Exception has occurred: NameError
name 'app' is not defined
File "/home/andri/mycode/unysa/main.py", line 9, in <module>
    @app.route('/webhook', methods=['GET','POST'])

```

Gambar 41. Debug Program di Vscode

Untuk pengujian *chatbot* dilaksanakan menggunakan enam karakteristik *software quality model* ISO/IEC 20510 yang meliputi *functional suitability*, *performance efficiency*, *portability*, *usability*, *reliability*, dan *maintainability*. Pengujian bertujuan untuk mengetahui dan memastikan kualitas *chatbot* telah memenuhi standar perangkat lunak.

a. *Functional Suitability*

Functional testing bertujuan untuk menjamin kelayakan setiap fungsi *chatbot* dapat berjalan dengan baik. Pengujian dilaksanakan oleh ahli pengembangan perangkat lunak berbasis *chatbot*, ahli materi (Humas UNY), dan pengguna akhir.

1) Ahli Pengembangan Perangkat Lunak

Pengujian fungsionalitas *chatbot* dilakukan oleh 2 ahli pengembangan perangkat lunak yang disajikan pada Tabel 31 berikut.

Tabel 31. Penguji Ahli Perangkat Lunak

No	Nama	Profesi	Instansi
1	Pradipta Setyahadi, M.Sc	IT Consultant	PT. Gamatechno Indonesia
2	Andhika Yuwana	Software Developer	Ajaro Indonesia

Terdapat tiga karakteristik pengujian fungsionalitas pada *chatbot*, yakni *conversation design*, *entities*, dan *fulfillment testing*. Pengujian dilakukan dengan memberikan 15 butir fungsi yang perlu diuji. Setiap fungsi yang berjalan dengan sesuai menurut penguji maka diberikan nilai 1, dan yang tidak berjalan dengan sesuai diberikan nilai 0. Hasil pengujian tersaji pada Tabel 32 berikut.

Tabel 32. Hasil Pengujian oleh Ahli Perangkat Lunak

Butir Pertanyaan	Skor Penguji			Skor Maksimal
	P1	P2	P-Total	
1	1	1	2	2
2	0	0	0	2
3	1	0	1	2
4	1	1	2	2
5	1	1	2	2
6	1	1	2	2
7	1	1	2	2
8	1	1	2	2
9	1	1	2	2
10	1	1	2	2
11	1	1	2	2
12	1	1	2	2
13	1	1	2	2
14	1	1	2	2
15	1	0	1	2
Total	14	12	26	30

Perhitungan hasil pengujian fungsional *chatbot* dilakukan dengan rumus berikut:

$$X = I/P$$

Keterangan:

$$X = 26/30$$

I = Jumlah fitur yang berhasil diterapkan

$$X = 0,86$$

P = Jumlah fitur yang dirancang

Hasil pengujian menunjukkan bahwa sebagian besar (0,86) fitur *chatbot* berhasil diimplementasikan dan berjalan dengan baik sesuai yang direncanakan. Sehingga *chatbot* dapat memenuhi syarat karakteristik *functional suitability*.

2) Validasi Materi

Validasi materi pada penelitian ini bertujuan untuk menilai kelayakan materi yang disajikan sebagai sumber informasi bagi masyarakat umum melalui *chatbot*. Validasi dilakukan oleh Bapak R.Dedy Herdito, S.E., M.M selaku staf Humas UNY. Penilaian dilaksanakan dengan cara meninjau ulang seluruh materi pada *chatbot* dan mencoba beberapa skenario pertanyaan, lalu dilakukan penilaian terhadap aspek pada kuesioner. Hasil validasi materi tersaji pada Tabel 33 berikut.

Tabel 33. Hasil Validasi Materi

Butir Pertanyaan	Skor Pengujii	Skor Maksimal	Persentase
1	5	5	100%
2	5	5	100%
3	5	5	100%
4	5	5	100%
5	5	5	100%
6	5	5	100%
7	5	5	100%
8	5	5	100%
9	5	5	100%
10	5	5	100%
11	5	5	100%
12	5	5	100%
13	5	5	100%
14	4	5	80%
15	5	5	100%
16	5	5	100%
17	5	5	100%
Total	84	85	

$$\begin{aligned}\text{Persentase Keseluruhan (\%)} &= (\text{skor perolehan skor maksimum}) \times 100\% \\ &= (84 \div 85) \times 100\% \\ &= 98,82\%\end{aligned}$$

Berdasarkan interpretasi nilai pada Tabel 14 maka hasil validasi materi masuk dalam kategori Sangat Layak untuk digunakan.

3) User Acceptance Test

UAT dilaksanakan dengan menguji secara langsung fungsi *chatbot* kepada pengguna terbatas dengan menggunakan skenario tertentu yang sudah disediakan oleh Alma Bot. Selain itu, alma bot juga berperan untuk mengakumulasikan hasil yang diterima dari pengguna dan mengklasifikasikannya menjadi 7 karakteristik tingkat kemampuan *chatbot* yakni *Personality*, *Onboarding*, *Navigation*, *Understanding*, *Answering*, *Error Management*, dan *Intelligence*.

Berdasarkan hasil UAT yang dilakukan oleh 21 mahasiswa UNY dari 7 fakultas dalam Lampiran 24, selanjutnya nilai diinterpretasikan sesuai dengan Tabel 14 tentang kelayakan terhadap seluruh karakteristik *chatbot* disajikan pada Tabel 34 berikut.

