

# Implementasi Chatbot FAQ pada Aplikasi Monev Kinerja Direktorat Jenderal Anggaran Menggunakan Framework Rasa Open Source

Arif Rachman\*, Iffatul Mardiyah, Miftahul Jannah

Program Studi Magister Manajemen Sistem Informasi, Universitas Gunadarma, Depok, Indonesia

Email: <sup>1</sup>\*emailearif@gmail.com, <sup>2</sup>iffatul@staff.gunadarma.ac.id, <sup>3</sup>miftah@staff.gunadarma.ac.id

Email Penulis Korespondensi: emailearif@gmail.com

**Abstrak**—Direktorat Jenderal Anggaran (DJA) merupakan unit organisasi pada Kementerian Keuangan dengan tugas untuk menyediakan sistem informasi terkait kinerja penganggaran. Dinamika perubahan kebijakan yang terjadi belakangan ini mengakibatkan perubahan sistem informasi yang telah dikembangkan. DJA sudah mensosialisasikan proses bisnis maupun sistem yang ada, namun masih banyak pengguna bertanya melalui saluran pelayanan pelanggan DJA yang hanya bisa merespon pada jam kerja. Penelitian ini akan mengusulkan solusi untuk optimalisasi layanan tersebut dengan membuat chatbot berbasis Natural Language Processing menggunakan framework Rasa Open Source yang akan dipasang pada salah satu sistem core DJA yaitu Aplikasi Monev Kinerja. Chatbot tersebut akan menjawab pertanyaan pengguna terkait dengan aplikasi tersebut secara spontan. Data yang digunakan pada penelitian ini adalah data Frequently Asked Questions (FAQ), knowledge base kemenkeupedia, Focus Group Discussion (FGD) dan data Aplikasi Monev Kinerja yang diambil melalui API (Application Programming Interface). Hasil penelitian ini adalah Chatbot FAQ yang disematkan pada aplikasi monev kinerja. Pengujian prediksi intent menghasilkan nilai akurasi 0.986, nilai weighted precision 0.973, recall 0.986, dan f1-score 0.980 kemudian prediksi respon menghasilkan nilai akurasi 0.980, nilai weighted precision 0.986, recall 0.980, dan f1-score 0.980. Hal ini menunjukkan bahwa chatbot mampu mengidentifikasi intent dengan sangat baik dan memberikan respon dengan tepat kepada pengguna.

**Kata Kunci:** Chatbot; FAQ; Natural Language Processing; Rasa Open Source; Application Programming Interface

**Abstract**—Direktorat Jenderal Anggaran (DJA) is an organizational unit within the Ministry of Finance with the task of providing an information system related to budgeting performance. The dynamics of policy changes that have occurred recently have resulted in changes to the information system that has been developed by DJA. DJA has socialized the existing business processes and systems, but many users still ask questions through the DJA customer service channel which can only respond during business hours. This research will propose a solution for optimizing these services by creating a chatbot based on Natural Language Processing using the Rasa Open Source framework, which will be installed on one of the DJA's core systems, namely the Performance Monitoring and Evaluation Application. The chatbot will spontaneously answer user questions related to the application. The data used in this study are Frequently Asked Questions (FAQ) data, knowledge base Kemenkeupedia, Focus Group Discussions (FGD) and Performance Monev Application data taken via the API (Application Programming Interface). The results of this study are Chatbot FAQs embedded in the performance monitoring and evaluation application. The intent prediction test produces an accuracy value of 0.986, a weighted precision value of 0.973, a recall of 0.986, and an f1-score of 0.980 then the response prediction produces an accuracy value of 0.980, a weighted precision value of 0.986, a recall of 0.980, and an f1-score of 0.980. This shows that the chatbot is able to identify intent very well and respond appropriately to the user.

**Keywords:** Chatbot; FAQ; Natural Language Processing; Rasa Open Source; Application Programming Interface

## 1. PENDAHULUAN

Direktorat Jenderal Anggaran (Ditjen Anggaran) adalah unit organisasi di bawah Kementerian Keuangan. Salah satu fungsi dari Ditjen Anggaran adalah untuk menyediakan sistem informasi terkait kinerja anggaran. Kinerja anggaran adalah capaian Kinerja atas penggunaan anggaran Kementerian/Lembaga yang tertuang dalam dokumen anggaran [1]. Berdasarkan Enterprise Arsitektur Ditjen Anggaran tahun 2022, Aplikasi Monev Kinerja SMART (Sistem Monitoring dan Evaluasi Kinerja Terpadu) termasuk salah satu aplikasi utama. Pengguna dari aplikasi tersebut adalah seluruh satuan kerja yang ada di seluruh Kementerian/Lembaga. Jumlah satuan kerja pada tahun 2023 adalah sekitar 19.000 satuan kerja. Dinamika perubahan kebijakan akhir-akhir ini menyebabkan perubahan pada sistem informasi yang dikembangkan oleh Ditjen Anggaran. Proses bisnis maupun sistem yang ada sudah disosialisasikan oleh Ditjen Anggaran, akan tetapi masih banyak pengguna yang bertanya melalui saluran pelayanan pelanggan Ditjen Anggaran yaitu SAPA Anggaran.

Ditjen Anggaran perlu meningkatkan peran teknologi untuk meningkatkan pelayanan kepada pengguna. Perlu pembangunan chatbot yang bisa menjawab pertanyaan pengguna sistem informasi secara cepat dan tepat. Chatbot yang bisa memangkas proses bisnis pelayanan pelanggan yang sudah ada. Teknologi terbaru yang bisa diterapkan adalah teknologi kecerdasan buatan atau *Artificial Intelligence* (AI). AI adalah kemampuan sistem untuk mengartikan data, belajar dari data dan menggunakan pembelajaran tersebut untuk mencapai tujuan dan tugas tertentu melalui proses adaptasi yang fleksibel [2]. Teknologi kecerdasan buatan diimplementasikan agar komputer mampu melakukan pekerjaan seperti manusia. Pengembangan chatbot menggunakan kecerdasan buatan untuk mengenali bahasa alami manusia dan agar bisa dipahami oleh mesin. Hal ini biasa disebut dengan *Natural Language Processing* (NLP). NLP adalah bidang di persimpangan ilmu komputer, kecerdasan buatan, dan linguistik. NLP berfokus pada pembangunan sistem yang dapat memproses dan memahami bahasa manusia [3]. Chatbot ini akan disematkan pada aplikasi monev kinerja Direktorat Jenderal Anggaran dalam bentuk widget.

