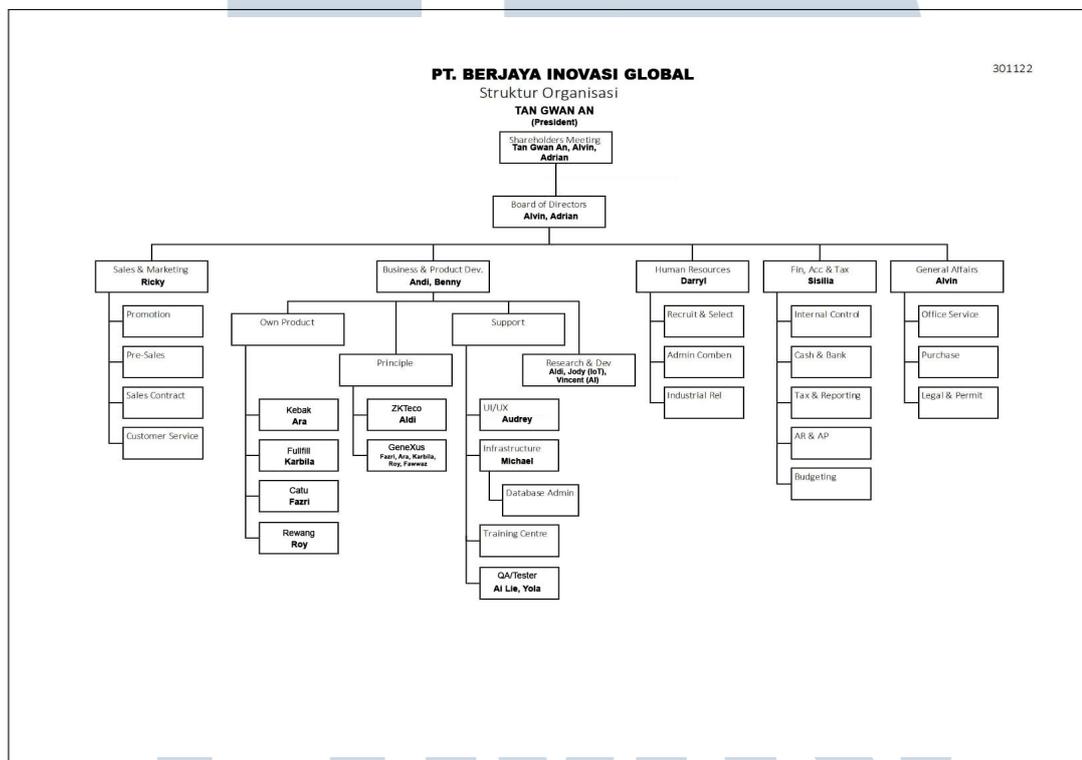


BAB 3 PELAKSANAAN KERJA MAGANG

3.1 Kedudukan dan Organisasi

Selama melakukan magang, kedudukan dan koordinasi sebagai pekerja magang dapat dilihat dalam struktur organisasi dalam gambar 3.1



Gambar 3.1. Struktur Organisasi PT. Berjaya Inovasi Global

Merujuk pada gambar 3.1 yang memuat struktur organisasi PT Berjaya Inovasi Global kerja magang dilaksanakan sebagai Artificial Intelligence Engineer didalam divisi Business & Product Development dan tim Research & Development. Di bawah supervisi dari Andi Brewster selaku kepala dari divisi Business & Product Development selama periode kerja magang.

Selama periode kerja magang, supervisor bisa melihat presensi, hasil kerja, dan kendala yang dihadapi dalam pengerjaan proyek-proyek melalui trello. Trello bekerja sebagai logbook yang mendokumentasikan presensi harian magang, screenshot hasil kerja yang dilakukan, checklist hal-hal yang perlu dikerjakan, dan error yang ditemukan. Untuk proyek-proyek yang melibatkan penggunaan

platform Genexus, fitur *Team Development* bekerja seperti Github dimana anggota-anggota dari tim mulai mengerjakan dari versi proyek yang sama dan setiap anggota bisa melakukan commit untuk mengupdate versi proyek yang bisa diakses semua anggota serta melakukan pull untuk mengupdate versi yang ada didalam *workspace* lokal.