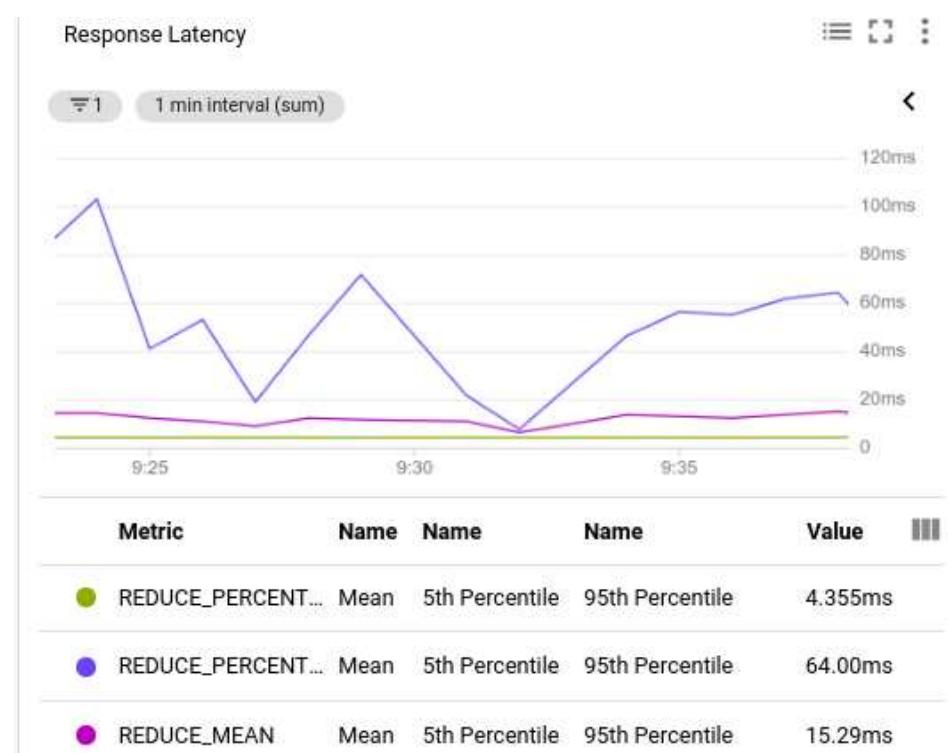
Tabel 34. Interpretasi Nilai Pengujian UAT

Karakteristik	Persentase Hasil	Keterangan
<i>Personality</i>	99.05%	Sangat Baik
<i>Onboarding</i>	100.00%	Sangat Baik
<i>Navigation</i>	78.10%	Baik
<i>Understanding</i>	62.86%	Baik
<i>Answering</i>	87.62%	Sangat Baik
<i>Error Management</i>	85.71%	Sangat Baik
<i>Intelligence</i>	74.29%	Baik
Rata-rata	83,95%	Sangat Baik

Berdasarkan Tabel 34 *chatbot* menghasilkan nilai rata-rata 83,95% dengan kategori Sangat Baik. Namun *chatbot* masih memerlukan pengembangan lebih lanjut untuk meningkatkan seluruh karakteristik *chatbot* khususnya pada karakteristik *navigation*, *understanding*, dan *intelligence*.

b. *Performance Efficiency*

Karakteristik *performance efficiency* dianalisis menggunakan Google Stackdriver saat pengujian *reliability* berlangsung dengan menggunakan 395x2 pertanyaan yang dilaksanakan selama 15 menit. Hasil pengujian *performance efficiency* terhadap kecepatan respon *chatbot* (*response latency*) tersaji pada Gambar 42 berikut.

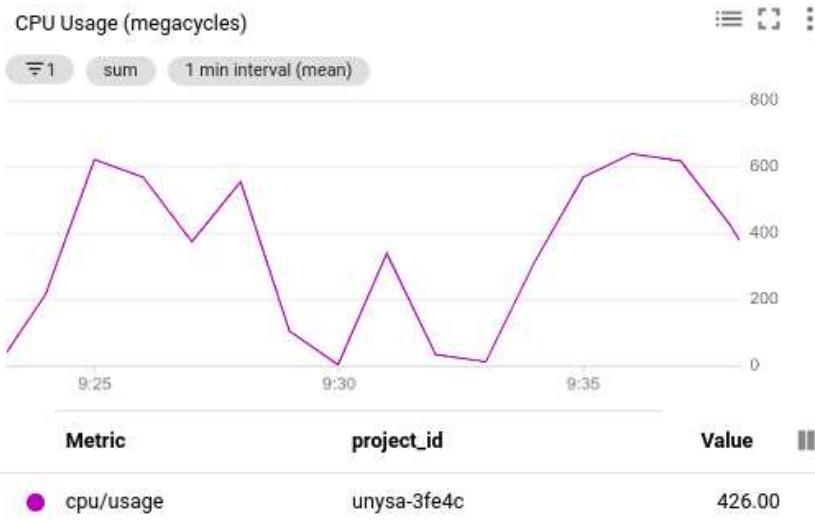


Gambar 42. Grafik *Response Latency*

Hasil pada Gambar 42 menunjukan bahwa waktu respon *chatbot* (*response latency*) rata-rata sebesar 15 ms. Artinya lebih cepat dibandingkan waktu standar (2 detik). Sedangkan untuk penggunaan memori pada app engine rata-rata sebesar 509 MiB (Gambar 43) dan CPU rata-rata sebesar 426 *megacycles* (Gambar 44) dengan kapasitas maksimal yang dapat di *scale up* otomatis sesuai kebutuhan penggunaan. Sehingga sumber daya memenuhi kelayakan performa.