Chatbot adalah program komputer yang dirancang untuk berkomunikasi dengan satu atau lebih pengguna melalui media teks atau audio [4]. Chatbot diprogram dengan kumpulan aturan dan algoritma tertentu untuk merespons masukan atau pertanyaan yang diterima dari pengguna. Chatbot dapat dibangun dengan menggunakan Natural Language

Processing yang merupakan salah satu bidang teknologi kecerdasan buatan yang mempelajari komunikasi manusia dengan komputer dengan bahasa alami [5]. Chatbot ini bisa juga diintegrasikan dengan platform chat seperti whatsapp, telegram, facebook messenger, line, dan juga web [6]. Chatbot dapat membantu pengguna dengan cepat dan tepat tanpa perlu melibatkan manusia secara langsung. Dengan adanya chatbot, interaksi antara manusia dan teknologi semakin mudah dan efisien.

FAQ atau Frequently Asked Question adalah pertanyaan yang sering ditanyakan oleh pengguna yang biasanya dijawab oleh manusia [7]. Namun, dengan kemajuan teknologi dan kecerdasan buatan, FAQ dapat dijawab secara otomatis oleh chatbot atau asisten virtual [8]. Hal ini memungkinkan pengguna untuk mendapatkan jawaban yang cepat dan efektif tanpa harus menunggu respon dari manusia. Meskipun demikian, FAQ tetap harus dikelola dengan baik dan diperbarui secara berkala agar informasi yang disajikan tetap akurat dan sesuai dengan kebutuhan pengguna. Dalam beberapa kasus, FAQ juga dapat membantu perusahaan dalam mengurangi jumlah pertanyaan yang harus dijawab oleh karyawan secara langsung, sehingga memungkinkan mereka untuk lebih fokus dalam tugas-tugas lainnya [9].

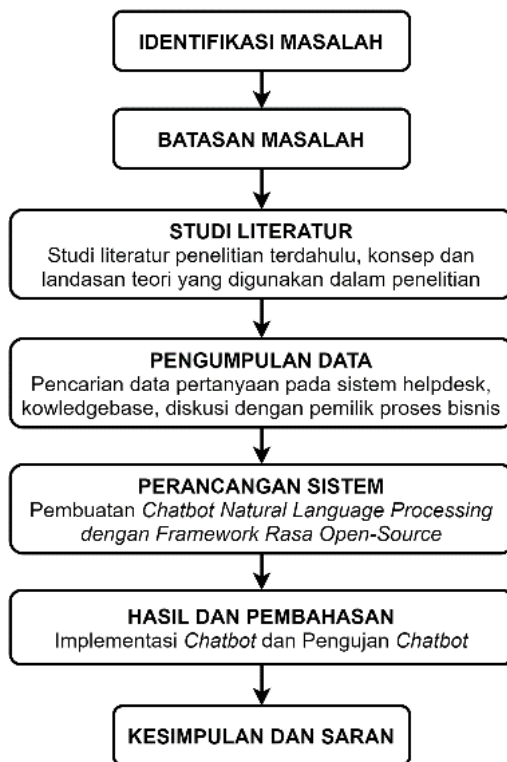
Rasa Open Source adalah framework pembangunan chatbot yang bersifat open source. Rasa Open Source memiliki keunggulan yaitu gratis, tidak ada pembatasan vendor, tidak ada resiko kebocoran data, kemampuan untuk menjalankan fungsi khusus, dokumentasi yang lengkap, komunitas global yang besar serta memiliki fungsi yang lengkap [10]. Rasa Open Source bisa dihubungkan dengan berbagai channel komunikasi seperti facebook messenger, telegram, google hangout chat, slack, twilio, dan website dalam bentuk widget. Berdasarkan pengujian yang dilakukan oleh Jiao [11], dari sisi ekstraksi entitas, metode yang digunakan pada Framework ini memiliki tingkat akurasi yang lebih baik daripada metode Recurrent Neural Network (RNN). Penelitian Singh yang membandingkan Framework Rasa dengan Dialog Flow menghasilkan kesimpulan bahwa Framework Rasa yang paling banyak dipilih oleh pengguna [12]. Karena pertimbangan tersebut peneliti menggunakan framework Rasa Open Source.

Penelitian terdahulu terkait penggunaan Natural Language Processing Rasa Open-Source Framework adalah penelitian oleh Zein Hanni Pradana, Hanin Nafi'ah, Raditya dan Artha Rochmanto tentang Pelayanan informasi berbasis chatbot yang menggunakan Framework RASA Open-Source di objek wisata candi prambanan [13]. Penelitian tersebut berhasil mendapatkan akurasi sebesar 0,91, presisi 0,97, recall 0,94 dan F-1 Score 0,95. Penelitian lain yang dilakukan oleh I Ketut Resika Arthana, Luh Joni Erawati Dewi, Ketut Agus Seputra dan Ni Wayan Marti tentang Undiksha Virtual Assistant (Shavira): Integration Frequency Asked Question with Rasa Framework dengan hasil pengujian mendapatkan akurasi sebesar 0,90 pengguna merasa puas terhadap chatbot tersebut [14]. Terdapat juga penelitian dari Dirko G. S. Ruindungan dan Agustinus Jacobus yaitu Pengembangan Chatbot untuk Layanan Informasi Interaktif Akademik menggunakan Framework Rasa Open Source, hasil evaluasi chatbot tersebut menghasilkan akurasi yang sangat tinggi yaitu 0,995 [15]. Penelitian lain menunjukkan bahwa chatbot dapat membantu mengurangi beban petugas administrasi [9]. Berdasarkan beberapa penelitian terdahulu, penelitian ini bertujuan untuk membangun Chatbot berbasis Natural Language Processing pada Aplikasi Monev Kinerja Ditjen Anggaran guna meningkatkan kualitas pelayanan kepada pengguna sistem informasi Ditjen Anggaran.

