**IMPLEMENTASI CHATBOT DENGAN MEMANFAATKAN API CHATGPT MELALUI GRADIO APP MENGGUNAKAN METODE NATURAL LANGUAGE PROCESSING (NLP)**

SKRIPSI



**OLEH:**

**MUHAMMAD IRGI AL GHITHRAF**

**201011400874**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS PAMULANG**

**TANGERANG SELATAN**

**2023**

**IMPLEMENTASI CHATBOT DENGAN MEMANFAATKAN API CHATGPT MELALUI GRADIO APP MENGGUNAKAN METODE NATURAL LANGUAGE PROCESSING (NLP)**

**SKRIPSI**

Diajukan Untuk Melengkapi Salah Satu Syarat

Memperoleh Gelar Sarjana Komputer



**OLEH:**

**MUHAMMAD IRGI AL GHITHRAF**

**201011400874**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS ILMU KOMPUTER**

**UNIVERSITAS PAMULANG**

**TANGERANG SELATAN**

**2023/2024**

## 

## *ABSTRACT*

*Rapid advances in artificial intelligence and natural language processing (NLP) have opened the door for the development of more sophisticated and interactive chatbots. In this project, the author proposes implementing a chatbot by utilizing the ChatGPT API from OpenAI, via the Gradio app, with an NLP approach. The goal is to build a responsive and intuitive interface that allows users to communicate with the system using natural language. Using the ChatGPT API involves sending a text request from the user and receiving a text response from the API. However, the response may require further processing using NLP techniques. This process includes cleaning up the text, removing special characters, and applying entity processing to produce a more structured and understandable response. In addition, the NLP approach is also applied to recognize the broader context of conversation. This can be achieved through analyzing the history of past conversations or by recognizing patterns in user input. Thus, chatbots can provide more relevant and contextual responses. The end result of this project is a Gradio app that allows users to naturally interact with chatbots. The integration between Gradio, the ChatGPT API and NLP methods ensures a comfortable and responsive interface. The success of this implementation will contribute to the development of chatbots that are more efficient and able to answer the needs of text-based communication in various situations.*

*Keywords: Chatbot, ChatGPT API, Gradio app, Natural Language Processing (NLP) and artificial intelligence.*

## ABSTRAK

Peningkatan pesat dalam kecerdasan buatan dan pemrosesan bahasa alami (*NLP*) telah membuka pintu bagi pengembangan *chatbot* yang lebih canggih dan interaktif. Dalam proyek ini, penulis mengusulkan implementasi *chatbot* dengan memanfaatkan *API ChatGPT* dari *OpenAI*, melalui *Gradio app*, dengan pendekatan *NLP*. Tujuannya adalah membangun antarmuka yang responsif dan intuitif yang memungkinkan pengguna berkomunikasi dengan sistem menggunakan bahasa alami. Penggunaan *API ChatGPT* melibatkan mengirimkan permintaan teks dari pengguna dan menerima respons teks dari *API*. Namun, respons tersebut mungkin memerlukan pemrosesan lebih lanjut menggunakan teknik-teknik *NLP*. Proses ini mencakup pembersihan teks, penghilangan karakter khusus, dan penerapan pemrosesan entitas untuk menghasilkan respons yang lebih terstruktur dan mudah dipahami. Selain itu, pendekatan *NLP* juga diterapkan untuk mengenali konteks percakapan yang lebih luas. Ini dapat dicapai melalui analisis histori percakapan sebelumnya atau dengan mengenali pola dalam masukan pengguna. Dengan demikian, *chatbot* dapat memberikan respons yang lebih relevan dan kontekstual. Hasil akhir dari proyek ini adalah *Gradio app* yang memungkinkan pengguna untuk berinteraksi dengan *chatbot* secara alami. Integrasi antara *Gradio, API ChatGPT,* dan metode *NLP* memastikan antarmuka yang nyaman dan responsif. Keberhasilan implementasi ini akan memberikan kontribusi pada perkembangan *chatbot* yang lebih efisien dan mampu menjawab kebutuhan komunikasi berbasis teks dalam berbagai situasi.

Kata Kunci: *Chatbot, API ChatGPT, Gradio app, Natural Language Processing (NLP)* dan kecerdasan buatan.

## KATA PENGANTAR

Assalamu’alaikum warahmatullahi wabarakatuh

Alhadulillah, Puji syukur panjatkan kehadirat Allah Yang Maha Esa, atas berkat, Kesehatan, Rahmat dan Karunia-Nya, Penulis dapat menyelesaikan skripsi ini dengan sebaik-baiknya dan di berikan kemudahan dalam pengerjaannya dengan judul “Implementasi *Chatbot* dengan Memanfaatkan *API ChatGPT* melalui *Gradio App* Menggunakan Metode *Natural Language Processing (NLP)*” ini dapat diselesaikan. Penulisan ini dilaksanakan penulis sebagai salah satu syarat kelulusan Strata Satu (S1) pada Program Studi Teknik Informatika Universitas Pamulang, Tangerang Selatan.

Dalam penyelesaian Skripsi ini, Penulis telah berusaha dengan sebaik mungkin dan telah berusaha dengan kemampuan penulis dan tentunya tidak lupa pula dengan bantuan seluruh pihak yang telah meluangkan waktunya untuk membantu kelancaran penulisan Skripsi ini. Maka dalam kesempatan ini Penulis memberikan ucapan terima kasih yang sebesar-besarnya kepada:

1. Allah SWT yang telah memberikan begitu banyak nikmat dan karunia diantaranya Iman dan Islam serta sehat dan umur panjang sehingga dapat menyelesaikan skripsi ini.
2. Bapak Dr. Pranoto, MM Selaku Ketua Yayasan Sasmita Jaya.
3. Bapak Dr. Drs. E. Nurzaman AM., M.M., M.Si., selaku Rektor Universitas Pamulang.
4. Bapak Dr. Ir. H. Sarwani, M.T., M.M, selaku Dekan Fakultas Ilmu Komputer Universitas Pamulang.
5. Bapak Achmad Udin Zailani, S. Kom., M. Kom., selaku Ketua Program Studi Teknik Informatika Universitas Pamulang.
6. Ibu Dr. Thoyyibah T, S. Kom., M. Kom, selaku Pembimbing Skripsi Pada Program Studi Teknik Informatika Univertas Pamulang.
7. Kedua Orang Tua dan Adik saya tercinta yang telah memberikan doa, kasih sayang, semangat, juga motivasi kepada penulis untuk berbagi hal dan khususnya dalam penyusunan dalam penyusunan skripsi ini.
8. Teman – teman seperjuangan kelas 07TPLP008 Universitas Pamulang yang telah memberikan dukungandan semangatnya dalam pembuatan skripsi ini.
9. Seluruh pihak-pihak yang telah mendukung penulis baik secara langsung maupun tidak langsung.
10. ” Membaca bukanlah tujuan akhir, melainkan alat untuk mencapai tujuan” – *Adolf Hitler*.

Penulis memahami bahwa masih banyak kekurangan dalam skripsi ini, maka dari itu penulis sangat mengharapkan kritik dan saran yang bersifat membangun untuk karya yang lebih baik ke depannya. Akhir kata, penulis berharap semoga skripsi ini dapat bermanfaat bagi semuanya, dan dapat dimanfaatkan dengan semestinya. Semoga Allah SWT membalas kebaikan dan selalu mencurahkan hidayahnya serta taufik-Nya, Amin.

Wassalamu’aikum warahmatullahi wabarakatuh

|  |
| --- |
| Tangerang Selatan,……..2023 |
| Hormat saya |
|  |
| (Muhammad Irgi Al Ghithraf) |

## DAFTAR ISI

[*ABSTRACT* ii](#_Toc148541006)

[ABSTRAK iii](#_Toc148541007)

[KATA PENGANTAR iv](#_Toc148541008)

[DAFTAR ISI vi](#_Toc148541009)

[DAFTAR GAMBAR viii](#_Toc148541010)

[DAFTAR TABEL ix](#_Toc148541011)

[BAB I PENDAHULUAN 1](#_Toc148541012)

[1.1 Latar Belakang 1](#_Toc148541013)

[1.2 Identifikasi Masalah 2](#_Toc148541014)

[1.3 Rumusan Masalah 2](#_Toc148541015)

[1.4 Batasan Penelitian 3](#_Toc148541016)

[1.5 Tujuan Penelitian 3](#_Toc148541017)

[1.6 Manfaat Penelitian 3](#_Toc148541018)

[1.6.1 Manfaat Bagi Penulis 3](#_Toc148541019)

[1.6.2 Manfaat Bagi Pengguna 4](#_Toc148541020)

[1.7 Metodelogi Penelitian 4](#_Toc148541021)

[1.7.1 Metode Pengumpulan Data 4](#_Toc148541022)

[1.7.2 Metode Perancangan Sistem 4](#_Toc148541023)

[1.8 Sistematika Penulisan 5](#_Toc148541024)

[BAB II LANDASAN TEORI 7](#_Toc148541025)

[2.1 Penelitian Terkait 7](#_Toc148541026)

[2.2 Teori Umum 11](#_Toc148541027)

[2.2.1 Kecerdasan Buatan *(Artificial Intelligence)* 11](#_Toc148541028)

[2.2.2 *Chatbot* 11](#_Toc148541029)

[2.2.3 *Application Programming Interface (API)* 11](#_Toc148541030)