Gambar 43. Penggunaan Memori Pada App Engine



Gambar 44. Penggunaan CPU Pada App Engine

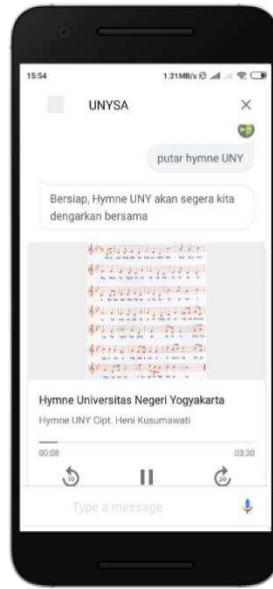
c. *Portability*

Pengujian karakteristik *portability* dilakukan bersama dengan pengujian fungsionalitas dengan menjalankan *chatbot* pada *frontend apps* yang sudah direncanakan, yakni Google Assistant, WhatsApp, Telegram, Line, dan Facebook Messenger. Hasil pengujian adalah 100% seluruh *frontend apps* yang sudah direncanakan berhasil menjalankan fungsionalitas *chatbot*.

Berikut adalah tampilan *chatbot* UNYSA yang diimplementasikan ke berbagai macam *frontend apps* yang sudah direncanakan. Konsep *wireframing* seluruhnya dapat diimplementasikan pada antarmuka google assistant seperti yang tersaji dalam Gambar 45 hingga 48 berikut.



Gambar 45. *Simple Responses*



Gambar 46. *Media Responses*

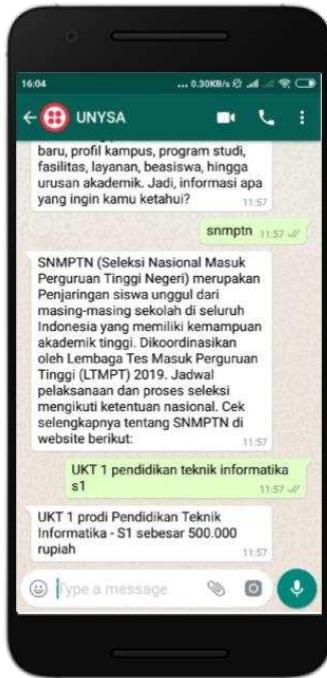


Gambar 47. *List Responses*



Gambar 48. *Basic Card Responses*

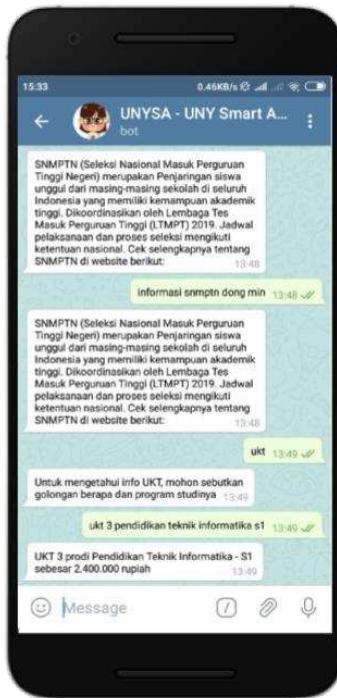
Sedangkan pada penelitian ini, penulis membuat *simple responses* untuk seluruh *frontend apps* seperti yang tersaji dalam Gambar 49 hingga 52 berikut.



Gambar 49. Tampilan pada WhatsApp



Gambar 50. Tampilan pada Messenger



Gambar 51. Tampilan pada Telegram



Gambar 52. Tampilan pada Line

d. *Usability*

Pengujian *usability* pada *chatbot* dilakukan setelah pengguna melaksanakan UAT agar mengenal fungsi dan penggunaan *chatbot*. Karakteristik *usability* pada *chatbot* UNYSA diuji dengan *USE Questionnaire* menggunakan likert skala 5 poin dengan jumlah 30 butir pertanyaan. Kuesioner diisi oleh 21 responden dari 7 fakultas di Universitas Negeri Yogyakarta. Hasil pengujian *usability* menggunakan *USE Questionnaire* disajikan pada Lampiran 25. Berdasarkan data hasil tersebut, sebaran total jawaban dapat dikelompokan menjadi sangat setuju (SS) berjumlah 210, setuju (S) berjumlah 340, netral (N) berjumlah 71, tidak setuju (TS) berjumlah 8, dan sangat tidak setuju (STS) berjumlah 1. Data tersebut kemudian dihitung menggunakan rumus berikut.

$$\begin{aligned}\text{Skor}_{\text{total}} &= (J_{\text{SS}} \times 5) + (J_{\text{S}} \times 4) + (J_{\text{N}} \times 3) + (J_{\text{TS}} \times 2) + (J_{\text{STS}} \times 1) \\ &= (210 \times 5) + (340 \times 4) + (71 \times 3) + (8 \times 2) + (1 \times 1) \\ &= 2640\end{aligned}$$

Untuk mendapatkan persentase *usability* dihitung menggunakan rumus berikut:

$$\begin{aligned}\text{Persentase} &= \frac{2640}{3150} \times 100\% \\ &= 83,81\%\end{aligned}$$

Hasil perhitungan persentase pengujian *usability* menghasilkan 83,81%.

Menurut klasifikasi pada Tabel 16, maka dapat dinyatakan bahwa UNYSA Sangat Layak dan memenuhi kriteria pada aspek *usability*. Mengukur nilai reliabilitas hasil pengujian *usability* dilakukan dengan perhitungan *Alpha Cronbach* menggunakan *software* PSPP. Hasil perhitungan nilai konsistensi tersaji pada Gambar 53 berikut.