Selain melaporkan hasil kerja, progress dan kendala proyek melalui trello setiap hari setelah jam kerja akan dilakukam laporan secara langsung ke supervisor untuk menerima saran atau solusi yang bisa digunakan untuk proyek, dan menerima arahan atau tugas baru untuk dikerjakan. Jenis meeting lain yang dilakukan selama durasi kerja magang adalah meeting dengan partner-partner dari perusahaan induk PT. Perkasa Pilar Utama seperti Genexus, SingleStore dan MarkLogic.

3.2 Tugas yang Dilakukan

Dalam proses pelaksanaan kerja magang terdapat tugas-tugas yang dikerjakan setiap harinya untuk berbagai proyek. Tugas-tugas tersebut melibatkan eksplorasi fitur-fitur dari tools, aplikasi, atau platform yang akan digunakan dalam development, mengikuti video tutorial yang dicari melalui Youtube atau yang diberikan secara langsung oleh supervisor, dan membuat program atau web application sesuai dengan kebutuhan proyek. Pekerjaan-pekerjaan yang dilakukan bisa dilihat pada tabel 3.1.



Tabel 3.1. Pekerjaan yang dilakukan tiap minggu selama pelaksanaan kerja magang

Minggu Ke -	Pekerjaan yang dilakukan
1	Explore fitur-fitur RasaGPT dan Flowise dan membuat AI Chatbot
2	Mempelajari LangChain dan LLM
3	Training Core Genexus
4	Melanjutkan Training Core Genexus
5	Merancang dan membuat webpanels untuk Knowledge Management System dan AI Chatbot yang bisa digunakan untuk mengupload dokumen-dokumen, informasi KTP dan BPKB, yang kemudian bisa ditanya dan dijawab oleh AI Chatbot.
6	Membuat Knowledge Base projek baru, membuat Transaction, Structured Data Type, dan Webpanels untuk Knowledge Management System, mengintegrasikan API OCR KTP ke Genexus. Kemudian melakukan migrasi database dari MySQL ke SingleStore
7	Mencari solusi untuk mengatasi token limit OpenAI untuk pdf-pdf yang lebih kompleks dan memiliki file size yang lebih besar. Melakukan testing OCR BPKB menggunakan Postman untuk mendapatkan setiap jenis respon data BPKB, kemudian membuat video demo untuk upload dan tanya PDF.
8	Membuat sidepanel Setting agar user bisa menentukan OpenAI key yang ingin mereka gunakan, memodifikasi respon dari Flowise agar berupa plain text dan bukan json. Melanjutkan kode event untuk Upload BPKB dan membuat Structured Data Type untuk respon OCR BPKB, dan membuat video demo untuk fitur
9	Mencari solusi pada masalah dimana sebuah BPKB yang lengkap memiliki 3 lembaran, namun saat OCR digunakan untuk mengekstrak tulisan BPKB maka informasi lembaran-lembaran yang sebelumnya sudah di-scan akan tertimpa. Kemudian merekam video demo upload dan tanya dokumen BPKB.
10	Mempelajari Python LangChain dan membuat AI Agent yang bisa digunakan untuk mengeksekusi tugas yang sudah dikonfigurasi. LangChain dipelajari untuk melihat apakah bisa digunakan sebagai alternatif dari Flowise yang lebih bisa dicustomize.