[2.2.4 *Natural Language Processing (NLP)* 12](#_Toc148541031)

[2.2.5 *Open AI ChatGPT* 13](#_Toc148541032)

[2.2.6 *Gradio App* 14](#_Toc148541033)

[2.2.7 *Flowchart Diagram* 14](#_Toc148541034)

[2.3 *Unified Modelling Language (UML)* 15](#_Toc148541035)

[2.3.1 *Use Case Diagram* 16](#_Toc148541036)

[2.3.2 *Activity Diagram* 18](#_Toc148541037)

[2.3.3 *Sequence Diagram* 19](#_Toc148541038)

[2.4 Pengujian Sistem 21](#_Toc148541039)

[2.4.1 Tujuan Pengujian 21](#_Toc148541040)

[2.4.2 Pengujian *BlackBox Testing* 21](#_Toc148541041)

[2.5 Perangkat Lunak Pendukung 22](#_Toc148541042)

[2.5.1 Bahasa Pemrograman *Python* 22](#_Toc148541043)

[2.5.2 *Visual Studio Code (VS Code)* 22](#_Toc148541044)

[2.5.3 *Uptime Robot* 22](#_Toc148541045)

[2.5.4 *Draw.io* 23](#_Toc148541046)

[2.5.5 *Figma* 23](#_Toc148541047)

[BAB III ANALISA DAN PERANCANGAN 16](#_Toc148541048)

[3.1 Analisa Kebutuhan 16](#_Toc148541049)

[3.2 Analisa Sistem 16](#_Toc148541050)

[3.2.1 Analisa Sistem Berjalan 16](#_Toc148541051)

[3.2.2 Analisa Sistem Usulan 17](#_Toc148541052)

[3.3 Perancangan *Unified Modelling Language (UML)* 17](#_Toc148541053)

[3.3.1 *Use Case* Diagram *Chatbot (ALICE)* 17](#_Toc148541054)

[3.3.2 *Activity Diagram* Berjalan *Chatbot (ALICE)* 18](#_Toc148541055)

[3.3.3 *Activity Diagram* Usulan *Chatbot (ALICE)* 19](#_Toc148541056)

[3.3.4 *Sequence Diagram Chatbot (ALICE)* 20](#_Toc148541057)

[3.4 *Flowchart Diagram Chatbot (ALICE)* 21](#_Toc148541058)

[3.5 Perancangan Antarmuka (*User Interface*) 22](#_Toc148541059)

[BAB 4 PENGUJIAN DAN IMPLEMENTASI 24](#_Toc148541060)

[DAFTAR PUSTAKA 7](#_Toc148541061)

## DAFTAR GAMBAR

[**Gambar 2.1:** *Natural Language Processing (NLP)* 12](#_Toc147674290)

[**Gambar 2.2:** *Logo Open AI ChatGPT* 13](#_Toc147674291)

[**Gambar 3.1:** *Use Case Diagram Chatbot (ALICE)* 17](#_Toc148260490)

[**Gambar 3.2:** *Activity Diagram Berjalan Chatbot (ALICE)* 18](#_Toc148260491)

[**Gambar 3.3:** *Activity Diagram Usulan Chatbot (ALICE)* 19](#_Toc148260492)

[**Gambar 3.4:** *Sequence Diagram Chatbot (ALICE)* 20](#_Toc148260493)

[**Gambar 3.5:** *Flowchart Diagram Chatbot (ALICE)* 21](#_Toc148260494)

[**Gambar 3.6:** Halaman *Laptop* *Chatbot (ALICE)* 22](#_Toc148260495)

[**Gambar 3.7:** Halaman *Smartphone* *Chatbot (ALICE)* 23](#_Toc148260496)

## DAFTAR TABEL

[**Tabel 2.1**: Penelitian Terkait 7](#_Toc147739668)

[**Tabel 2.2:** Simbol – Simbol *Flowchart Diagram* 14](#_Toc147739669)

[**Tabel 2.3:** Simbol – Simbol *Use Case Diagram* 16](#_Toc147739670)

[**Tabel 2.4:** Simbol – Simbol *Activity Diagram* 18](#_Toc147739671)

[**Tabel 2.5:** Simbol – Simbol *Sequence Diagram* 20](#_Toc147739672)

## BAB I PENDAHULUAN

### Latar Belakang

Kecerdasan buatan atau disebut juga *Artificial Intelligence (AI)* merupakan teknologi di bidang ilmu komputer yang mensimulasikan kecerdasan manusia ke dalam komputer untuk menyelesaikan berbagai persoalan dan pekerjaan seperti dan sebaik yang dilakukan manusia, bahkan bisa lebih baik dari manusia (Pratama, 2022). Salah satu aspek yang mendapat perhatian besar adalah Pemrosesan Bahasa Alami *NLP* *(Natural Language Processing)*, yaitu kemampuan komputer untuk memahami bahasa manusia seperti halnya manusia.

Seiring dengan perkembangan teknologi *AI* dan *NLP*, munculnya sebuah arsitektur model yang mampu mengatasi masalah pemrosesan bahasa dengan sangat baik, telah menjadi dasar bagi beberapa model terkemuka dalam pemrosesan bahasa alami. Salah satu implementasi terkemuka adalah *GPT* (*Generative Pre-trained Transformer*) yang dikembangkan oleh *OpenAI* (Setiawan, 2023).

Penggunaan *API ChatGPT* dari *OpenAI* untuk membangun *chatbot* yang lebih canggih dan responsif telah menjadi fokus penelitian dan pengembangan. *API ChatGPT* memungkinkan akses langsung ke model *GPT* yang dilatih secara besar-besaran, yang mampu menghasilkan teks yang lebih alami dan kontekstual. Aplikasi *Chatbot* sangat bergantung pada data train yang dibuat atau masukan dari pengembang chatbot saat mengenali dan merespons. Bisa dengan menggunakan metode penandaan untuk setiap pola kalimat untuk meningkatkan akurasi sistem (Chandra, 2021).

Dalam penelitian ini, penulis akan menggunakan *API ChatGPT* dalam implementasi *chatbot* yang terhubung dengan *Gradio app*. *Gradio app* adalah sebuah *framework* yang memfasilitasi pembuatan antarmuka pengguna interaktif tanpa memerlukan pengetahuan mendalam dalam pengembangan antarmuka. Dengan menggabungkan kemampuan *ChatGPT*, *Gradio app*, dan konsep *NLP (Natural Language Processing)* yang sederhana. Judul yang penulis ajukan yang berjudul “**IMPLEMENTASI *CHATBOT* DENGAN MEMANFAATKAN *API CHATGPT* MELALUI *GRADIO APP* MENGGUNAKAN METODE *NATURAL LANGUAGE PROCESSING (NLP)***” yang bertujuan untuk menciptakan *chatbot* yang dapat memberikan pengalaman berkomunikasi yang lebih alami, efektif, dan responsif bagi pengguna. Penggunaan metode *NLP* *(Natural Language Processing)* nanti akan digunakan untuk memungkinkan *chatbot* memahami bahasa manusia, mengenali makna dalam teks yang dimasukkan oleh pengguna, dan memberikan respon yang sesuai dan kontekstual.

### 1.2 Identifikasi Masalah

Bedasarkan latar belakang di atas, terdapat beberapa masalah yang dapat diidentifikasikan yaitu sebagai berikut:

1. Bagaimana memastikan bahwa respon yang dihasilkan oleh *Chatbot* mampu memenuhi harapan pengguna dan memberikan solusi yang tepat?
2. Bagaimana mengatasi variasi dalam gaya berbahasa dan penggunaan kata yang berbeda-beda antara pengguna?
3. Bagaimana memastikan bahwa pengguna dapat berinteraksi dengan *Chatbot* secara nyaman dan tanpa hambatan yang menghambat?

### 1.3 Rumusan Masalah

Berdasarkan latar belakang dan identifikasi masalah yang telah diuraikan sebelumnya, rumusan masalah dalam penelitian ini adalah:

1. Bagaimana mengatasi kendala dalam menghasilkan respon *Chatbot* yang relevan dan akurat terhadap permintaan dan pertanyaan pengguna?
2. Bagaimana melatih *Chatbot* agar mampu menyesuaikan diri dengan beragam ragam bahasa dan gaya ekspresi yang digunakan oleh pengguna?
3. Bagaimana merancang antarmuka *Gradio App* yang mampu memberikan pengalaman pengguna yang optimal dalam berinteraksi dengan *Chatbot*?

### 1.4 Batasan Penelitian

Berdasarkan latar belakang dan pengamatan yang dilakukan maka dapat diuraikan beberapa batasan penelitian yang dihadapi antara lain:

1. Penelitian ini akan berfokus pada *Chatbot* berbasis teks yang menggunakan *API ChatGPT* untuk menghasilkan respon teks.
2. Metode yang digunakan dalam penelitian ini adalah metode *Natural Language Processing (NLP)*.
3. Penelitian ini akan menggunakan *Free API ChatGPT* yang dimana 1 *Product Key* *API* hanya bisa menampung 3000 kata saja.