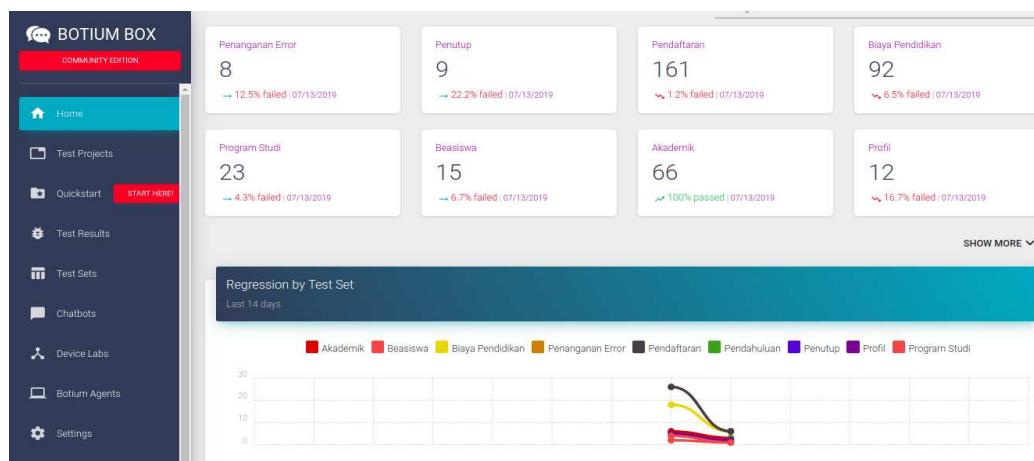
Case Processing Summary			Reliability Statistics		
Cases	Valid	N	%	Cronbach's Alpha	N of Items
Cases	Valid	21	100,00		
	Excluded	0	,00		
Total		21	100,00	,92	30

Gambar 53. Hasil Perhitungan *Alpha Cronbach*

Gambar 53 menunjukkan bahwa pengujian terhadap 30 butir pernyataan *usability* menghasilkan nilai *Cronbach's Alpha* 0,92. Berdasarkan Tabel 17 maka instrumen penelitian termasuk dalam kategori *Excellent*. Seluruh *item* dalam instrumen *usability* adalah reliabel atau konsisten karena nilai *Cronbach's Alpha* 0,92 melebihi batas standar (0,60) dan melebihi nilai r tabel (0,361 dengan signifikansi 5%).

e. *Reliability*

Pengujian karakteristik *reliability* menggunakan *software* bernama botium untuk melakukan *end to end testing* secara otomatis terhadap skenario percakapan yang sudah direncanakan. Tampilan dari *software* pengujian *chatbot* bernama botium tersaji pada Gambar 54 berikut.



Gambar 54. Tampilan Botium

Pengujian dilakukan sebanyak 10 kali untuk menghasilkan nilai yang optimal. Terbukti pada grafik regresi yang disajikan pada Gambar 55 menunjukan *chatbot* dapat menjadi lebih baik pada setiap kali pengujian, hal ini dikarenakan secara tidak langsung akan memberikan pelatihan pada *chatbot* untuk jawaban yang belum dapat direspon secara optimal.



Gambar 55. Regresi *Datasets*

Dalam pengujian diperlukan *datasets* yang berisi 395 skenario percakapan, terdiri dari beberapa variasi pertanyaan dengan jawaban yang diharapkan. Kategori pertanyaan dalam skenario meliputi pendahuluan, profil, akademik, beasiswa, program studi, biaya pendidikan, pendaftaran, penutup, dan penanganan kesalahan. Dalam proses pengujian, botium memvalidasi antara jawaban *chatbot* dengan yang diharapkan dalam *datasets*. Jika tidak sesuai maka botium akan menganggap bahwa skenario tersebut gagal dilaksanakan *chatbot*.

Dari seluruh hasil pengujian skenario pada Tabel 35 menunjukan bahwa nilai rata-rata keberhasilan *chatbot* sebesar 92%. Data tersebut menunjukan bahwa sebagian besar skenario percakapan yang sudah dirancang berhasil direspon dan dipahami oleh *chatbot*.

Tabel 35. Hasil Pengujian *Reliability*

Skenario (<i>Intent</i>)	Jumlah Pertanyaan	Pengujian Ke-										Rata- Rata
		1	2	3	4	5	6	7	8	9	10	
Pendahuluan	9	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%	100%
Profil	12	33%	92%	100%	100%	100%	92%	83%	92%	100%	83%	88%
Akademik	66	95%	95%	95%	98%	100%	98%	100%	100%	98%	100%	98%
Beasiswa	15	93%	93%	93%	87%	93%	93%	93%	93%	93%	93%	93%
Program Studi	23	87%	91%	96%	91%	91%	91%	91%	100%	96%	96%	93%
Biaya Pendidikan	92	96%	96%	96%	93%	97%	96%	95%	97%	95%	93%	95%
Pendaftaran	161	97%	96%	97%	99%	99%	99%	99%	100%	99%	99%	98%
Penutup	9	78%	78%	78%	78%	78%	78%	78%	78%	78%	78%	78%
Error Handling	8	88%	88%	88%	88%	88%	88%	88%	88%	88%	88%	88%
Total (Rata-Rata)											92%	

f. *Maintainability*

Karakteristik *maintainability* pada *chatbot* diuji dengan menggunakan *software Radon* yang khusus digunakan untuk bahasa pemrograman Python. Pengujian *Maintainability Index (MI)* pada Radon menggunakan perintah berikut:

\$radon mi unysa/* -s. Hasil pengujian MI tersaji pada Tabel 36 berikut.