Tabel 3.1. Pekerjaan yang dilakukan tiap minggu selama pelaksanaan kerja magang (lanjutan)

Minggu Ke -	Pekerjaan yang dilakukan
11	Membuat PDF, Text dan DirectoryLoader yang bisa digunakan agar user bisa memasukkan file kedalam sebuah directory dan user akan bisa bertanya ke AI tentang isi dokumen-dokumen menggunakan LangChain dan melakukan local deployment menggunakan LangServe. Membuat video untuk perbandingan performance Flowise dan LangChain.
12	Mengerjakan project Parking and Traffic Counter, serta Face Detection.
13	Menyelesaikan program Face Detection yang hanya akan mendeteksi wajah seseorang jika wajah tersebut memiliki rasio ukuran 60 – 70% dari keseluruhan foto. Kemudian mencoba mencari metode untuk menghubungkan program face detection agar bisa menerima input foto dari external. Berikutnya, menyelesaikan program Parking Counter yang akan mendeteksi apakah kendaraan berada dalam tempat parkir dan menghitung jumlah ruang parkir yang tersedia.
14	testing response MarkLogic melalui Postman dan mulai membuat webpanels didalam Genexus untuk menghubungkan Genexus dengan MarkLogic melalui http request get dan put, serta mencari solusi untuk digest Authorization yang gagal menghubungkan Genexus dengan MarkLogic.
15	Mengintegrasikan MarkLogic dengan Genexus sebagai alternatif dari SingleStoreDB. Alasan untuk penggantian database ini adalah karena MarkLogic sebagai NoSQL database lebih sesuai dengan kebutuhan KMS dan sudah memiliki AI chatbot sendiri, yang berarti Genexus tidak perlu memanfaatkan Flowise lagi untuk mendapat fitur chatbot.
16	Membuat webpanels untuk melakukan CRUD pada MarkLogic melalui Genexus. CRUD dan search dilakukan untuk data ktp dan PDF.

3.3 Uraian Pelaksanaan Magang

3.3.1 AI Based Knowledge Management System

Langkah pertama dalam proses pembuatan *knowledge management system* berbasis *Artificial Intelligence* adalah mempelajari dan memahami konsep *knowledge management system* berbasis *Artificial Intelligence* beserta contoh implementasinya. Pembelajaran ini dibantu oleh supervisor Andi Brewster yang menjelaskan konsep proyek tersebut sebagai sebuah *database* yang bisa diajak berbicara, para karyawan akan memiliki akses kepada satu *Knowledge Management System* yang berisi informasi-informasi yang dimiliki oleh perusahaan seperti dokumen-dokumen digital, *database* transaksi, karyawan dan informasi lain. Basis dari *knowledge management system* itu adalah Genexus untuk *User Interface (UI)*, SingleStore untuk *database* dan *vector store*, dan Flowise sebagai tools untuk mengintegrasikan LLM dah.

Genexus adalah sebuah platform *software development low-code* yang memanfaatkan *Artificial Intelligence* untuk melakukan automasi dalam membuat aplikasi *multi-platform*. [9] Ada beberapa alasan mengapa Genexus digunakan untuk membuat UI dibandingkan dengan platform *software development* lain ataupun bahasa pemrograman lain. Pertama, karena penggunaan Genexus dalam *software development* merupakan standar prosedur dalam PT Berjaya Inovasi Global, [8] dan alasan lainnya dari menggunakan Genexus adalah automasi yang dilakukan untuk membangun UI. Automasi pembangunan UI membatasi *customization* dalam mendesain UI, namun kekurangan tersebut diterima karena UI yang dibuat memprioritaskan fungsionalitas dan hanya berguna untuk menghubungkan user dengan database dan AI. Untuk menggunakan Genexus, terdapat beberapa hal yang diperlukan yaitu vpn, java tomcat, dan Genexus License yang perlu dibeli terlebih dahulu. Harga license ini bervariasi tergantung pada pricing plan dan fitur yang diinginkan, namun untuk sebuah perusahaan harga nya dimulai dari kisaran \$99 hingga \$329.