## **2. METODOLOGI PENELITIAN**

### **2.1 Tahapan Penelitian**

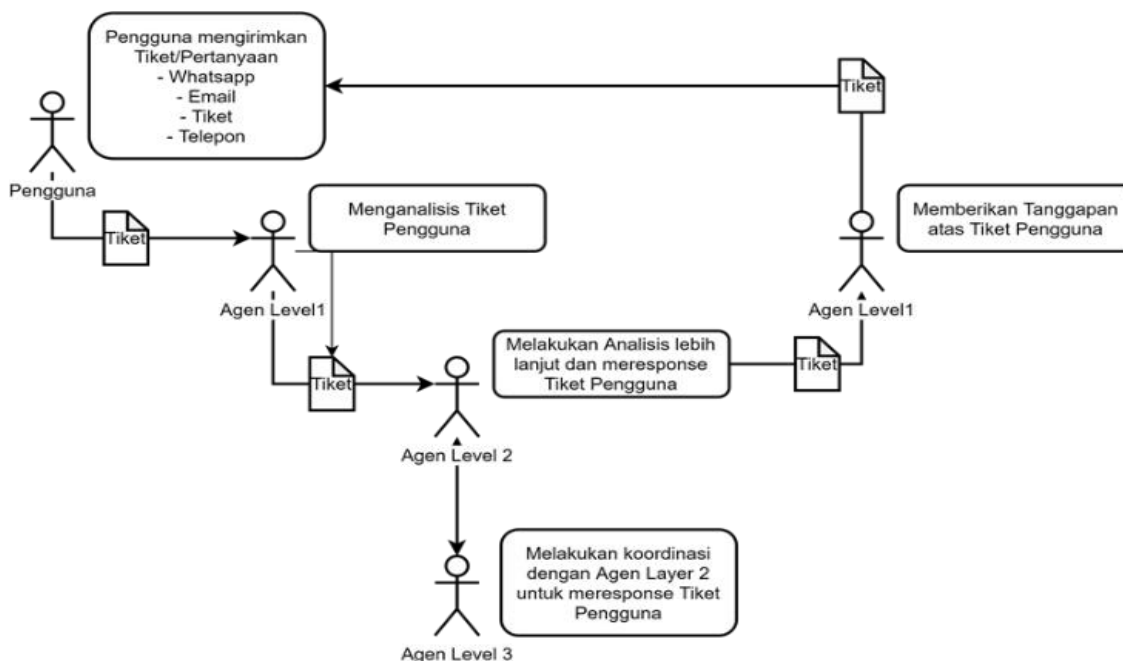
Tahap pertama penelitian ini dimulai dari identifikasi masalah yaitu melakukan analisis terhadap masalah yang ada pada sistem berjalan, dilanjutkan dengan batasan masalah agar penelitian ini menjadi lebih fokus dengan membatasi FAQ pada satu sistem yaitu aplikasi monev kinerja, kemudian melakukan studi literatur penelitian terdahulu, konsep dan landasan teori. Selanjutnya pengumpulan data pertanyaan, knowledgebase dan data pada aplikasi monev kinerja, yang selanjutnya dibahas pada Focus Group Discussion (FGD) dengan pemilik proses bisnis. Data tersebut disusun dalam bentuk daftar pertanyaan, topik dan jawaban. Kemudian dilakukan proses pembuatan chatbot dengan pendekatan Natural Language Processing dari data sebelumnya dengan menggunakan Framework Rasa Open Source dan mengimplementasikan chatbot tersebut. Tahapan penelitian digambarkan pada Gambar 1.



Gambar 1. Tahapan Penelitian

## 2.2 Analisis Sistem Berjalan

Layanan melalui tiket pada SAPA Anggaran dianalisis oleh agen level 1. Agen level 1 memilah tiket dan mendistribusikan tiket sesuai dengan aplikasi terkait, selanjutnya agen level 1 dan/atau level 2 melakukan analisis atas pertanyaan atau permintaan layanan. Dalam hal permintaan layanan atau jawaban atas pertanyaan telah terdapat pada knowledge base Kemenkeupedia maka agen level 1 dapat memberikan jawaban atas permintaan layanan tersebut dan jika jawaban atas permintaan layanan tidak tersedia maka agen meneruskan permintaan layanan ke agen level 2 dan 3. Seluruh proses layanan dilakukan oleh agen pada hari dan jam kerja dan dilakukan secara manual.



Gambar 2. Sistem Berjalan

## 2.2 Kelemahan Sistem Berjalan

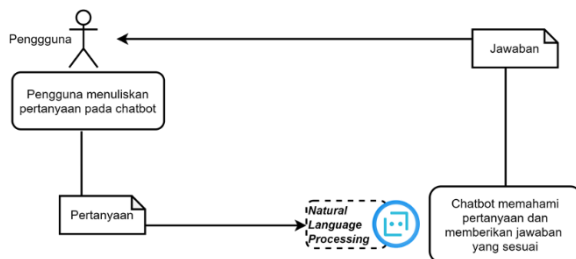
Kelemahan sistem berjalan adalah sebagai berikut:

- Layanan hanya diakses pada jam kerja.

- b. Layanan harus melewati agen level 1 untuk selanjutnya diteruskan ke agen level 2 sesuai dengan kategori pertanyaan secara manual.
- c. Proses tidak efisien karena agen harus menjawab pertanyaan sederhana yang sama dan berulang.

### 2.3. Analisis Sistem yang dikembangkan

Sistem yang diusulkan dalam penelitian ini adalah chatbot dengan pendekatan Natural Language Processing yang disematkan pada sistem informasi. Chatbot ini berupa widget chat yang dapat digunakan pengguna sistem untuk bertanya terkait dengan sistem informasi tersebut, kemudian chatbot akan menjawab pertanyaan tersebut dengan pengetahuan yang sudah diberikan sebelumnya. Proses pemberian pengetahuan dilakukan dengan pendekatan Natural Language Processing.