### 1.5 Tujuan Penelitian

Penelitian ini bertujuan untuk mengatasi masalah-masalah yang telah diidentifikasi dalam implementasi *Chatbot* dengan memanfaatkan *API ChatGPT* melalui *Gradio App* menggunakan metode *Natural Language Processing (NLP)*, dengan tujuan penelitian sebagai berikut:

1. Mengimplementasikan *Chatbot* yang mampu memberikan respon yang relevan, akurat, dan kontekstual terhadap berbagai permintaan dan pertanyaan pengguna dengan menggunakan *API ChatGPT* melalui *Gradio App* berbasis *NLP*.
2. Meningkatkan kualitas respon *Chatbot* dengan mengoptimalkan penggunaan model bahasa berbasis *GPT* (*Generative Pre-trained Transformer*) untuk menghasilkan respon yang lebih tepat dan informatif.

### 1.6 Manfaat Penelitian

Penelitian ini diharapkan memberikan manfaat sebagai berikut:

### Manfaat Bagi Penulis

1. Penulis dapat lebih memahami dan memperluas wawasan dan dapat mengembangkan ilmu yang didapat dari proses perkuliahan yang berlangsung.
2. Penelitian ini adalah salah satu syarat kelulusan dalam menyelesaikan program Strata 1 (S1).

### 1.6.2 Manfaat Bagi Pengguna

Penulis berharap dari penelitian ini pengguna mendapatkan interaksi dengan baik, solusi yang memadai, dan respon yang relevan akan berkontribusi pada peningkatan kepuasan pengguna terhadap penggunaan *Chatbot*.

### 1.7 Metodelogi Penelitian

Dalam Implementasi *Chatbot* dengan Memanfaatkan *API ChatGPT* melalui *Gradio App* Menggunakan Metode *Natural Language Processing (NLP)*, Penulis menggunakan metode penelitian yaitu:

### Metode Pengumpulan Data

1. Metode Studi Pustaka

Teknik pengumpulan data dengan mengumpulkan dan mempelajari buku-buku referensi dan sumber-sumber yang berkaitan dengan topik penelitian.

1. Metode Kuesioner

Teknik Pengumpulan data yang dilakukan dengan cara memberi seperangkat pertanyaan atau pernyataan tertulis kepada responden untuk dijawab.

1. Metode Observasi

Teknik pengumpulan data dengan cara melakukan pengamatan langsung pada *chatbot* yang sudah pernah dibuat sebelumnya dan pengamatan menganai beberapa permasalahan yang berhubungan dengan pentingnya pembuatan *chatbot* sebagai layanan informasi yang akurat bagi penggunannya.

### Metode Perancangan Sistem

Dalam melakukan perancangan aplikasi ini, metode yang digunakan oleh penulis adalah metode *waterfall*. Metode *Waterfall* merupakan metode yang menyediakan pendekatan alur hidup perangkat lunak secara skuensial atau terurut (Kurniawati, 2021). Tahapan dalam metode *waterfall* adalah sebagai berikut:

1. Rekayasa Sistem dan Pemodelan Sistem

Tahap ini sangat menekankan pada masalah pengumpulan kebutuhan pengguna pada tingkat sistem dengan mendefinisikan konsep sistem. Rekayasa sistem meliputi pengumpulan kebutuhan pada tingkat sistem dengan sejumlah kecil analisis.

1. Desain

Proses pembuatan desain tahap awal merancang tampilan *website*, sehingga menghasilkan desain *interface* yang interaktif.

1. Perkodean dan Pengujian

Dibuat program dengan menggunakan bahasa pemrograman *Python* yang mengacu kepada desain yang telah dibuat sehingga siap dijalankan dan diuji.

1. Uji Coba

Setelah program selesai dibuat, maka dilakukan pengujian terhadap sistem tersebut, apakah aplikasi tersebut berjalan sesuai dengan yang diharapkan atau tidak. Apabila masih terdapat kesalahan pada aplikasi maka dilakukan tahapan analisis kembali untuk mengetahui kesalahan yang muncul.

1. Penerapan

Tahap ini bisa dikatakan terakhir dalam pembuatan sebuah aplikasi. Setelah melakukan analisa, desain, dan pengkodean maka aplikasi yang sudah jadi digunakan oleh *user*.

### 1.8 Sistematika Penulisan

Sistematika Penulisan ini dibuat dengan tujuan memberikan gambaran mengenai isi proposal secara singkat, sehingga pembaca lebih mudah untuk memahami.

**BAB I PENDAHULUAN**

Dalam bab ini akan diuraikan secara singkat tentang latar belakang, identifikasi masalah, rumusan masalah, batasan penelitian, tujuan penelitian, manfaat penelitian dan sistematika penulisan.

**BAB II TINJAUAN PUSTAKA**

Pada bab ini berisi mengenai landasan teori sebagai parameter rujukan untuk terlaksana nya peneltian ini, pada bab ini juga ada penelitian terkait yang penulis ambil untuk menjadi referensi.

**BAB III ANALISA DAN PERANCANGAN**

Pada bab ini akan diuraikan tentang analisa kebutuhan, *activity diagram*, *flowchart diagram* dan *usecase diagram*.

**BAB IV IMPLEMENTASI DAN PENGUJIAN**

Pada bab ini berisi mengenai implementasi pada *chatbot* yang dapat meliputi implementasi pengujian, pengujian menggunakan *blackbox* dan hasil analisa.

**BAB V PENUTUP**

Pada bab ini berisi kesimpulan dari sistem yang telah dibuat oleh penulis atau dalam kata lain rangkuman dari semua yang telah dilakukan serta saran untuk kepentingan kedepannya.

## BAB II LANDASAN TEORI

### 2.1 Penelitian Terkait

Dalam menyusun skripsi ini, penulis terinspirasi dan mereferensi dari penelitian-penelitian terdahulu yang masih berkaitan dengan latar belakang masalah pada skripsi ini. Berikut ini penelitian terdahulu yang berkaitan dengan skripsi, antara lain:

**Tabel 2.1**: Penelitian Terkait

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Judul Penelitian** | **Metode** | **Hasil** | **Kesimpulan** |
| 1. | *Chatbot Telegram* Menggunakan *Natural Language Processing*.  Peneliti: Muhammad Furqan.  Tahun: 2023 | *NATURAL LANGUAGE PROCESSING (NLP)* | *Chatbot* Informasi Mahasiswa Berbasis *Telegram* Pada sistem ini telah berjalannya pengembangan *chatbot* berbasis *telegram* dengan konten berisi informasi mahasiswa, Yang dimana mahasiswa dapat dengan mudah mengetahui Jadwal kelas, Tugas Mata kuliah dan Pengumuman Penting Mahasiswa. | Bot perlu di training terlebih dahulu beberapa kata yang akan digunakan di dalam Chat Bot agar Bot dapat menentukan kata yang memiliki akurasi tertinggi untuk mengirimkan trigger ke server bot. |
| 2. | Sistem Notifikasi Kepangkatan Dosen Secara *Realtime* Berbasiskan *Telegram Bot Application Programming Interface.*  Peneliti: I Ketut Gede Sugita.  Tahun: 2023 | *Research and Development (R&D).* | Sistem Notifikasi Kepangkatan Dosen Secara *Realtime* Berbasiskan *Telegram Bot API* dibuat dengan memperhitungkan 3 entitas utama, mulai dari administrasi, dosen, serta pimpinan yang terlibat memiliki peran masing-masing dalam sistem. | Sistem Notifikasi Kepangkatan Dosen Secara *Realtime* Berbasiskan *Telegram* *Bot Application Programming Interface* menghasilkan sebuah sistem yang mencakup kenaikan pangkat dan menggunakan *Telegram bot* untuk memberikan notifikasi dan sebagai media interaksi dua arah untuk mendapatkan informasi seputar kepegawaian dosen dan kenaikan pangkat. |
| 3. | Pengembangan *Chatbot* Informasi Mahasiswa Berbasis *Telegram* dengan Metode *Natural Language Processing*.  Peneliti: Alfan Adi Chandra.  Tahun: 2021 | *Natural Language Processing (NLP).* | Untuk dapat menjalankan *ChatBot* dengan baik diperlukan *Server* yang baik dan cepat kemudian memerlukan Desain Arsitektur *ChatBot* untuk mengetahui masing masing kegunaan perintah pada *ChatBot*. | Data yang diperlukan untuk dikirimkan kepada pengguna harus di input ke dalam *source code* terlebih dahulu agar bisa mengirimkan pesan sesuai *trigger* yang dikirimkan dari website wit.ai. |
| 4. | Pengujian *Black Box* pada Aplikasi Sistem Seleksi *Sales* Terbaik Menggunakan Teknik *Equivalence Partitions.*  Peneliti: Fadhila Cahya Ningrum.  Tahun: 2019 | *BlackBox* | Aplikasi seleksi sales terbaik ditemukan kesalahan dalam mengedit data di mana tidak sesuai antara pesan yang muncul yaitu data berhasil disimpan sedangkan yang terjadi data tersebut tida berubah. | Pengujian dengan metode *Black Box* berbasis *Equivalence Partitions* dapat membantu proses pembuatan case pengujian, uji kualitas dan menemukan kesalahan yang tidak terdeteksi yang disebabkan oleh kesalahan pengetikan. |
| 5. | Penerapan Media *Chat GPT* pada Pembelajaran Manajemen Pendidikan terhadap Kemandirian Mahasiswa.  Peneliti: Hary Murcahyanto.  Tahun: 2023. | *Pre-experimental* | Hasil penelitian menunjukkan bahwa penggunaan *Chat GPT* pada pembelajaran manajemen pendidikan dapat meningkatkan kemandirian mahasiswa, sehingga dapat menjadi alternatif metode pembelajaran yang efektif dan perlu dilakukan filtrasi dan diarahkan oleh dosen. | Meskipun demikian, penelitian ini memiliki keterbatasan yaitu hanya dilakukan pada satu kelompok sampel yang relatif kecil, sehingga perlu dilakukan penelitian lebih lanjut dengan sampel yang lebih besar dan dari berbagai perguruan tinggi. Penggunaan *Chat GPT* dalam pembelajaran juga memiliki tantangan nilai etika dan orisinalitas karya, sehingga perlu dipertimbangkan dengan kritis. |

### 2.2 Teori Umum

Pada bagian ini akan membahas mengenai semua teori yang digunakan peneliti untuk dasar penulisan dalam mengimplementasikan sebuah *chatbot* dengan memanfaatkan *API ChatGPT* melalui *Gradio App* menggunakan metode *Natural Language Processing (NLP)*.