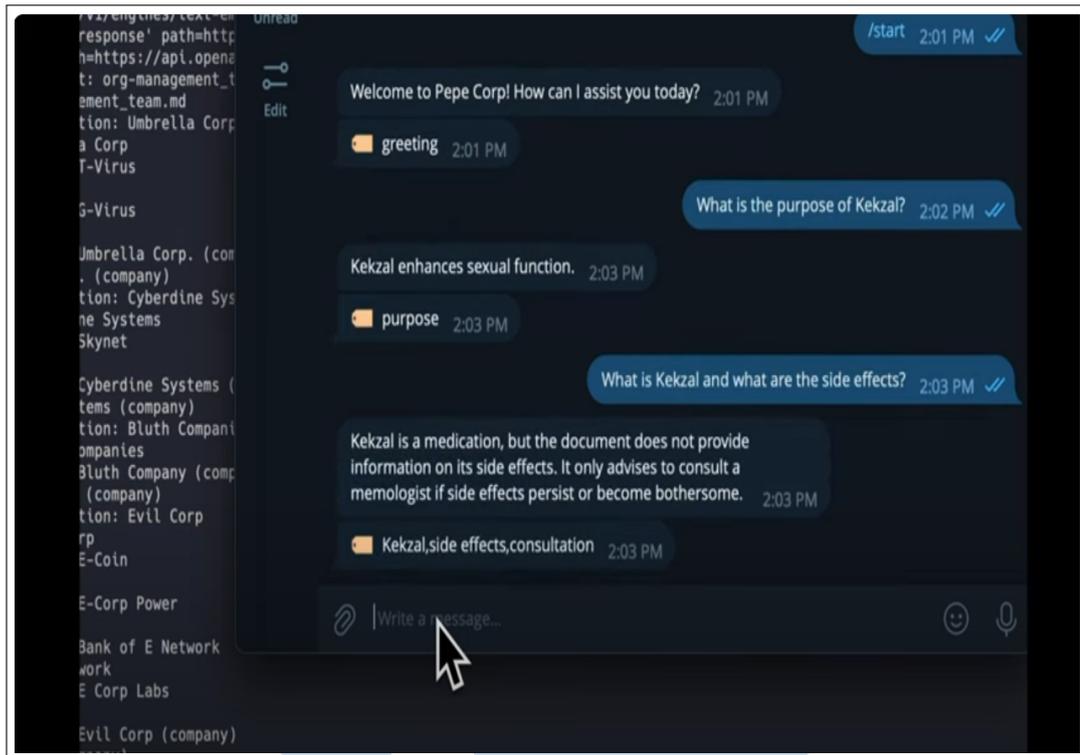
SingleStore atau yang dulu dikenal sebagai memsql adalah platform database yang digunakan untuk menganalisis data, menyimpan data transaksi dan kontekstualisasi data melalui *vector searching*. [10] SingleStore digunakan karena *processing power* yang kuat dan *low-latency* dalam *data ingestion* mendukung dalam melakukan *real-time analytics*. Single, kontekstualisasi data melalui *vector searching* juga membantu AI dalam mencari jawaban berdasarkan konteks

pertanyaan user. Namun performance yang tinggi dan *low-latency* SingleStore memiliki kelemahan yaitu tingkat penggunaan memory yang besar. SingleStore pun juga bersifat berbayar dengan *currency* berupa credit, dengan harga \$3.60 per credit. Konsumsi credit pun juga bervariasi pada kekuatan komputasi yang digunakan, dan storage yang digunakan per Gigabyte setiap bulannya.

Peran AI didalam knowledge management system ini adalah sebagai *chatbot* dan *virtual assistant* yang membantu dalam proses retrieval informasi, yang akan mempercepat proses pencarian informasi oleh karyawan. Konsolidasi seluruh informasi dan dokumen kedalam satu knowledge management system ini pun juga meningkatkan aksesibilitas dan distribusi informasi ke seluruh perusahaan. Dari konsep projek yang diberikan juga ditentukan beberapa *requirement* yang perlu dipenuhi yaitu user harus bisa mengupload file berupa PDF, KTP dan BPKB kedalam database SingleStore, informasi yang tersimpan kedalam database dapat diakses oleh Chatbot, dan Chatbot harus memberikan jawaban yang akurat sesuai dengan data yang tersimpan.