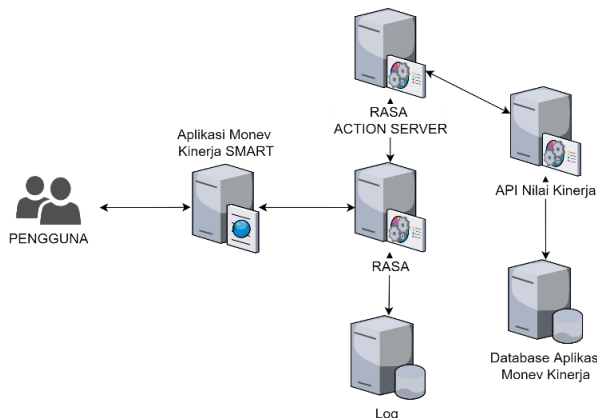
Gambar 3. Sistem yang dikembangkan

### 2.3. Desain Sistem

Pada tahap desain sistem dijelaskan mengenai arsitektur chatbot yang akan dibangun beserta desain sistem dengan menggunakan UML. UML (Unified Modeling Language) adalah metode pemodelan visual yang digunakan dalam desain sistem berorientasi objek [16]. Model sistem yang digunakan adalah Use Case Diagram, Activity Diagram dan Sequence Diagram.

#### a. Arsitektur Chatbot

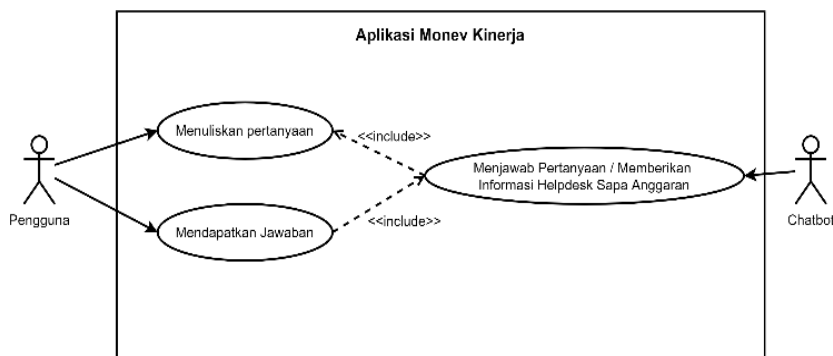
Chatbot disematkan pada aplikasi monev kinerja. Setelah pengguna mengakses sistem, akan muncul widget chat pada bagian kanan bawah. Widget tersebut terkoneksi dengan server chatbot. Arsitektur chatbot dijelaskan pada Gambar 4.



Gambar 4. Arsitektur Chatbot

#### b. Use Case Diagram

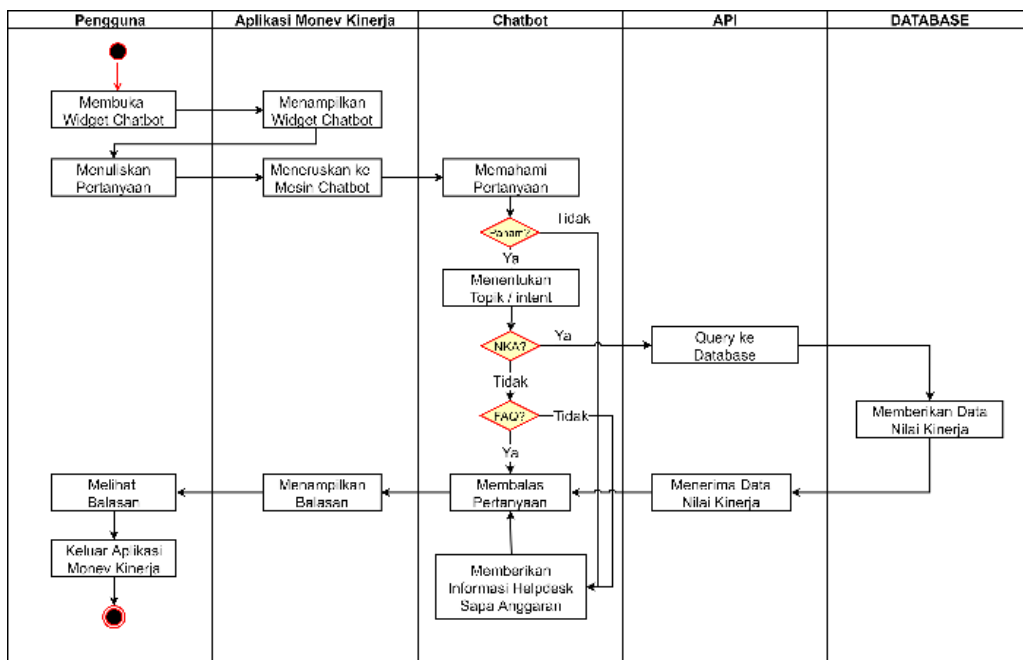
Use Case Diagram adalah sebuah UML diagram yang dapat merepresentasikan interaksi yang terjadi antara pengguna dengan sistem chatbot. Use Case Diagram dari sistem yang akan dibangun dalam dilihat pada Gambar 5.