Tabel 36. Hasil pengujian MI

<i>Source Code</i>	<i>MI Score</i>	<i>Rank</i>	<i>Maintainability</i>
unysa/main.py	98.74	A	Very high
unysa/processReq.py	70.06	A	Very high
unysa/src/defaultLang.py	100.00	A	Very high
unysa/src/googleSheet.py	100.00	A	Very high
unysa/src/df_response_lib.py	50.79	A	Very high
unysa/src/dictFunc.py	81.41	A	Very high
unysa/src/webscrap.py	81.44	A	Very high
Rata-Rata	83.21	A	Very high

Hasil pada Tabel 36 menunjukan bahwa seluruh *source code* berhasil memenuhi standar kriteria *MI*. Sedangkan untuk pengujian diuji menggunakan perintah berikut: `$radon cc unysa/* -s --total-average`. Hasil pengujian tersaji pada Gambar 56 berikut.

```
36 blocks (classes, functions, methods) analyzed.  
Average complexity: A (3.25)
```

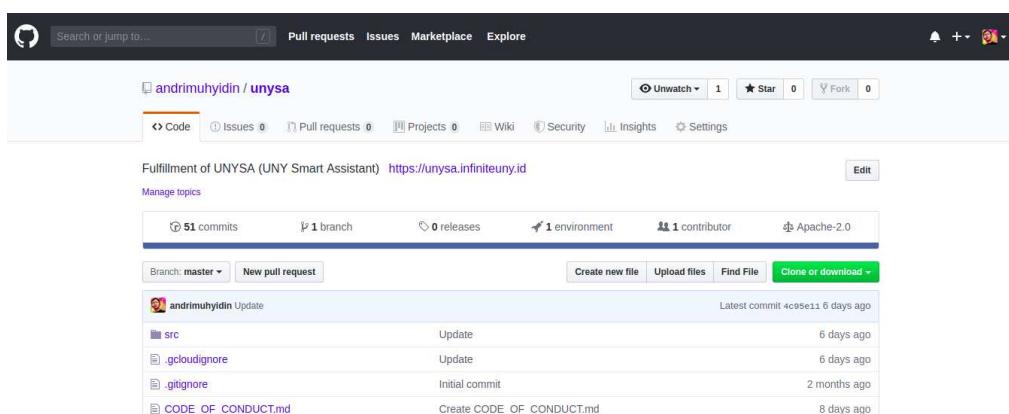
Gambar 56. Hasil pengujian CC

Berdasarkan skor pengujian CC yang mendapat nilai rata-rata keseluruhan sebesar 3,25 (Peringkat A). Maka dapat disimpulkan bahwa *chatbot* memiliki tingkat risiko yang rendah (*low*) dengan blok sederhana. Sehingga hasil pengujian MI dan CC menunjukan bahwa *chatbot* UNYSA memenuhi standar kriteria *Maintainability*.

5. Deployment

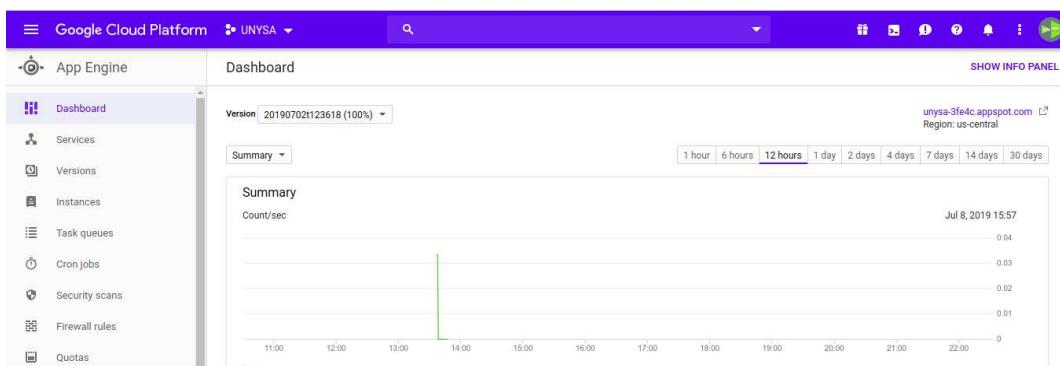
Pada tahap *deployment*, peneliti melakukan instalasi *fulfillment code* yang berada dalam repositori *github* (Gambar 57) untuk diintegrasikan pada google cloud: app engine. Hal ini dikarenakan mendukung model pengembangan agile dengan fitur CI/CD. Kode program dapat diakses melalui tautan berikut:

[https://github.com/andrimuhyidin/unysa.](https://github.com/andrimuhyidin/unysa)



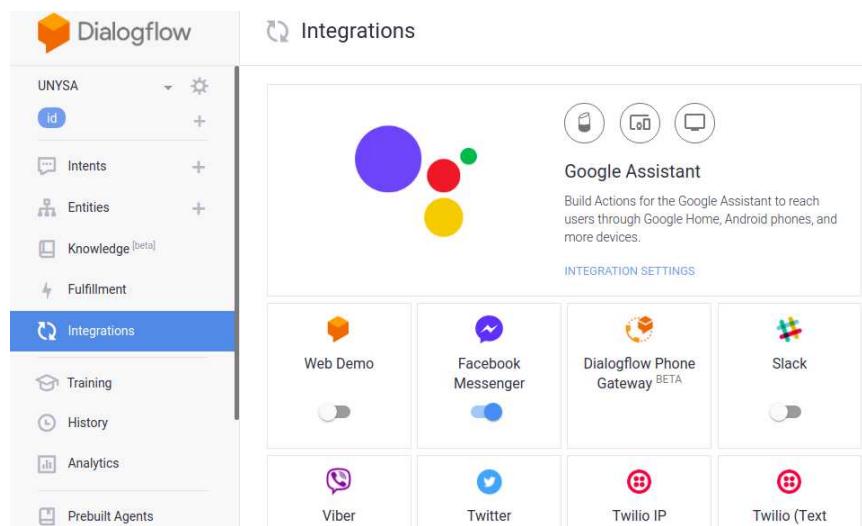
Gambar 57. Fulfillment Code di Github

Dengan terpasangnya *fulfillment code* pada app engine, maka *chatbot* siap untuk digunakan baik untuk uji coba maupun rilis secara luas. Jika terdapat perbaikan *code* atau penambahan data, maka secara otomatis *server* akan melakukan *update*. Sedangkan untuk rilis dan sosialisasi aplikasi dilakukan terbatas saat melakukan *User Acceptance Test* dan *Usability Testing*. Contoh tampilan app engine tersaji pada Gambar 58 berikut.