Salah satu contoh atau bukti konsep yang diberikan oleh supervisor adalah RasaGPT, sebuah LLM chatbot yang mengintegrasikan Rasa, yaitu sebuah *converastional AI software*, bot Telegram sebagai *base model chatbot* dan LangChain sebagai library Python yang digunakan untuk indexing, retrieval dan context injection. User Interface (UI) dari Rasa GPT bisa dilihat pada gambar 3.2

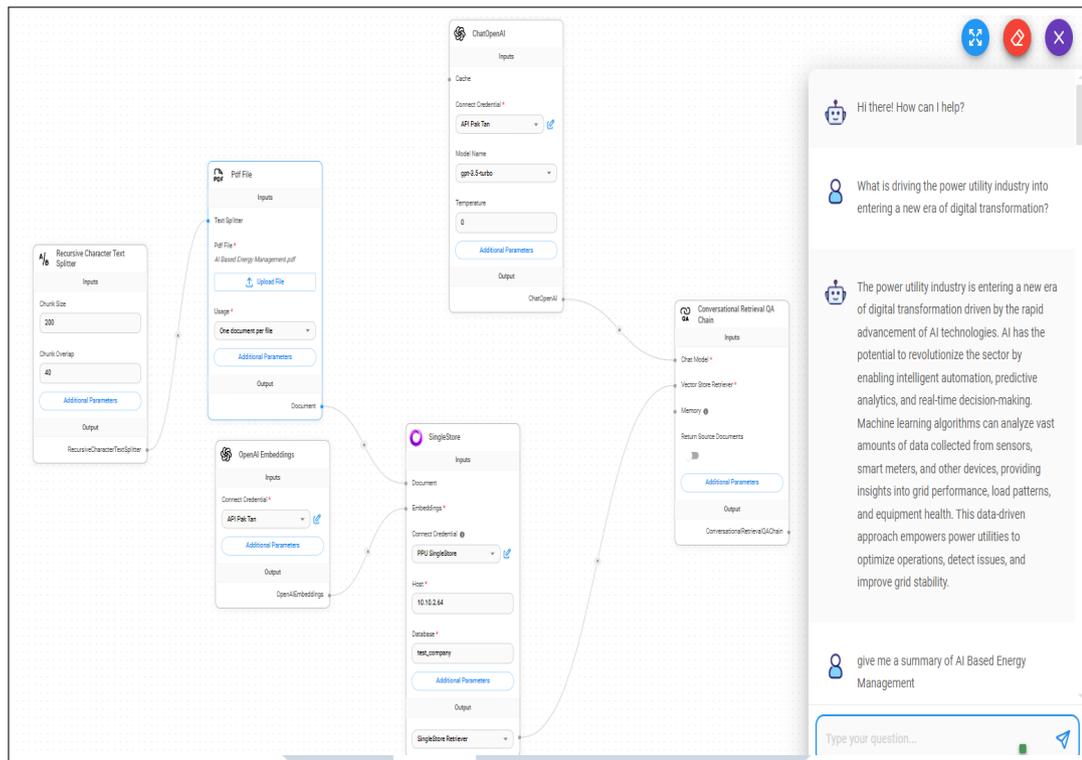




Gambar 3.2. User Interface RasaGPT

Namun RasaGPT tidak bisa digunakan karena *dependencies* command 'make' yang terus mengalami error, sehingga dianjurkan oleh supervisor untuk mencari alternatif selain RasaGPT yang berfungsi sebagai GPT untuk *database*.