Gambar 5. Use Case Diagram

c. Activity Diagram

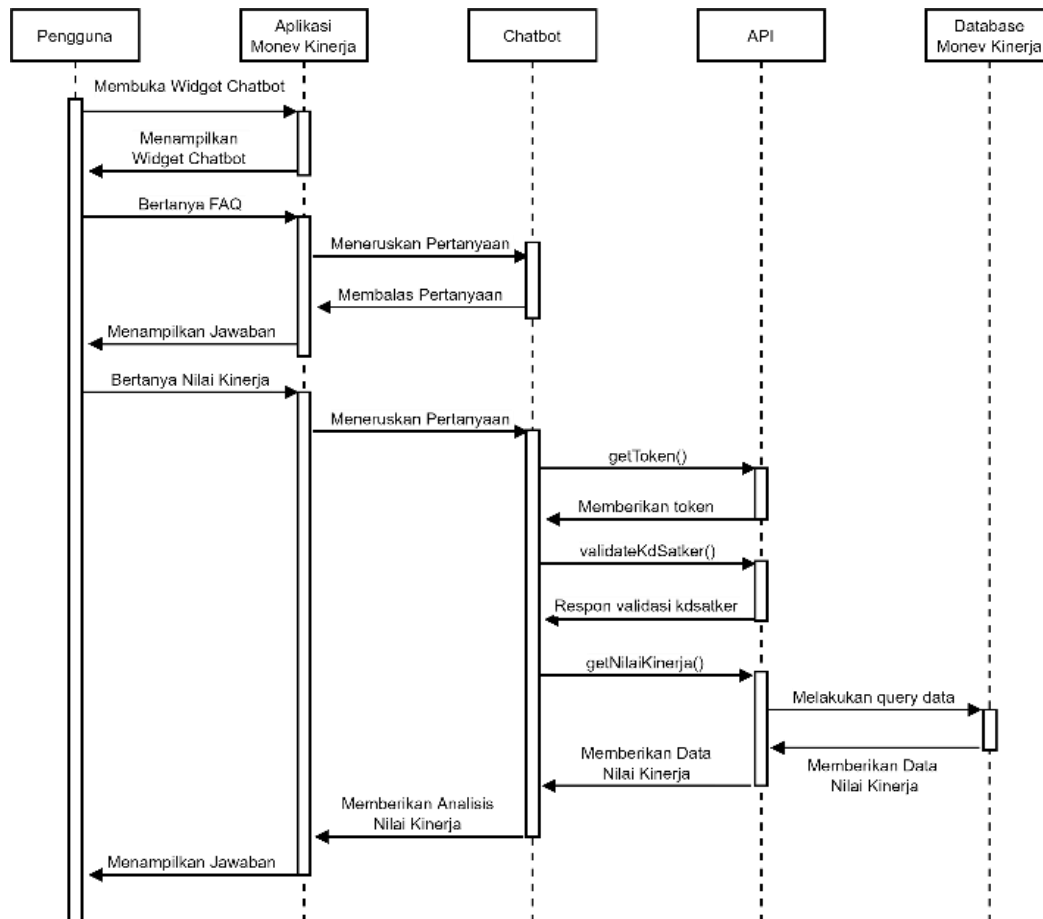
Activity diagram adalah diagram yang menggambarkan aktivitas dalam sebuah sistem yang sedang berjalan. Activity diagram dari sistem yang akan dibangun dapat ditunjukkan pada Gambar 6.



Gambar 6. Activity Diagram

d. Sequence Diagram

Sequence diagram adalah suatu diagram yang menggambarkan interaksi antar objek pada sistem dalam sebuah urutan waktu atau rangkaian waktu. Sequence diagram dari sistem yang akan dibangun ditunjukkan pada Gambar 7.



Gambar 7. Sequence Diagram

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Pembangunan Chatbot

Pembangunan chatbot dengan pendekatan *Natural Language Processing* (NLP) menggunakan Framework Rasa Open Source. Proses yang dilakukan yaitu dengan mengumpulkan data pertanyaan dan jawaban, cleansing data, mengubah data sesuai format framework Rasa Open Source, kemudian membuat model dan menggunakan model tersebut sebagai mesin penjawab widget chatbot yang ada pada aplikasi monev kinerja.

a. Pengumpulan Data.

Data yang dibutuhkan dalam pembangunan chatbot meliputi data pertanyaan oleh pengguna dan jawaban yang disampaikan agen SAPA Anggaran, data knowledge base, data focus group discussion (FGD) dan data nilai kinerja anggaran. Data diperoleh melalui permohonan resmi pada Sub Bagian Layanan Anggaran dan Tata Usaha, Bagian Umum, Sekretariat Jenderal, Direktorat Jenderal Anggaran.

b. Cleansing Data.

Pada tahap cleansing data dilakukan pengolahan data tidak lengkap dan data seragam agar diperoleh data bersih yang sesuai konteks dan siap digunakan untuk proses selanjutnya.

c. Format Data.

Data dirubah dalam format YAML Ain't a Markup Language (YAML) sesuai dengan standard framework Rasa Open Source. YAML adalah bahasa yang sering digunakan pada file konfigurasi pipeline [18]. Data training yang digunakan untuk pelatihan model pada framework Rasa Open Source terdiri dari beberapa bentuk sebagai berikut.

1. Intents

Maksud yang coba disampaikan atau dicapai pengguna. Contoh: menyapa, bertanya terkait formula nilai kinerja anggaran.

```
- intent: faq/smart_login
  examples: |
    - bagaimana login aplikasi smart?
    - alamat login aplikasi smart
    - login smart
    - apa password default smart
    - apa username default smart
    - bagaimana login aplikasi smart pertama sekali
    - cara login aplikasi smart
    - kendala login aplikasi smart
- intent: faq/smart_lupa_password
  examples: |
    - saya lupa password smart
    - bagaimana kalo saya lupa password smart
    - lupa password aplikasi smart
    - bagaimana jika saya lupa email dan password
    - lupa email dan password aplikasi smart
```

Gambar 8. Contoh Intents

2. Response

Pesan yang dikirim asisten ke pengguna. Response dapat berupa teks, tombol, gambar, dan konten lainnya.

```
responses:
  utter_faq/smart_login:
    - text: |
      Login aplikasi SMART dilakukan melalui situs monev.kemenkeu.go.id dengan username dan password default sebagai berikut:
      Level satker
      username: me+kodesatker+00
      password: me+kodesatker+00
      Contoh: me12345600
      [Login] (https://monev.kemenkeu.go.id/)
  utter_faq/smart_lupa_password:
    - text: |-
      Jika mengalami lupa password untuk login ke laman SMART, pengguna dapat melakukan langkah-langkah berikut.
      1. Klik [Lupa Password?].
      2. Masukkan username default dan alamat email terdaftar.
      3. Masukkan kode keamanan/captcha.
      4. Klik tombol Kirim Email.
      5. Buka kotak masuk alamat email terdaftar.
      6. Buka email terkait reset password dan catat/simpan password baru yang sudah di-generate oleh sistem.
      7. Login ke SMART dengan username default dan password baru.
      [Link Reset] (https://monev.kemenkeu.go.id/user/formresetjendela)
```

Gambar 9. Contoh Response

3. Action

Aksi yang dilakukan bot dalam percakapan. Contoh: memanggil API atau mengirim respons kembali ke pengguna.

4. Stories

Jenis data pelatihan yang digunakan untuk melatih model. Stories dapat digunakan untuk melatih model yang mampu menggeneralisasi ke jalur percakapan yang tidak terlihat.

5. Rules

Rules adalah jenis data pelatihan yang digunakan untuk melatih model manajemen dialog. Rules menggambarkan percakapan pendek yang harus selalu mengikuti jalur yang sama.