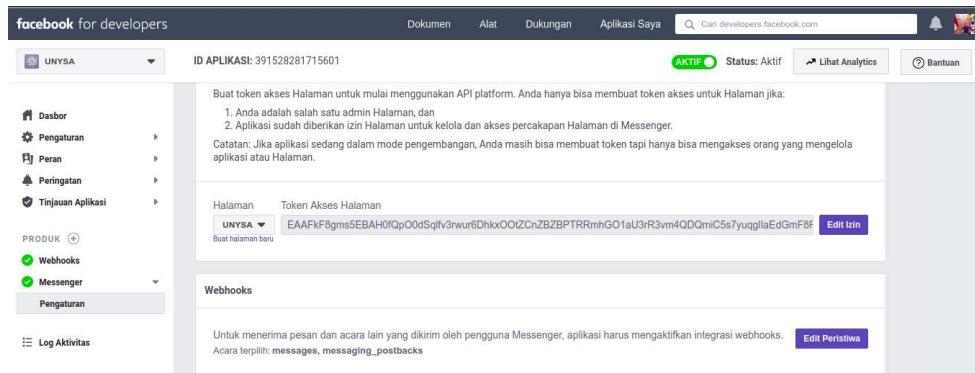
Gambar 58. Tampilan Google Cloud: App Engine

Langkah selanjutnya adalah melakukan integrasi antara dialogflow dan *frontend apps*. Integrasi berfungsi supaya pengguna dapat menghubungi *agent* atau *bot* menggunakan *frontend apps* yang didukung. Integrasi pada dialogflow disajikan pada Gambar 59 berikut.

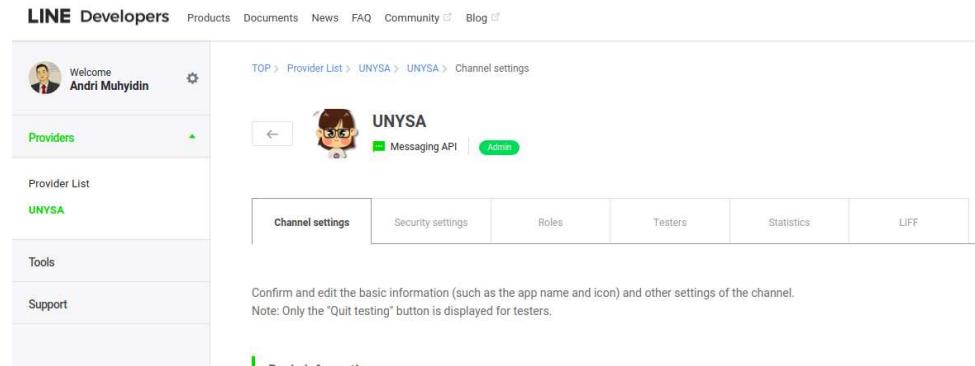


Gambar 59. Integrasi pada *Dialogflow*

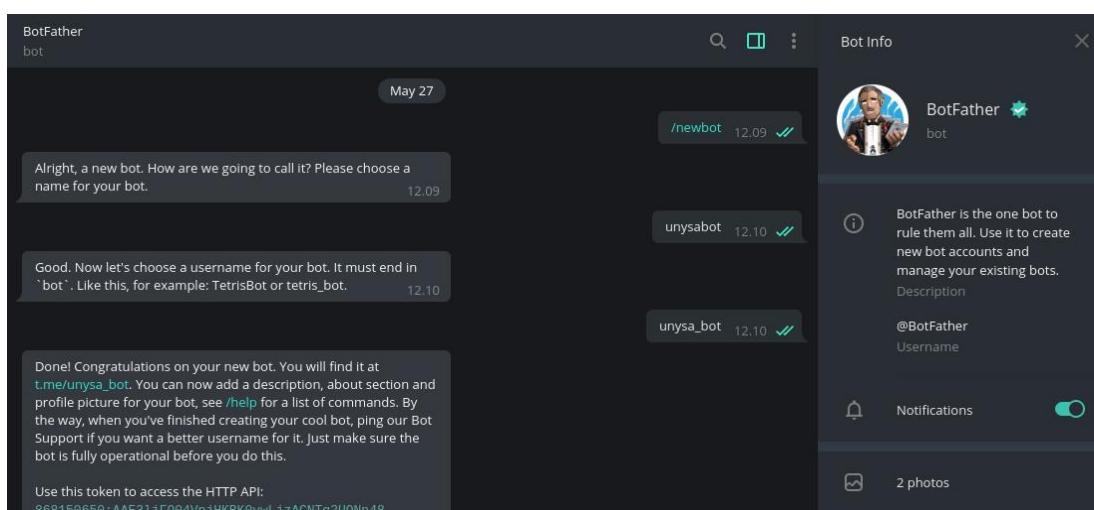
Frontend Apps harus dilakukan konfigurasi untuk mendapatkan *API Service* yang akan digunakan untuk menghubungkan antara *agent* pada dialogflow dengan pengguna melalui *frontend apps*. Hasil konfigurasi setiap *frontend apps* disajikan pada Gambar 60 hingga 64 berikut.