### 2.2.1 Kecerdasan Buatan *(Artificial Intelligence)*

Menurut (Kalsum, 2022) Kecerdasan buatan (*Artificial Intelligence*) merupakan suatu inovasi baru dalam ilmu pengetahuan. Adanya kecerdasan buatan dimulai sejak munculnya komputer *modern* pada tahun 1940 dan tahun 1950 *AI* merupakan alat atau komputer yang dapat melakukan tugas yang dilakukan oleh manusia, oleh karena itu *AI* juga dapat dimanfaatkan dalam kegiatan bermain *game*.

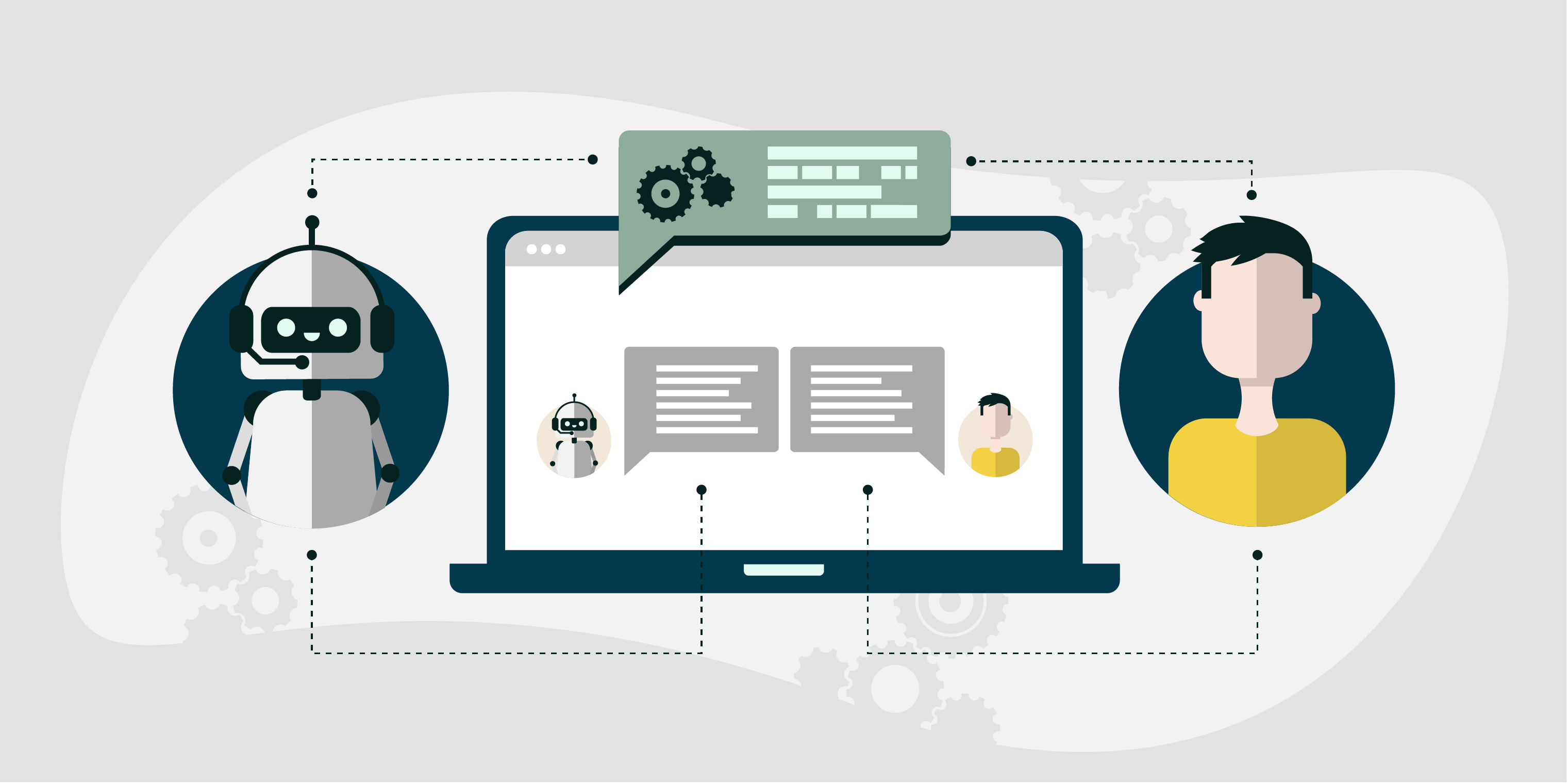
### 2.2.2 *Chatbot*

Menurut (Mhd. Furqan, 2023) *Chatbot* adalah implementasi kecerdasan buatan yang mensimulasikan percakapan *real time* antara mesin dan manusia melalui tulisan, suara dan visual. Aplikasi *Chatbot* sangat bergantung pada data train yang dibuat atau masukan dari pengembang *chatbot* saat mengenali dan merespons. Bisa dengan menggunakan metode penandaan untuk setiap pola kalimat untuk meningkatkan akurasi sistem.

### 2.2.3 *Application Programming Interface (API)*

Menurut (Afriansyah, 2021) *Application Programming Interface (API)* adalah antarmuka yang dibangun oleh pengembang sistem sehingga beberapa atau seluruh fungsi sistem dapat diakses secara terprogram. *Application Programming Interface (API)* merupakan ekspresi terfokus keseluruhan fungsional dalam suatu modul *software* yang dapat diakses oleh orang yang membutuhkan dengan cara yang telah ditentukan layanan. Representasi terfokus dari fungsi yang dideklarasikan dalam *API* dimaksudkan untuk menyediakan rangkaian layanan yang spesifik untuk target tertentu.

### 2.2.4 *Natural Language Processing (NLP)*



**Gambar 2.1:** *Natural Language Processing (NLP)*

Menurut (H. Husamuddin, 2020) dalam penelitian yang dilakukan dengan menggunakan metode *Natural Language Processing (NLP)*. Metode ini memudahkan pengguna untuk menggunakan sistem ini. Pertama, pengguna memasukkan teks di klien obrolan yang disediakan. Teks dapat berupa jawaban atas pertanyaan atau pernyataan. Selain itu, input teks diteruskan ke sistem *chatbot*, untuk mensimulasi percakapan dengan menggunakan bahasa sehari hari.

Beberapa metode yang umum digunakan dalam *Natural Language Processing (NLP)*:

1. Tokenisasi

Proses memecah teks menjadi unit-unit lebih kecil, seperti kata atau frasa. Tokenisasi adalah langkah pertama dalam pemrosesan teks dan membantu dalam analisis lebih lanjut.

1. *Part-of-Speech Tagging*

Metode ini melibatkan mengidentifikasi bagian dari pidato (*part-of-speech*) dari setiap kata dalam sebuah kalimat, seperti kata benda, kata kerja, kata sifat, dll.

1. Model Bahasa dan Pembelajaran Mesin

Pendekatan terbaru menggunakan model bahasa berbasis pembelajaran mesin, seperti *GPT (Generative Pre-trained Transformer)* yang diberi tahu dari korpus teks besar untuk memahami dan menghasilkan teks yang lebih alami.

1. Konversi Teks menjadi Suara

Menggunakan *gTTS* *(Google Text-to-Speech)* untuk mengubah teks jawaban yang dihasilkan oleh model menjadi file suara.

1. Penerjemahan Mesin

Menerjemahkan teks dari satu bahasa ke bahasa lain dengan bantuan komputer.

1. Analisis Sentimen

Menentukan sentimen (positif, negatif, atau netral) dalam teks. Ini bisa digunakan untuk menganalisis pandangan orang terhadap suatu topik.

1. Pengenalan Nama Entitas

Melibatkan mengidentifikasi entitas seperti nama orang, tempat, tanggal, dan sebagainya dalam teks.

### 2.2.5 *Open AI ChatGPT*



**Gambar 2.2:** *Logo Open AI ChatGPT*

Menurut (Setiawan, 2023) *ChatGPT* merupakan mesin cerdas yang dilatih untuk bisa menirukan percakapan manusia menggunakan teknologi *NLP (Natural Language Processing)*. Kenyataannya *ChatGPT* dapat dimanfaatkan untuk menghasilkan suatu tulisan yang cukup ilmiah dengan *prompt* yang dirumuskan di awal dengan teknik yang baik dan efektif.

Teknologi tersebut dirilis oleh perusahaan Amerika, *Open AI* pada tanggal 30 November 2022. Model tersebut dapat memahami dan merespons input pengguna dengan cara yang meniru percakapan manusia, sehingga interaksi menjadi lebih alami dan menarik. Selain itu, *ChatGPT* dapat menghasilkan teks dalam berbagai gaya dan format, seperti artikel berita, surel, dan puisi, sehingga menjadi teknologi yang serbaguna dan berguna untuk berbagai aplikasi.