6. Domain

Domain mendefinisikan wilayah chatbot bekerja. Domain terdiri dari intents, response, entities dan action.

7. Entities

Kata kunci yang dapat diekstraksi dari pesan pengguna. Contoh: nomor telepon, nama orang, lokasi, nama satuan kerja.

Tabel 1. Dataset Chatbot

No	Intent	Jumlah Sampel
1.	general greeting	14
2.	islamic greeting	9
3.	general not answering	16
4.	general thanks	11
5.	general agree	16
6.	faq/smart login	8
7.	faq/smart lupa password	5
8.	smart reset password	6
9.	faq/smart perhitungan kinerja	14
10.	faq/smart perhitungan penyerapan	19
11.	faq/smart perhitungan konsistensi	21
12.	faq/smart perhitungan cro	23
13.	faq/smart perhitungan efisiensi	28
14.	faq/smart pengisian satker	13
15.	faq/smart pengisian unit	17
16.	faq/smart pengisian kl	14
17.	faq/smart update data	15
18.	faq/smart capaian kuning	6
19.	faq/smart bobot indikator	17
20.	faq/smart batas pengisian	8
21.	faq/smart nilai turun	9
22.	faq/smart informasi kinerja	19
23.	faq/smart tidak bisa input	15
24.	smart tanya nilai kinerja	17
25.	inform kodesatker	40
26.	faq/bot challenge	4
27.	out of scope	3

d. Modeling

Modeling menggunakan framework Rasa Open Source dengan perintah shell “**rasa train**”. Gambar 10 menunjukkan konfigurasi yang digunakan dalam pembuatan model.

```
1 recipe: default.v1
2 assistant_id: 26230327-113525-basic-scuba
3 language: en
4 pipeline:
5 * * No configuration for the NLU pipeline was provided. The following default pipeline
6 * * was used to train your model.
7 * * If you'd like to customize it, uncomment and adjust the pipeline.
8 * * See https://rasa.com/docs/rasa/tuning-your-model for more information.
9 - name: WhitespaceTokenizer
10 - name: CountVectorsFeaturizer
11 - name: CountVectorsFeaturizer
12 analyzer: char_wb
13 min_ngram: 1
14 max_ngram: 4
15 - name: DIETClassifier
16 epochs: 100
17 constrain_similarities: true
18 * - name: EntitySynonymMapper
19 - name: ResponseSelector
20 epochs: 100
21 constrain_similarities: true
22 retrieval_intent: faq
23 - name: FallbackClassifier
24 threshold: 0.7
25 ambiguity_threshold: 0.1
26 policies:
27 * * No configuration for policies was provided. The following default policies were
28 * * used to train your model.
29 * * If you'd like to customize them, uncomment and adjust the policies.
30 * * See https://rasa.com/docs/rasa/policies for more information.
31 - name: MemoizationPolicy
32 - name: RulePolicy
33 - name: UnexpectIntentPolicy
34 max_history: 5
35 epochs: 100
36 - name: TEDPolicy
37 max_history: 5
38 epochs: 100
39 constrain_similarities: true
```

Gambar 10. Konfigurasi Model

Penjelasan pipeline NLU dan pipeline policies ada pada Tabel 2 dan 3.

**Tabel 2.** Pipeline NLU

Pipeline NLU	Penjelasan
WhitespaceTokenizer	Tokenizer adalah pemisahan teks menjadi kata-kata / tokens. Whitespace tokenizer adalah tokenizer yang menggunakan pemisah spasi.
CountVectorsFeaturizer	CountVectorsFeaturizer digunakan untuk membuat representasi kumpulan kata-kata dari pesan, maksud, dan respons pengguna.
DIETClassifier	Dual Intent Entity Transformer (DIET) digunakan untuk klasifikasi maksud dan ekstraksi entitas.
ResponseSelector	ResponseSelector digunakan untuk pemilihan respon.
FallbackClassifier	FallbackClassifier digunakan untuk mengklasifikasikan pesan pada intent nlu_fallback jika skor klasifikasi maksud NLU di bawah ambang fallback pada file konfigurasi.

**Tabel 3.** Pipeline Policies

Pipeline NLU	Penjelasan
MemoizationPolicy	MemoizationPolicy mengingat alur percakapan data training. Aturan ini memeriksa apakah percakapan saat ini cocok dengan alur percakapan pada file stories.yml. Jika demikian, sistem akan memprediksi tindakan selanjutnya sesuai alur percakapan pada story.yml.
RulePolicy	RulePolicy adalah aturan yang menangani percakapan pendek sesuai data training pada file rules.yml.
UnexpectEDIntentPolicy	UnexpectEDIntentPolicy adalah arsitektur multi tugas untuk prediksi tindakan yang tidak diharapkan.
TEDPolicy	TEDPolicy adalah arsitektur multi tugas untuk prediksi tindakan selanjutnya dan pengenalan entitas.

### 3.2 Pengembangan API

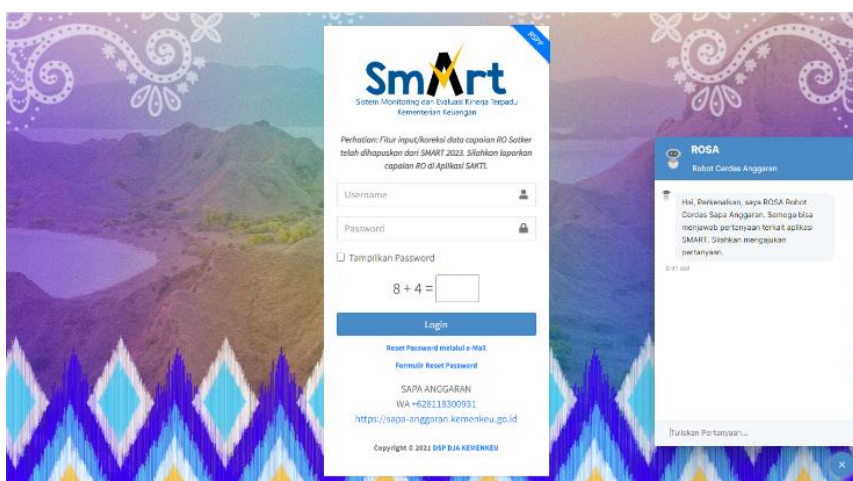
Untuk mendukung pengembangan chatbot pada aplikasi monev kinerja, peneliti membutuhkan API. *Application Programming Interface* (API) adalah seperangkat metode perangkat lunak yang memungkinkan dua aplikasi berkomunikasi satu sama lain melalui seperangkat aturan yang ditentukan oleh penyedia [17]. API diperlukan untuk melakukan validasi kode satuan kerja dan mengambil data nilai kinerja anggaran satuan kerja. Proses ini selanjutnya akan disematkan pada API Monev Kinerja dengan menambahkan dua service baru yang ditunjukkan pada Tabel 4.