Platform alternatif yang ditemukan adalah Flowise, sebuah platform open source yang berfungsi untuk membantu dalam pembuatan alur kerja LLM.[11] Flowise sendiri bersifat *no code*, yang berarti user tidak perlu membuat kode sendiri untuk membangun sebuah *AI chatbot* ataupun *AI Agent*, dan hanya perlu melakukan drag and drop modul yang sudah disediakan. Sederhananya penggunaan Flowise datang dengan sebuah kekurangan yaitu proses *setup* awal cukup rumit dan memerlukan panduan, tapi panduan mendetail bisa ditemukan melalui Youtube atau *official documentation* Flowise. Flowise walaupun bersifat open-source dan gratis, masih memiliki aspek berbayar melalui pemakaian API key LLM yang digunakan, dan biaya yang dikeluarkan melalui API key ini sesuai dengan jumlah token yang dipakai setiap kali LLM dipanggil. Gambar 3.3 menunjukkan UI dari platform Flowise, serta chatflow yang digunakan untuk proses tanya jawab.

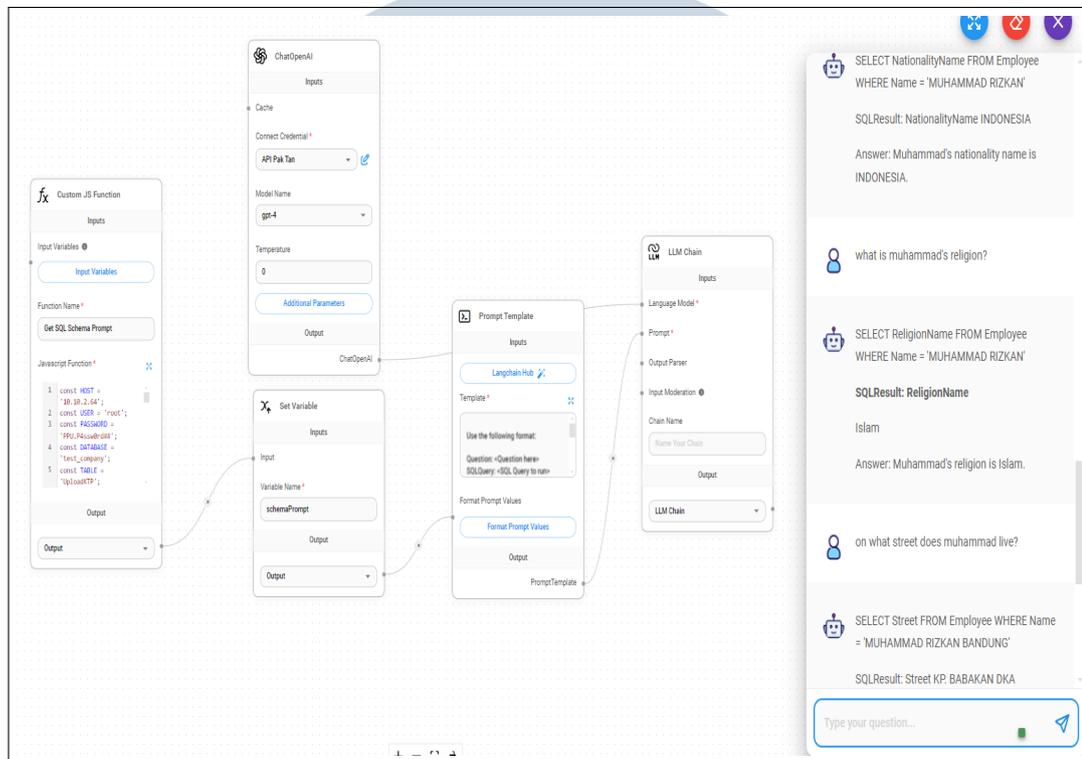


Gambar 3.3. Chatflow untuk tanya jawab pdf

Setelah melakukan instalasi Flowise, dilakukan eksplorasi fitur-fitur Flowise. Salah satu model chatflow yang digunakan adalah *Conversational QA Retrieval Chain*, yang berfungsi untuk menjawab pertanyaan user terkait dengan file yang dikirimkan. Percobaan chain tersebut dilakukan dengan menggunakan file TXT, Word, dan PDF yang sederhana seperti cerita pendek hingga dokumen yang lebih kompleks seperti buku sains dan teknologi untuk mengetes kemampuan AI. Saat kemampuan AI sudah dipastikan untuk proses tanya jawab pdf, maka dibuat chatflow berikutnya