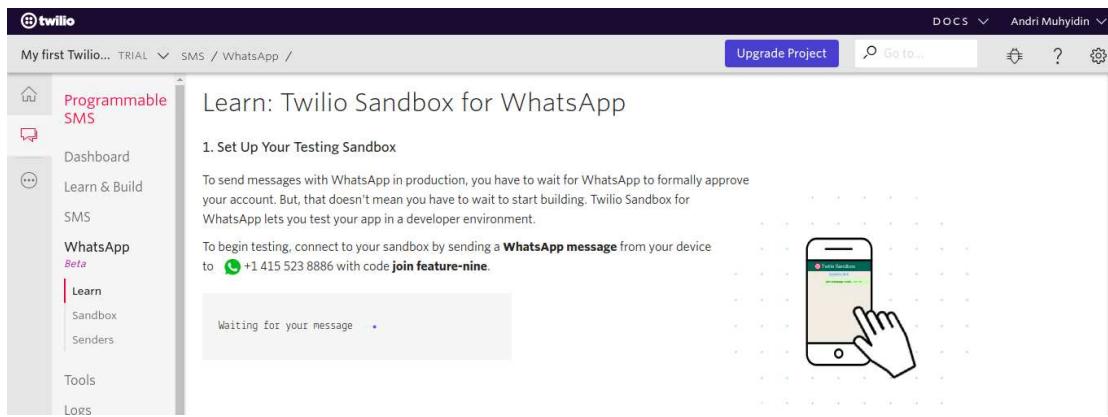
Gambar 60. Hasil Konfigurasi API Service pada Facebook Messenger



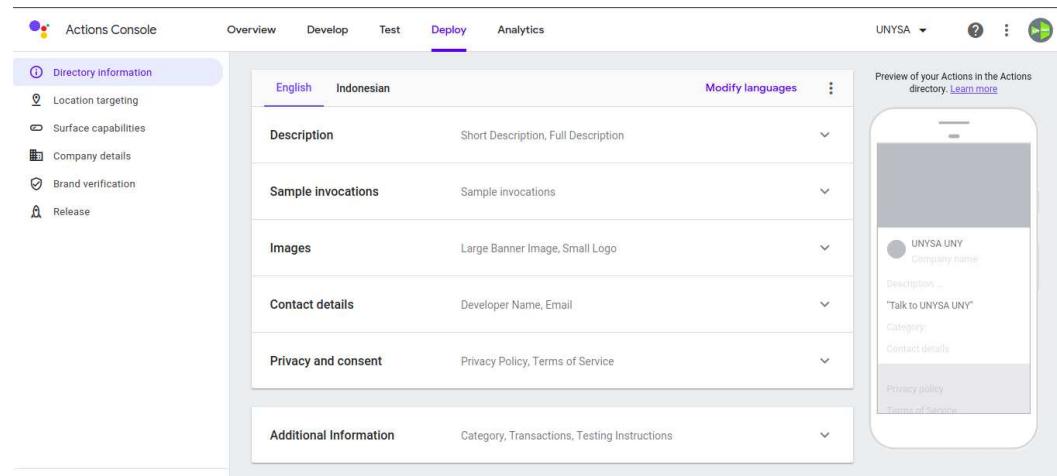
Gambar 61. Hasil Konfigurasi API Service pada Line



Gambar 62. Hasil Konfigurasi API Service pada Telegram



Gambar 63. Hasil Konfigurasi API Service pada Twilio (WhatsApp)



Gambar 64. Hasil Konfigurasi API Service pada Google Assistant

B. Pembahasan

1. Pembahasan Produk

Beberapa fitur dan kemampuan *chatbot* yang berhasil dikembangkan untuk mengatasi permasalahan dalam penelitian ini meliputi:

- Chatbot* dapat beroperasi secara *realtime*, sehingga berpeluang membantu humas dalam meningkatkan pelayanan publik terutama untuk membantu menjawab pertanyaan yang sama secara iteratif dan menyediakan layanan informasi secara *realtime*.

- b. *Chatbot* dapat diakses melalui berbagai macam *frontend apps* seperti WhatsApp, Telegram, Line, Facebook Messenger, dan Google Assistant. Hal ini dapat memudahkan publik untuk bertanya informasi seputar kampus UNY dari berbagai media komunikasi yang dimiliki.
- c. *Chatbot* memiliki kemampuan manajemen *error*, navigasi, serta pemahaman terhadap konteks pertanyaan yang diberikan pengguna. Sehingga pengguna dapat bertanya menggunakan bahasa alami manusia. Karena pada dasarnya setiap pengguna memiliki jenis atau variasi pertanyaan yang berbeda tetapi dalam konteks yang sama.
- d. *Chatbot* dapat melakukan percakapan ringan (*small talk*). Hal ini bertujuan untuk memberikan kesan humanis atau ramah terhadap pengguna.
- e. *Chatbot* mendukung beragam jenis respon yang memanfaatkan penggunaan media gambar dan suara. Sehingga dapat memberikan respon yang lebih interaktif kepada pengguna. Namun untuk saat ini konten video belum dapat ditampilkan secara langsung pada *frontend apps*, melainkan berupa *link* sumber video tersebut.
- f. Konten *chatbot* meliputi informasi umum seputar perguruan tinggi (UNY) seperti tentang pendaftaran, biaya pendidikan, akademik, beasiswa, profil, dan program studi. Konten tersebut bersumber pada *website* resmi UNY dan penggunaan google form untuk respon yang diisi oleh kontributor.
- g. *Chatbot* dapat diberikan pelatihan supaya lebih memahami pertanyaan pengguna, sehingga respon yang dihasilkan lebih relevan.