**Tabel 4.** Service API Monev Kinerja

Method	Service	Keterangan
POST	ValidSatker()	Service untuk melakukan validasi kode satker
POST	NilaiKinerjaSatker()	Service untuk mendapatkan data nilai kinerja anggaran satuan kerja

### 3.3 Pengembangan Widget Chat

Tahap selanjutnya adalah pembuatan widget chat pada aplikasi monev kinerja yang terhubung dengan mesin chatbot. Berikut tampilan widget chat pada halaman login aplikasi monev kinerja.



**Gambar 11.** Widget Chat

### 3.4 Pengujian API

Pengujian API menggunakan metode Black Box. Pengujian Black Box memastikan semua proses sesuai dengan kebutuhan awal [19] Peneliti membuat beberapa *test case* untuk mengecek apakah API berjalan sesuai dengan spesifikasi kebutuhan awal. Hasil pengujian menunjukkan bahwa seluruh fungsi berjalan sesuai dengan kebutuhan awal.

### 3.5 Pengujian Chatbot

Proses penilaian akurasi dilakukan dengan melihat persentase prediksi intent dan respon yang berhasil diprediksi dengan tepat oleh chatbot. Berikut tahap-tahap pengujian chatbot.

#### a. Pembagian Dataset

Total dataset ada 373 data. Pada tahap ini dilakukan pembagian data training dengan data testing menggunakan framework rasa open source dengan perintah **“`rasa data split nlu`”**. Perintah tersebut membagi data pertanyaan menjadi 80% data training dan 20% data testing [20]. Data training berjumlah 300 data dan data testing berjumlah 73 data. Pemilihan data training dan testing dilakukan secara acak.

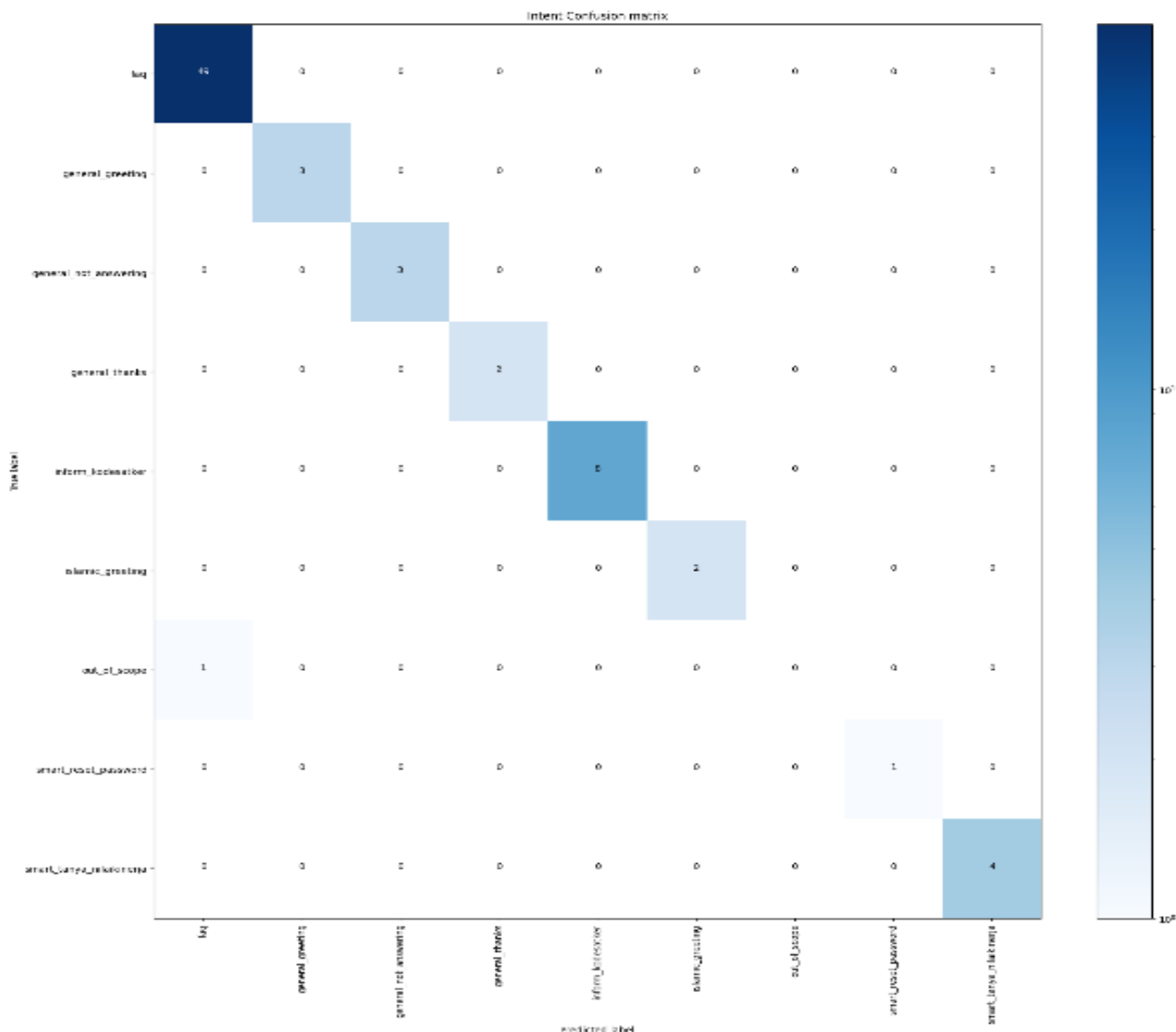
#### b. Pembuatan model dengan data testing

Pembuatan model dari data training dilakukan dengan perintah **“`rasa train nlu --nlu .\train_test_split(training_data.yml)`”**.