### 2.2.6 *Gradio App*

*Gradio App* adalah platform yang memudahkan pengembang dalam membuat antarmuka pengguna *(UI)* interaktif untuk model pembelajaran mesin dengan cara yang lebih mudah dan cepat. Dengan *Gradio App*, pengembang dapat membuat antarmuka yang memungkinkan pengguna berinteraksi langsung dengan model pembelajaran mesin tanpa perlu menulis kode pemrograman yang rumit.

### 2.2.7 *Flowchart Diagram*

Menurut (Achlison, 2020) *Flowchart* adalah adalah alur kerja dari suatu proses terhadap sistem yang telah dibuat agar dapat dengan mudah untuk dipahami dan dijelaskan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses (instruksi) dengan proses lainnya dalam suatu program.

**Tabel 2.2:** Simbol – Simbol *Flowchart Diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Simbol** | **Nama** | **Keterangan** |
| 1. |  | *Arrows* | Penghubung yang menunjukkan hubungan antara bentuk-bentuk representatif. |
| 2. |  | *Start/end* | Oval melambangkan titik awal atau akhir. |
| 3. |  | *Input/Output* | Jajar genjang melambangkan *input* atau *output* (*READ, WRITE*). |
| 4. |  | *Process* | Persegi panjang melambangkan sebuah proses (Menyatakan *assignment* *statement*). |
| 5. |  | *Decision* | Belah ketupat mengindikasikan sebuah keputusan (*True, False*). |

### 2.3 *Unified Modelling Language (UML)*

Menurut (Putra, 2019) *Unified Modelling Language (UML)* adalah salah satu standar bahasa yang banyak digunakan di dunia industri untuk mendefinisikan *requirement*, membuat analisis dan desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek.

### 2.3.1 *Use Case Diagram*

Menurut (Aliman, 2021) *Use Case* merupakan langkah pertama dalam memodelkan sebuah sistem. *Use Case* merupakan pemodelan untuk kebutuhan sebuah sistem fungsional, setiap *Use Case* digambarkan sebagai kunci dari suatu skenario yang dilakukan oleh aktor dan diringkas dalam sebuah batas sistem, setiap *Use Case* dihubungkan dengan sebuah garis notasi.

**Tabel 2.3:** Simbol – Simbol *Use Case Diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Simbol** | **Nama** | **Keterangan** |
| 1. |  | Aktor | Menspesifikasikan himpunan peran yang pengguna mainkan ketika berinteraksi dengan *use case* diagram |
| 2. |  | *Dependency* | Hubungan dimana perubahan yang terjadi pada suatu elemen mandiri (*Dependency*) *independent* akan mempengaruhi elemen yang bergantung padanya elemen yang tidak mandiri. |
| 3. |  | *Association* | Abstraksi dari penghubung antara actor dengan *use case*. |
| 4. |  | *Generalization* | Menunjukan spesialisasi aktor untuk dapat berpartisipasi dengan *use case*. |
| 5. |  | *Extend* | Menspesifikasikan bahwa target *use case* memperluas perilaku dari sumber *use case* pada suatu titik yang diberikan. |
| 6. |  | *Include* | Menspesifikasikan bahwa sumber *use case* secara eksplisit. |
| 7. |  | *System* | Menspesifikasikan paket yang menampilkan sistem secara terbatas. |
| 8. |  | *Use Case* | Deskripsi dari urutan aksi- aksi yang metapilkan oleh sistem yang menghasilkan suatu hasil yang terukur bagi suatu aktor. |

### 2.3.2 *Activity Diagram*

Menurut (Arianti, 2022) *Activity diagram* adalah suatu diagram yang menggambarkan konsep aliran data/kontrol, aksi terstruktur serta dirancang dengan baik dalam suatu sistem. *Activity diagram* digambarkan dengan sebuah alur secara terstruktur proses kerja dari *use case* yang sedang diproses dari titik awal sampai titik akhir, setiap aktivitas digambarkan dengan notasi-notasi sesuai fungsinya.

**Tabel 2.4:** Simbol – Simbol *Activity Diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Simbol** | **Nama** | **Keterangan** |
| 1. |  | *Initial Node* | Bagaimana objek dibentuk atau diawali. |
| 2. |  | *Activity Final Node* | Bagaimana objek dibentuk dan diakhiri. |
| 3. |  | *Activity* | Memperlihatkan bagaimana masing-masing kelas antarmuka saling berinteraksi satu sama lain/ |
| 4. |  | *Action* | *State* dari sistem yang mencerminkan eksekusi dari suatu aksi. |
| 5. |  | *Decision* | Digunakan untuk menggambarkan suatu keputusan / tindakan yang harus diambil pada kondisi tertentu. |
| 6. |  | *Line Connector* | Digunakan untuk menghubungkan satu simbol dengan simbol lainnya. |

### 2.3.3 *Sequence Diagram*

Menurut (Prasetya, 2022) *Sequence diagram* atau diagram urutan adalah sebuah diagram yang digunakan untuk menjelaskan dan menampilkan interaksi antar objek-objek dalam sebuah sistem secara terperinci. Selain itu *sequence diagaram* juga akan menampilkan pesan atau perintah yang dikirim, beserta waktu pelaksanaannya. Objek- objek yang berhubungan dengan berjalannya proses operasi biasanya diurutkan dari kiri ke kanan.

**Tabel 2.5:** Simbol – Simbol *Sequence Diagram*

|  |  |  |  |
| --- | --- | --- | --- |
| **No** | **Simbol** | **Nama** | **Keterangan** |
| 1. |  | *Actor* | Menggambarkan orang yang sedang berinteraksi dengan sistem. |
| 2. |  | *Entity Class* | Gambaran sistem sebagai landasan dalam menyusun basis data. |
| 3. |  | *Boundary Class* | Menangani komunikasi antar lingkungan sistem. |
| 4. |  | *Control Class* | Bertanggung jawab terhadap kelas-kelas terhadap objek yang berisi logika. |
| 5. |  | *Activation* | Mewakili proses durasi aktivasi sebuah operasi. |
| 6. |  | *Life Line* | Komponen yang digambarkan garis putus-putus terhubung dengan objek. |
| 7. |  | *A Message* | Menggambarkan pengiriman pesan |

### 2.4 Pengujian Sistem

### 2.4.1 Tujuan Pengujian

Menurut (Rosalina, 2020) dalam Sulistyanto & SN, 2014. Tujuan pengujian sistem perangkat lunak adalah cara untuk mendapatkan sebuah informasi mengenai kualitas dari perangkat lunak yang diuji dan pengujian *software* sangat penting dilakukan dengan tujuan untuk mengetahui kesalahan yang ada saat dilakukannya pembuatan *software*.

### 2.4.2 Pengujian *BlackBox Testing*

Menurut (Ningrum, 2019) Metode *Blackbox* *Testing* adalah sebuah metode yang dipakai untuk menguji sebuah *software* tanpa harus memperhatikan detail *software*. Pengujian ini hanya memeriksa nilai keluaran berdasarkan nilai masukan masing-masing. Tidak ada upaya untuk mengetahui kode program apa yang output pakai.

### 2.5 Perangkat Lunak Pendukung

Perangkat lunak pendukung untuk pembuatan *chatbot* adalah komponen-komponen atau hal penting yang dibutuhkan, diantara nya adalah:

### 2.5.1 Bahasa Pemrograman *Python*

Menurut (Clinton, 2019) Bahasa Pemrograman *Python* merupakan bahasa pemrograman yang freeware atau perangkat bebas dalam arti sebenarnya, tidak ada batasan dalam penyalinan atau mendistribusikannya. Lengkap dengan *source code*, *debugger* dan *profiler*, antarmuka yang terkandung di dalamnya untuk pelayanan antarmuka, fungsi sistem, *GUI* (antarmuka pengguna grafis), dan basis datanya.

### 2.5.2 *Visual Studio Code (VS Code)*

Menurut (A. Y. Permana, 2019) *Visual Studio Code (VS Code)* ini adalah sebuah teks *editor* ringan dan handal yang dibuat oleh *Microsoft* untuk sistem operasi *multiplatform*, artinya tersedia juga untuk versi *Linux, Mac,* dan *Windows*. *Teks* *editor* ini secara langsung mendukung bahasa pemrograman *JavaScript, Typescript,* dan *Node.js*, serta bahasa pemrograman lainnya dengan bantuan *plugin* yang dapat dipasang *via marketplace Visual Studio Code* (seperti C++, C#*, Python, Go, Java,* dst).

### 2.5.3 *Uptime Robot*

Menurut (Gutama, 2021) *Uptime Robot* merupakan sebuah aplikasi *web* layanan monitoring yang berfungsi untuk memantau *server* atau *website* apakah dapat diakses dengan baik atau tidak. Tidak bisa teraksesnya *website* biasanya disebabkan oleh *hacker*, *CPanel stuck*, gangguan *hardware*, *maintenance* rutin, *over CPU*, dan sebagainya. Dengan menggunakan aplikasi *Uptime Robot* kita dapat mengetahui penyebab dari *website* tersebut sehingga tidak bisa diakses. Setiap kali aplikasi mengalami *downtime*, *Uptime Robot* akan mengirimkan pemberitahuan melalui *email* yang terdaftar sehingga kita dapat mengetahuinya sedini mungkin.