Chatflow berikutnya adalah chatflow untuk chatbot yang bisa mengakses database, dan mengeksekusi query kedalam database dan mengembalikan hasil query tersebut kepada user layaknya sebuah manusia. Percobaan chatflow dilakukan secara berkala dimulai dari menggunakan database lokal seperti sqlite, mysql, dan SingleStore yang dihost secara lokal melalui docker hingga SingleStore yang sudah dihost oleh perusahaan. Kompleksitas data yang diakses pun juga ditingkatkan, dari database yang hanya berisi 20 baris data dengan 3-5 kolom hingga 700 baris data yang digenerate menggunakan chatGPT, berupa tabel *Employee* dengan kolom seperti *FirstName*, *LastName*, *Profession*, *Wage*, dan *HomeAddress*. Namun karena data yang digenerate oleh chatGPT memiliki tingkat kemiripan yang besar, AI

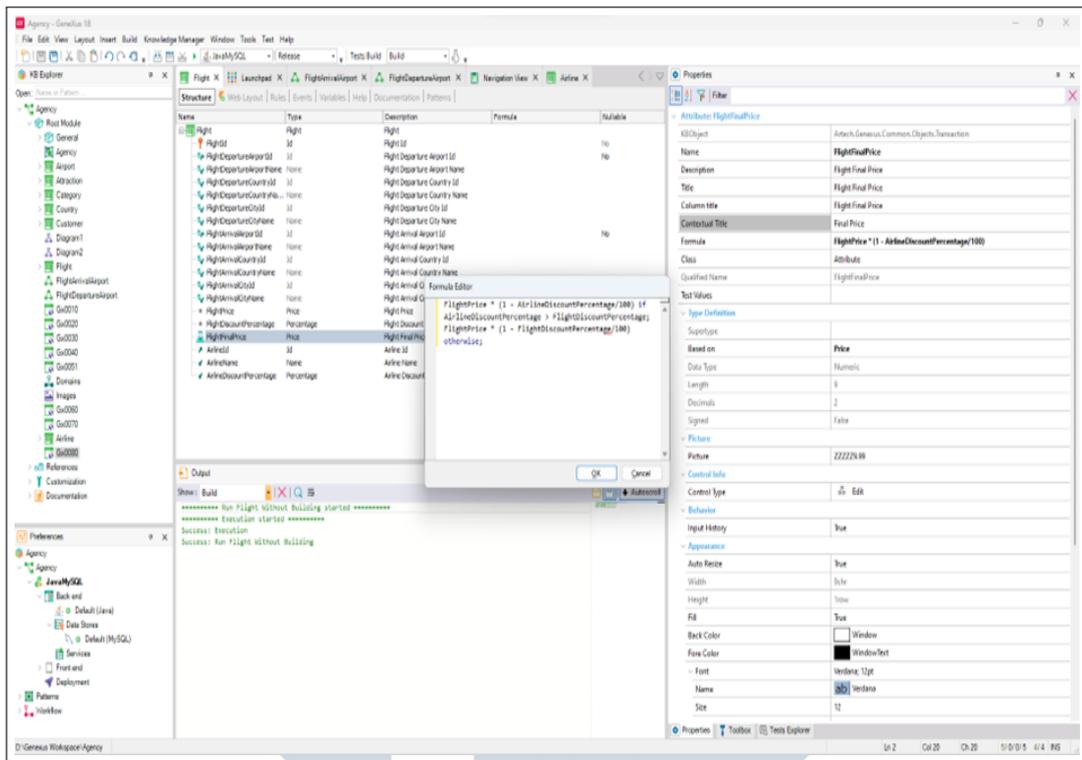
sering mengalami halusinasi dan memberikan jawaban yang tidak sepenuhnya akurat. Chatflow untuk chatbot database bisa dilihat pada gambar 3.4.