#### c. Pengujian model dengan data testing

Pengujian model yang sudah dibuat sebelumnya pada data testing dengan perintah **“`rasa test nlu --nlu .\train_test_split(test_data.yml)`”**. Framework Rasa Open Source secara otomatis membuat hasil prediksi intent dan respon dalam bentuk Confusion Matrix.

#### d. Confusion Matrix Prediksi Intent



Gambar 12. Confusion Matrix Prediksi Intent



## REFERENCES

- [1] Peraturan Menteri Keuangan Republik Indonesia, “Peraturan Menteri Keuangan Republik Indonesia Nomor 22/PMK.02/2021 tentang Pengukuran dan Evaluasi Kinerja Anggaran atas Pelaksanaan Rencana Kerja dan Anggaran Kementerian Negara/Lembaga,” Jakarta, 2021. [Online]. Available: [www.jdih.kemenkeu.go.id](http://www.jdih.kemenkeu.go.id)
- [2] A. Kaplan and M. Haenlein, “Siri, Siri, in my hand: Who’s the fairest in the land? On the interpretations, illustrations, and implications of artificial intelligence,” *Bus Horiz*, vol. 62, no. 1, pp. 15–25, Jan. 2019, doi: 10.1016/j.bushor.2018.08.004.
- [3] S. Vajjala, B. Majumder, A. Gupta, and H. Surana, “Practical Natural Language Processing A Comprehensive Guide to Building Real-World NLP Systems,” 2020.
- [4] Achmad Musyaffa Taufiqi, *Membangun Mesin Percakapan Pintar dengan Python dan Telegram Bot*. Deepublish, 2021.
- [5] S. Mulyatun, H. Utama, and A. Mustopa, “PENDEKATAN NATURAL LANGUAGE PROCESSING PADA APLIKASI CHATBOT SEBAGAI ALAT BANTU CUSTOMER SERVICE,” 2021.
- [6] V. Boddula, A. Rao Polkampally, S. Sohail, and S. Sumair Pasha, “CHATBOT USING RASA,” 2022. [Online]. Available: [www.irjmet.com](http://www.irjmet.com)
- [7] R. Vannala, S. B. Swathi, and Y. Puranam, “AI Chatbot For Answering FAQ’s,” in 2022 IEEE 2nd International Conference on Sustainable Energy and Future Electric Transportation (SeFeT), IEEE, Aug. 2022, pp. 1–5. doi: 10.1109/SeFeT55524.2022.9908774.
- [8] F. Sethi, “FAQ (Frequently Asked Questions) ChatBot for Conversation,” *International Journal of Computer Sciences and Engineering Open Access Research Paper*, vol. 8, no. 10, 2020, doi: 10.26438/ijcse/v8i10.710.
- [9] K. Lee, J. Jo, J. Kim, and Y. Kang, “Can Chatbots Help Reduce the Workload of Administrative Officers? - Implementing and Deploying FAQ Chatbot Service in a University,” in *Communications in Computer and Information Science*, Springer Verlag, 2019, pp. 348–354. doi: 10.1007/978-3-030-23522-2\_45.
- [10] X. Kong and G. Wang, *Conversational AI with Rasa : build, automate, and deploy AI-powered text and voice-based assistants and chatbots*. 2021.
- [11] A. Jiao, “An Intelligent Chatbot System Based on Entity Extraction Using RASA NLU and Neural Network,” *J Phys Conf Ser*, vol. 1487, no. 1, p. 012014, Mar. 2020, doi: 10.1088/1742-6596/1487/1/012014.
- [12] S. Singh and S. Singh, “EasyChair Preprint Effective Analysis of Chatbot Frameworks: RASA and Dialogflow,” 2022. [Online]. Available: <https://rasa.com/>
- [13] Zein Hanni Pradana, Hanin Nafi’ah, and Raditya Artha Rochmanto, “Chatbot-based Information Service using RASA Open-Source Framework in Prambanan Temple Tourism Object,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 6, no. 4, pp. 656–662, Aug. 2022, doi: 10.29207/resti.v6i4.3913.
- [14] I. K. R. A. Arthana, L. J. E. D. Dewi, K. A. Seputra, and N. W. Marti, “Undiksha Virtual Assistant (Shavira): Integration Frequency Asked Question with Rasa Framework,” 2021.
- [15] D. G. S. Ruindungan and A. Jacobus, “Chatbot Development for an Interactive Academic Information Services using the Rasa Open Source Framework,” *Jurnal Teknik Elektro dan Komputer*, vol. 10, 2021.
- [16] V. Yasin, “Tools Rekayasa Perangkat Lunak dalam Membuat Pemodelan Desain Menggunakan Unified Modeling Language (UML),” *TRIDHARMADIMAS: Jurnal Pengabdian Kepada Masyarakat Jayakarta*, vol. 1, no. 2, p. 139, Dec. 2021, doi: 10.52362/tridharmadimas.v1i2.666.
- [17] S. Jonnada and J. K. Joy, “Measure your API Complexity and Reliability,” in 2019 IEEE 17th International Conference on Software Engineering Research, Management and Applications (SERA), IEEE, May 2019, pp. 104–109. doi: 10.1109/SERA.2019.8886790.
- [18] C. Chandrasekara and P. Herath, “Creating and Using YAML Build Pipelines,” in *Hands-on Azure Pipelines*, Berkeley, CA: Apress, 2020, pp. 123–142. doi: 10.1007/978-1-4842-5902-3\_8.
- [19] Y. Dwi Wijaya and M. Wardah Astuti, “Pengujian Blackbox Sistem Informasi Penilaian Kinerja Karyawan Pt Inka (Persero) Berbasis Equivalence Partitions Blackbox Testing Of Pt Inka (Persero) Employee Performance Assessment Information System Based On Equivalence Partitions,” *Jurnal Digital Teknologi Informasi*, vol. 4, p. 2021, 2021.
- [20] Rasa Technologies GmbH, “Testing Your Assistant.” <https://rasa.com/docs/rasa/testing-your-assistant/> (accessed Jul. 13, 2023).