### 2.5.4 *Draw.io*

Menurut (Purnomo, 2022) *Draw.io* adalah sebuah *website online* untuk menggambarkan macam-macam sebuah *diagram*. Fitur yang ada pada situs ini dapat digunakan hanya dengan bermodalkan *browser* yang telah mendukung *HTML* 5. *Draw.io* mempunyai tampilan yang menarik responsif sehingga mudah di akses menggunakan telepon pintar dan *PC*.

### 2.5.5 *Figma*

Menurut(Muhyidin, 2020) *Figma* adalah salah satu *design tool* yang biasanya digunakan untuk membuat tampilan aplikasi *mobile, desktop, website* dan lain-lain. *Figma* bisa digunakan di sistem operasi *windows, linux* ataupun *mac* dengan terhubung ke *internet*. Umumnya *Figma* banyak digunakan oleh seseorang yang bekerja dibidang *UI/UX, web design* dan bidang lainnya yang sejenis. Selain mempunyai kelengkapan fitur layaknya *Adobe XD*, *Figma* memiliki keunggulan yaitu untuk pekerjaan yang sama dapat dikerjakan oleh lebih dari satu orang secara bersama-sama walaupun ditempat yang berbeda. Hal tersebut bisa dikatakan kerja kelompok dan karena kemampuan aplikasi *figma* tersebut lah yang membuat aplikasi ini menjadi pilihan banyak *UI/UX designer* untuk membuat *prototype website* atau aplikasi dengan waktu yang cepat dan efektif.

## BAB III ANALISA DAN PERANCANGAN

### 3.1 Analisa Kebutuhan

Analisa Kebutuhan pengguna dan pemangku kepentingan dalam implementasi *Chatbot* dengan memanfaatkan API *ChatGPT* melalui *Gradio App* menggunakan Metode *Natural Language Processing (NLP)*. Analisa kebutuhan ini menjadi landasan untuk merancang *Chatbot* yang responsif dan sesuai dengan harapan pengguna.

Kebutuhan yang dimaksud adalah memahami jenis pertanyaan atau permintaan yang paling sering diajukan oleh pengguna kepada *Chatbot*. penulis perlu memahami preferensi pengguna terkait dengan bahasa dan gaya komunikasi yang pengguna sukai. Selain itu, penulis harus mengetahui sejauh mana pengguna mengharapkan *Chatbot* dapat memberikan solusi atau informasi yang relevan.

Selain itu, kebutuhan kemudahan penggunaan menjadi fokus utama. penulis harus memahami tingkat pengetahuan dan keterampilan pengguna dalam menggunakan teknologi dan *Chatbot*. Identifikasi hambatan atau kendala yang mungkin dihadapi oleh pengguna dalam berinteraksi dengan *Chatbot* juga menjadi bagian penting dalam analisa ini. Penulis juga harus memahami preferensi pengguna terkait dengan tampilan dan navigasi antarmuka pengguna.

### 3.2 Analisa Sistem

### 3.2.1 Analisa Sistem Berjalan

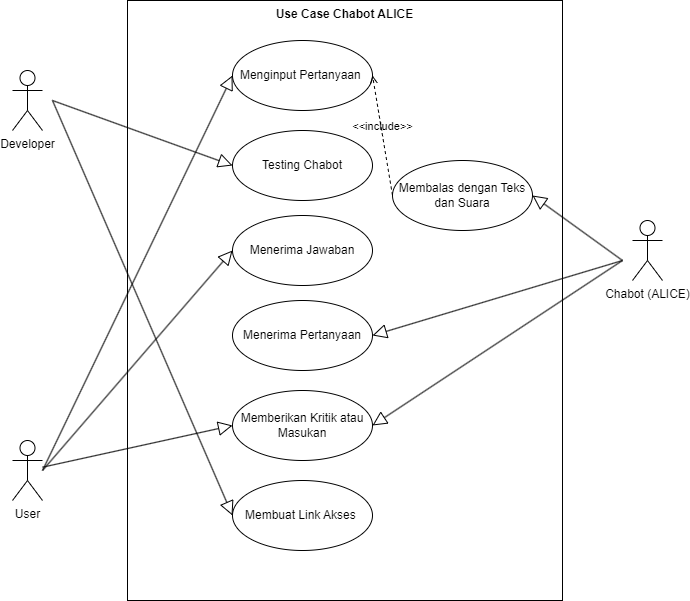
Analisa Sistem Berjalan dalam Implementasi *Chatbot* dengan memanfaatkan *API ChatGPT* melalui *Gradio App* menggunakan Metode *Natural Language Processing (NLP)*. Analisa Sistem Berjalan ini memungkinkan pengguna untuk mengajukan pertanyaan dalam bentuk teks dan menerima respons dari *Chatbot* berdasarkan kumpulan aturan yang telah ditentukan sebelumnya.

### 3.2.2 Analisa Sistem Usulan

Analisa Sistem Usulan dalam Implementasi *Chatbot* dengan memanfaatkan *API ChatGPT* melalui *Gradio App* menggunakan Metode *Natural Language Processing (NLP)*. Analisa Sistem Usulan ini memungkinkan pengguna untuk mengajukan pertanyaan dalam bentuk teks dan menerima respons dari *Chatbot* berupa teks dan suara berdasarkan kumpulan aturan yang telah ditentukan sebelumnya.

### 3.3 Perancangan *Unified Modelling Language (UML)*

### 3.3.1 *Use Case* Diagram *Chatbot (ALICE)*

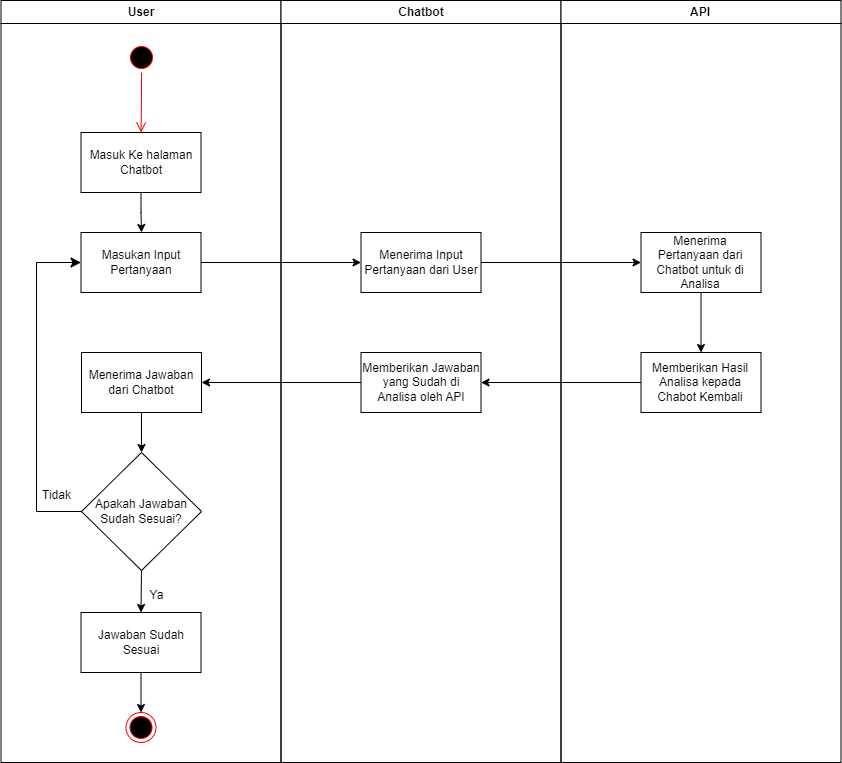


**Gambar 3.1:** *Use Case Diagram Chatbot (ALICE)*

Berikut ini adalah penjelasan dari (Gambar 3.1) U*se Case Diagram* di atas:

1. *Developer* merupakan aktor yang terpenting karena, *developer* membuat *chatbot (ALICE)*, membuat *link* akses *Gradio App* untuk *user* agar bisa menuju halaman *chatbot (ALICE)* dan menguji coba *chatbot* *(ALICE)* sebelum digunakan oleh *user*.
2. *User* merupakan aktor yang dapat menginput pertanyaan, menerima jawaban dan memberikan kritik atau masukan kepada *chatbot (ALICE).*
3. *Chatbot (ALICE)* merupakan aktor yang dapat menerima *input* pertanyaan dari *user,* membalas pertanyaan berupa teks dan suara yang sesuai *input* pertanyaan dari *user* dan *Chatbot (ALICE)* bisa memberikan kritik dan masukan kepada *user* berdasarkan *input* pertanyaan yang ditanyakan.