2. Pembahasan Hasil Pengujian

Berikut pembahasan hasil pengujian *chatbot* pada enam karakteristik ISO/IEC 20510 yang meliputi *functional suitability*, *performance efficiency*, *portability*, *usability*, *reliability*, dan *maintainability*.

a. Functional Suitability

- 1) Ahli pengembang perangkat lunak (*chatbot*)

Terdapat tiga aspek yang diuji untuk memenuhi karakteristik *functional suitability* pada *chatbot* yakni *conversation design*, *entities*, dan *fulfillment testing*.

Hasil pengujian menunjukkan bahwa nilai X berhasil mendekati 1 dengan memperoleh nilai 0,86. Angka tersebut menunjukkan bahwa sebagian besar fitur berhasil diterapkan dan berjalan dengan benar sesuai rancangan, sehingga *chatbot* memenuhi syarat *functional suitability*. Untuk beberapa fungsi yang tidak berjalan memerlukan perbaikan dan tambahan data yang dapat dilaksanakan pada penelitian selanjutnya sesuai dengan saran dari penguji berikut ini.

- a) Pradiptya Setyahadi, M.Sc menyatakan saran bahwa “Kedepan bisa diintegrasikan dengan sistem yang ada di UNY, agar proses pendaftaran mahasiswa, jadwal kuliah, pemilihan mata kuliah, pendaftaran beasiswa, pendaftaran sidang skripsi.”
- b) Sedangkan menurut Andhika Yuwana menyatakan saran bahwa “Perlu tambahan data untuk *training* yang lebih (misal hal-hal sederhana, seperti *small talk*), harapannya *bot* dapat lebih mengerti dengan baik. Selain itu, untuk perangkat *deployment* (Google App Engine) mungkin bisa dipindahkan ke yang berbayar jika sudah masuk di *production*, karena jika menggunakan

free ada kondisi *sleep* yang mana itu menyebabkan *delay* dalam membalas suatu *chat*.”

2) Validasi Materi

Validasi materi dalam penelitian dilakukan oleh staf humas UNY. Hasil validasi mendapatkan nilai sebesar 98,82%. Jika diinterpretasikan dalam kategori kelayakan menurut Tabel 14 maka konten atau materi pada *chatbot* UNYSA Sangat Layak untuk digunakan. Humas UNY berharap untuk penelitian lanjutan dapat diintegrasikan dengan sistem internal UNY.

3) *User Acceptance Test*

Pengujian UAT dilaksanakan dengan bantuan *software* Alma Bot yang menyediakan skenario pengujian dan melakukan kalkulasi nilai karakteristik *chatbot* dari hasil pengujian yang dilakukan oleh pengguna. Alma bot mengukur 7 kategori karakteristik *chatbot* dengan hasil nilai rata-rata 83,95% yang masuk dalam kategori Sangat Baik.

Berdasarkan seluruh pengujian yang dilaksanakan, dapat disimpulkan bahwa *chatbot* UNYSA berhasil memenuhi standar karakteristik *functional suitability*.

b. *Performance Efficiency*

Pengujian *performance efficiency* dilakukan dengan memberikan 395x2 skenario dengan waktu pengujian selama 15 menit menghasilkan nilai rata-rata waktu respon *chatbot* (*response latency*) sebesar 15 ms yang artinya lebih cepat dibanding waktu respon standar (2 detik). Sedangkan untuk sumber daya pada *chatbot* didapatkan informasi bahwa rata-rata penggunaan *memory* sebesar 509 MiB dan CPU sebesar 426 *megacycles* dengan kapasitas app engine yang dapat

dilakukan *scale up* secara otomatis. Sehingga dapat disimpulkan bahwa *chatbot* UNYSA berhasil memenuhi standar karakteristik *performance efficiency*.

c. *Portability*

Hasil pengujian menunjukan bahwa seluruh (100%) *frontend apps* yakni Google Assistant, WhatsApp, Telegram, Line, dan FB Messenger dapat menjalankan *chatbot* dengan baik. Sehingga dapat disimpulkan bahwa *chatbot* UNYSA berhasil memenuhi standar karakteristik *portability*.

d. *Usability*

Hasil pengujian *usability* menggunakan *USE Questionnaire* mendapatkan nilai 83,81% dengan kategori Sangat Layak dan memenuhi standar kriteria *usability*. Sedangkan untuk instrumen pengujian *usability* mendapatkan nilai *cronbach's alpha* 0,92 dengan kategori *Excellent*. Nilai tersebut melebihi batas standar 0,60 dan r tabel 0,361(signifikansi 5%), maka instrumen *usability* dapat dikatakan reliabel atau konsisten. Berdasarkan hasil pengujian tersebut, dapat disimpulkan bahwa *chatbot* UNYSA berhasil memenuhi standar karakteristik *usability*.

e. *Reliability*

Pengujian *reliability* dilaksanakan dengan bantuan *software* botium dengan memberikan 395 skenario pengujian. Pengujian dilakukan sebanyak 10 kali dan menghasilkan nilai rata-rata 92%. Jika nilai diinterpretasikan menurut Tabel 18 maka masuk dalam kategori Sangat Layak. Sehingga dapat disimpulkan bahwa *chatbot* UNYSA berhasil memenuhi standar karakteristik *reliability*.

f. *Maintainability*

Pengujian *maintainability* menghasilkan nilai rata-rata *Maintainability Index* (MI) sebesar 83,21 (A) yang artinya sangat mudah dirawat. Sedangkan pengujian *Cyclomatic Complexity* (CC) yang terdiri dari 36 blok (*classes, functions, methods*) menghasilkan nilai rata-rata sebesar 3,25 (A) yang masuk dalam kategori *low risk program*. Sehingga dapat disimpulkan bahwa *chatbot* UNYSA berhasil memenuhi standar karakteristik *maintainability*.