### 3.3.2 *Activity Diagram* Berjalan *Chatbot (ALICE)*

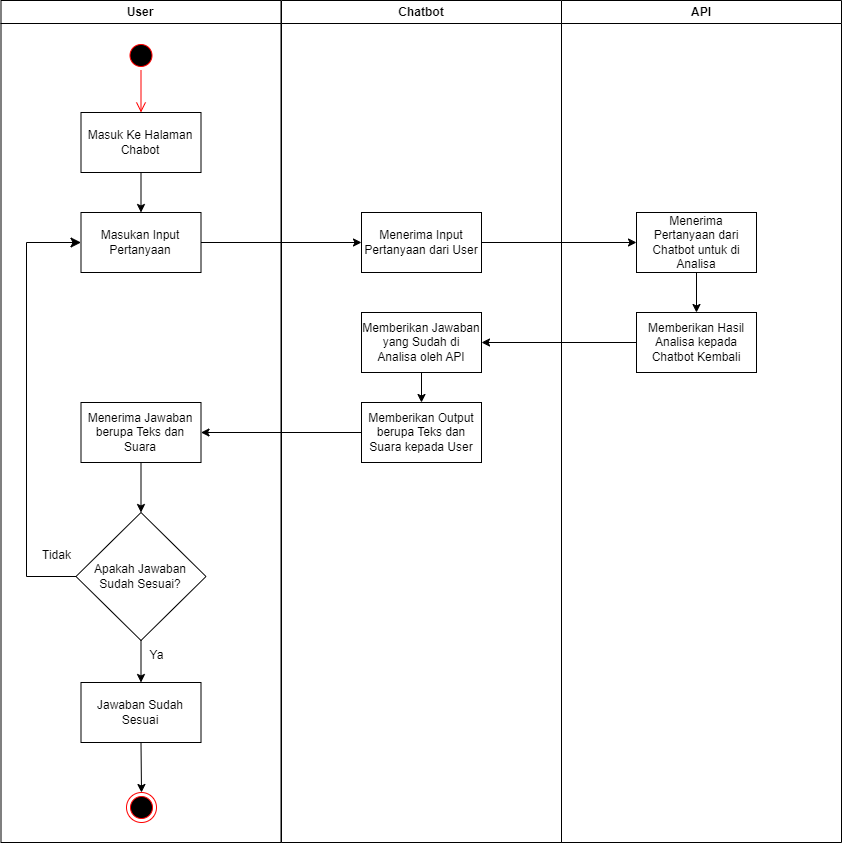


**Gambar 3.2:** *Activity Diagram Berjalan Chatbot (ALICE)*

Berikut ini adalah penjelasan dari (Gambar 3.2) *Activity Diagram* Berjalan di atas:

1. Langkah pertama adalah *user* masuk ke halaman *chatbot (ALICE)*.
2. *User* memasukan pertanyaan yang ingin ditanyakan.
3. *Chatbot (ALICE)* menerima *input* pertanyaan dari *user*.
4. Setelah *chatbot (ALICE)* mendapatkan pertanyaan, *chatbot (ALICE)* memberikan petanyaan kepada *API ChatGPT* untuk dianalisa.
5. *API ChatGPT* akan menganalisa pertanyaan yang telah diberikan oleh *chatbot (ALICE)*, kemudian hasil analisa akan diberikan kembali ke *chatbot (ALICE)* untuk memberikan jawaban yang telah dianalisa kepada *user*.
6. *Chatbot (ALICE)* menerima hasil pertanyaan yang telah dianalisa oleh *API ChatGPT*.
7. *User* mendapatkan jawaban dari *chatbot (ALICE)*.
8. Terakhir*, user* akan menyesuaikan antara jawaban dengan pertanyaan, apakah jawaban sudah sesuai dengan apa yang ditanyakan? Jika iya, maka *user* akan mendapatkan jawaban yang sudah sesuai dengan pertanyaan dan jika tidak, maka *user* dapat memasukan *input* pertanyaan kembali kepada *chatbot (ALICE)* agar mendapatkan jawaban yang sesuai.

### 3.3.3 *Activity Diagram* Usulan *Chatbot (ALICE)*

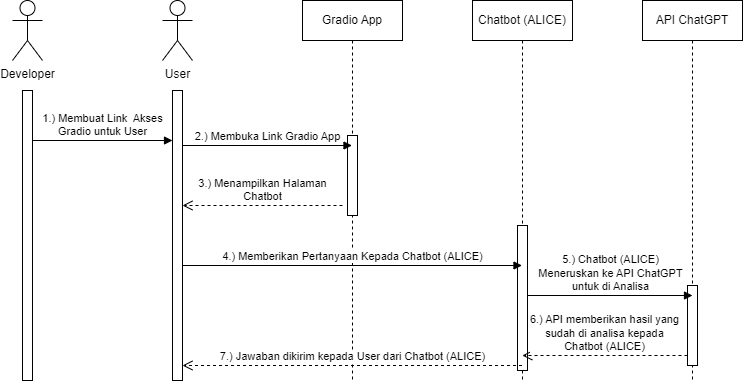


**Gambar 3.3:** *Activity Diagram Usulan Chatbot (ALICE)*

Berikut ini adalah penjelasan dari (Gambar 3.3) *Activity Diagram* Usulan di atas:

1. Langkah pertama adalah *user* masuk ke halaman *chatbot (ALICE)*.
2. *User* memasukan pertanyaan yang ingin ditanyakan.
3. *Chatbot (ALICE)* menerima *input* pertanyaan dari *user*.
4. Setelah *chatbot (ALICE)* mendapatkan pertanyaan, *chatbot (ALICE)* memberikan petanyaan kepada *API ChatGPT* untuk dianalisa.
5. *API ChatGPT* akan menganalisa pertanyaan yang telah diberikan oleh *chatbot (ALICE)*, kemudian hasil analisa akan diberikan kembali ke *chatbot (ALICE)* untuk memberikan jawaban yang telah dianalisa kepada *user*.
6. *Chatbot (ALICE)* menerima hasil pertanyaan yang telah dianalisa oleh *API ChatGPT* dan mengirimkan jawaban kepada *user* yaitu berupa teks dan suara.
7. Terakhir*, user* akan menyesuaikan antara jawaban dengan pertanyaan, apakah jawaban sudah sesuai dengan apa yang ditanyakan? Jika iya, maka *user* akan mendapatkan jawaban yang sudah sesuai dengan pertanyaan dan jika tidak, maka *user* dapat memasukan *input* pertanyaan kembali kepada *chatbot (ALICE)* agar mendapatkan jawaban yang sesuai.

### 3.3.4 *Sequence Diagram Chatbot (ALICE)*

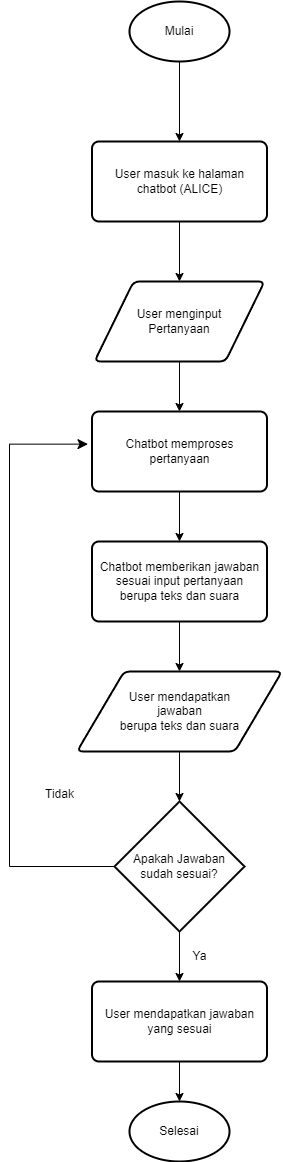


**Gambar 3.4:** *Sequence Diagram Chatbot (ALICE)*

Berikut ini adalah penjelasan dari (Gambar 3.4) *Sequence Diagram* di atas:

1. Langkah pertama yaitu *developer* membuat sebuah *link* *gradio app* sebagai akses *user* untuk menuju ke halaman *chatbot (ALICE)*.
2. *User* mendapatkan sebuah *link* yang sudah dibuat oleh *developer* untuk bisa masuk ke halaman *chatbot (ALICE)*.
3. *Link gradio* akan menampilkan halaman *chatbot (ALICE)* untuk *user*.
4. *User* meng-*input* pertanyaan yang ingin ditanyakan kepada *chatbot (ALICE)*.
5. *Chatbot (ALICE)* menerima *input* pertanyaan dari *user*, kemudian *chatbot (ALICE)* meneruskan pertanyaan tersebut kepada *API ChatGPT* untuk di analisa.
6. Setelah itu, *API ChatGPT* akan memberikan hasil yang sudah di analisa kepada *chatbot (ALICE)* yaitu berupa jawaban dari pertanyaan tersebut.
7. Terakhir, *chatbot (ALICE)* menjawab pertanyaan yang sudah di analisa berupa jawaban tersebut kepada *user*.

### 3.4 *Flowchart Diagram Chatbot (ALICE)*

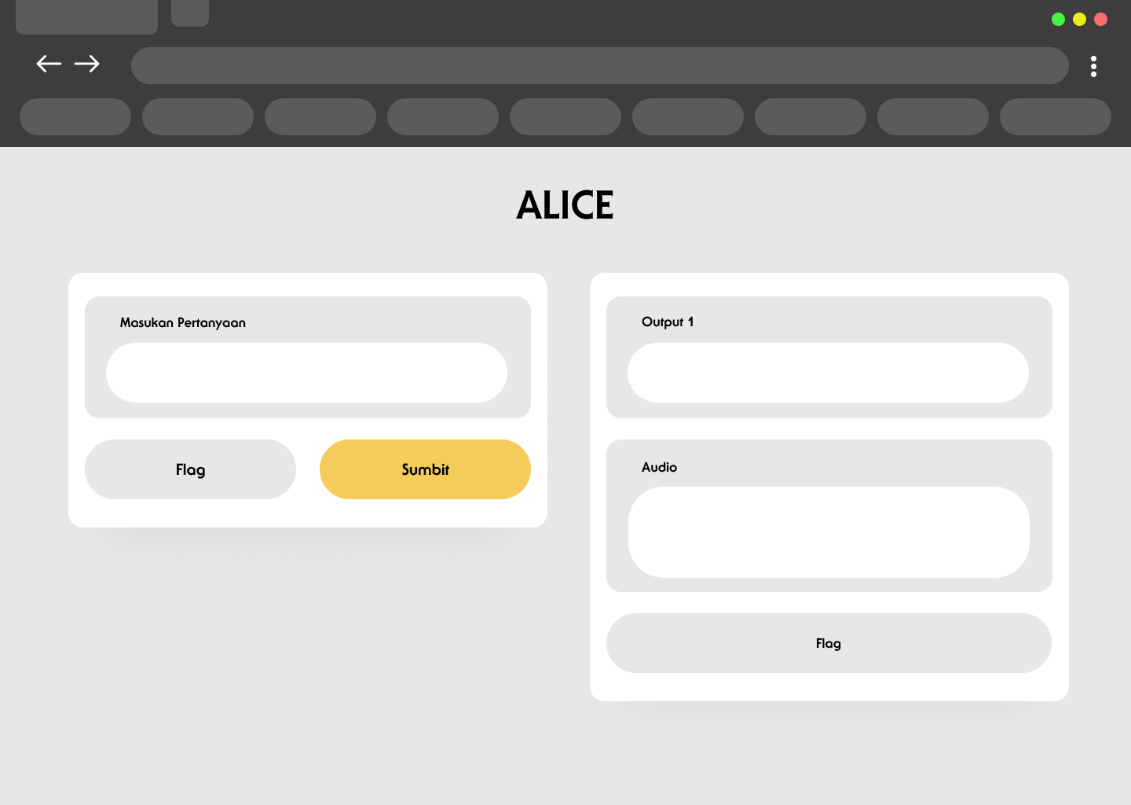


**Gambar 3.5:** *Flowchart Diagram Chatbot (ALICE)*

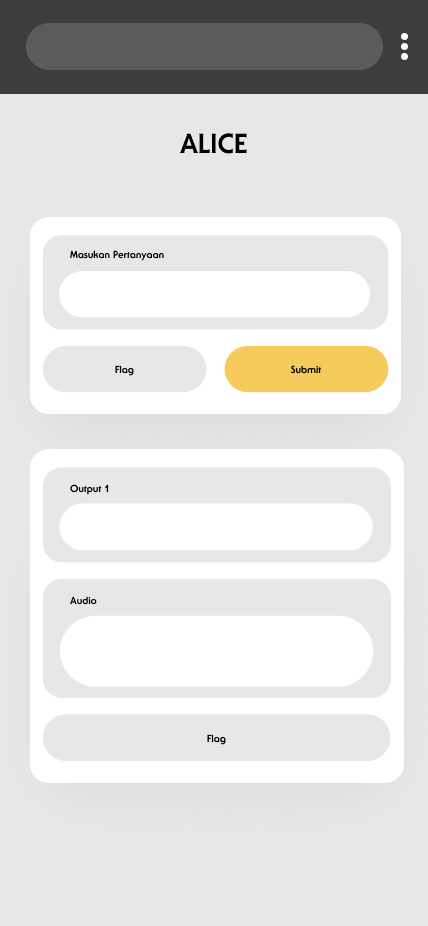
Berikut ini adalah penjelasan pada (Gambar 3.5) *Flowchart Diagram* di atas:

1. Langkah pertama, *user* masuk ke halaman *chatbot (ALICE)* menggunakan *link* yang sudah disediakan.
2. *User* meng-*input* pertanyaan yang ingin ditanyakan.
3. *Chatbot (ALICE)* memproses *input* pertanyaan dari *user*.
4. *Chatbot (ALICE)* akan memberikan jawaban yang sesuai *input* pertanyaan yang berupa teks dan suara.
5. Terakhir, *user* mendapatkan jawaban berupa teks dan suara dan menyesuikan jawaban dengan pertanyaan yang diinputkan.

### 3.5 Perancangan Antarmuka (*User Interface*)

****

**Gambar 3.6:** Halaman *Laptop* *Chatbot (ALICE)*

****

**Gambar 3.7:** Halaman *Smartphone* *Chatbot (ALICE)*

Gambar 3.6 dan gambar 3.7 adalah tampilan antarmuka *gradio app* *chatbot (ALICE)*, tampilan tersebut adalah tampilan yang di akses melalui *laptop* atau *smartphone* pengguna dan memberikan simulasi tanya jawab pada *chatbot (ALICE).*

## BAB 4 PENGUJIAN DAN IMPLEMENTASI

## DAFTAR PUSTAKA

Chandra, A. A., Nathaniel, V., Satura, F. R., Dharma Adhinata, F., & Studi, P. (n.d.). Pengembangan Chatbot Informasi Mahasiswa Berbasis Telegram dengan Metode Natural Language Processing. In *Jurnal ICTEE* (Vol. 3, Issue 1).

Furqan, M., & Naufal Shidqi, M. (2023). Chatbot Telegram Menggunakan Natural Language Processing. *Walisongo Journal of Information Technology*, *5*(1), 15–26. https://doi.org/10.21580/wjit.2023.5.1.14793

Pratama, R. A. (2022). Pemanfaatan Media Pembelajaran 3 Dimensi Untuk Materi Kecerdasan Buatan Dalam Mata KuliahKecerdasan Buatan. *https://jurnal.polsri.ac.id/*, 4.

Afriansyah, R., Sholeh, M., Andayati, D. (2021). Perancangan Aplikasi Pemrograman Antarmuka Berbasis Web Menggunakan Gaya Arsitektur Representasi Untuk Sistem Presensi Sekolah, 9(1), 84-93.

Setiawan, A. (2023). Penggunaan ChatGPT Untuk Pendidikan di Era Education 4.0. *PETISI*, 57.

Husamuddin, H. (2020). Otomatisasi Layanan *Frequensi Ask Questions* Berbasis *Natural Language Processing* pada Telegram Bot, 148.

Astiningrum, M. (2018). Implementasi NLP dengan Konversi Kata pada Sistem Chatbot Konsultan Laktasi. *Jurnal Informatika Polinema*, 47.

Clinton, R. M. (2019). Purwarupa Sistem Daftar Pelanggaran Lalulintas Berbasis Mini - Komputer Raspberry Pi. *Jurnal Teknik Elektro dan Komputer*, 183.

Permana, A. Y. (2019). Perancangan Sistem Informasi Penjualan Perumahan Mengunakan Metode Sdlc Pada Pt. Mandiri Land Prosperous Berbasis Mobile. *Jurnal Pelita Bangsa*, 155.

Gutama, R. (2021). Implementasi Scrum Pada Manajemen Proyek Pengembangan Aplikasi Sistem Monitoring Dan Evaluasi Pembangunan (SMEP). Yogjakarta: Universitas Islam Indonesia.

Ningrum, F. C. (2019). Pengujian Black Box pada Aplikasi Sistem Seleksi Sales Terbaik Menggunakan Teknik Equivalence Partitions. Jurnal Informatika Universitas Pamulang, 126.

Kurniawati. (2021). Penerapan Metode Waterfall Untuk Perancangan Sistem Informasi Inventory Pada Toko Keramik Bintang Terang. *PROSISKO*, 48.

Murcahyanto, H. (2023). Penerapan Media Chat GPT pada Pembelajaran Manajemen Pendidikan terhadap Kemandirian Mahasiswa. *Edumatic*, 117.

Putra, D. W. (2019). Unified Modelling Language (UML) dalam Perancangan Sistem Informasi Permohonan Pembayaran Restitusi SPPD. *TEKNOIF*, 33.

Aliman, W. (2021). Perancangan Perangkat Lunak Untuk Menggambar Diagram Berbasis Android. *Syntax Literate*, 3094.

Achlison, U. (2020). Analisis Implementasi Pengukuran Suhu Tubuh Manusia dalam Pandemi Covid-19 di Indonesia. Jurnal Ilmiah Komputer Grafis, 103.

Arianti, T. (2022). Perancangan Sistem Informasi Perpustakaan Menggunakan Diagram Uml (*Unified Modelling Language*). Jurnal Ilmiah Komputer Terapan Dan Informasi (Jikti), 20.

Prasetya, A. F. (2022). Perancangan Aplikasi Rental Mobil Menggunakan Diagram UML (*Unified Modelling Language*). Jurnal Ilmiah Komputer Terapan dan Informasi (JIKTI), 15.

Rosalina, A. (2020). Pengujian *Black Box* pada Sistem Informasi Penjualan HI Shoe Store Menggunakan Teknik *Equivalence Partitions*. Jurnal Informatika Universitas Pamulang, 26.

Purnomo, E. (2022). Pengembangan Aplikasi Alat Kesehatan Menggunakan Teknologi Realitas Tertambah Berbasis Android (Studi Kasus: Farouk Farisi Medika). *KALBISIANA*, 2908.

Kalsum, U. (2022). Pengenalan Kecerdasan Buatan (Artificial Intelligence) Kepada Para Remaja. *Jurnal Teknik Informatika*, 2.

Muhyidin, M. A. (2020). Perancangan *Ui/Ux* Aplikasi My Cic Layanan Informasi Akademik Mahasiswa Menggunakan Aplikasi *Figma*. *JURNAL DIGIT*, 211.

asd