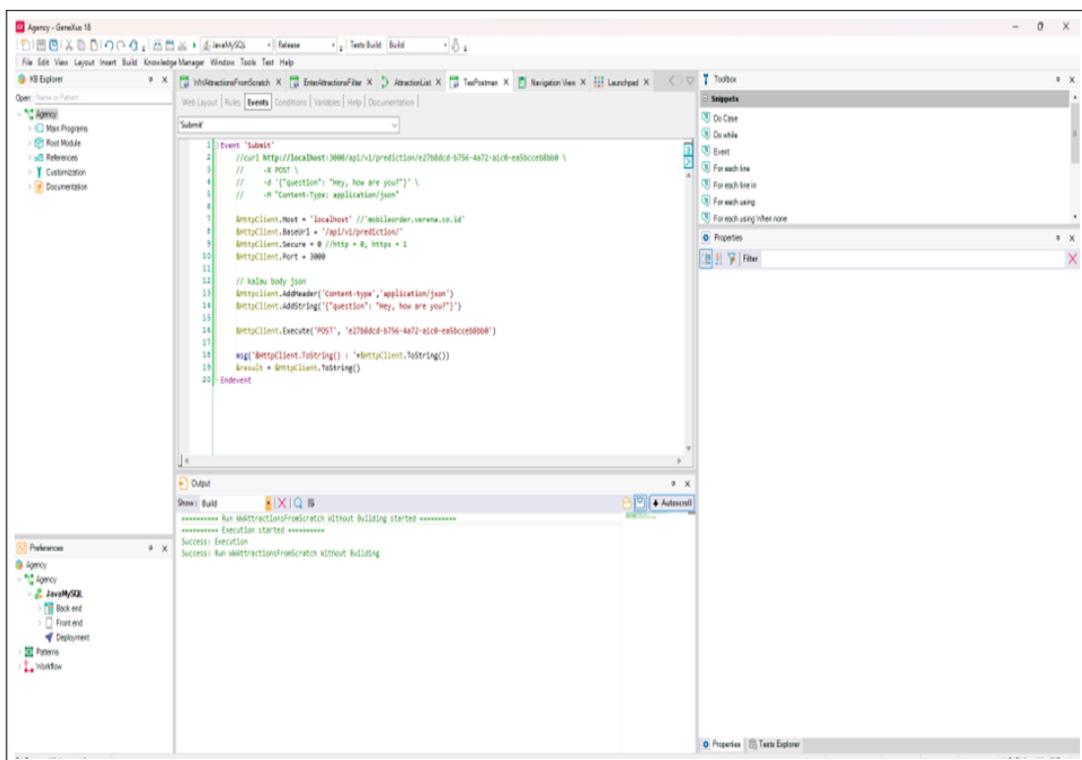
Gambar 3.4. Chatflow database reader

Setelah kedua chatflow tersebut sudah dibuat, dilakukan deployment Flowise menggunakan Railway agar chatflow yang dibuat bisa dimanfaatkan secara online dan tidak hanya terbatas kepada workspace lokal.

Untuk menggunakan Genexus, terdapat beberapa hal yang diperlukan yaitu vpn, java tomcat, dan Genexus License yang perlu dibeli terlebih dahulu. Setelah menerima license developer Genexus, hal pertama yang dilakukan adalah untuk mengikuti video Training Core Genexus. Dalam video training tersebut, akan dibuat sebuah web application yang mensimulasikan sebuah Travel Agency. Tujuan dari training dan pembuatan web application Travel Agency adalah untuk mengenalkan dan memahami komponen-komponen utama yang tersedia dalam Genexus seperti Knowledge Base, Transaction, Webpanels, Structured Data Type, Procedures, WorkWith pattern, Attribute, dan Variabel. Gambar 3.5 dan 3.6 menunjukkan dua komponen utama yang digunakan dalam development proyek.

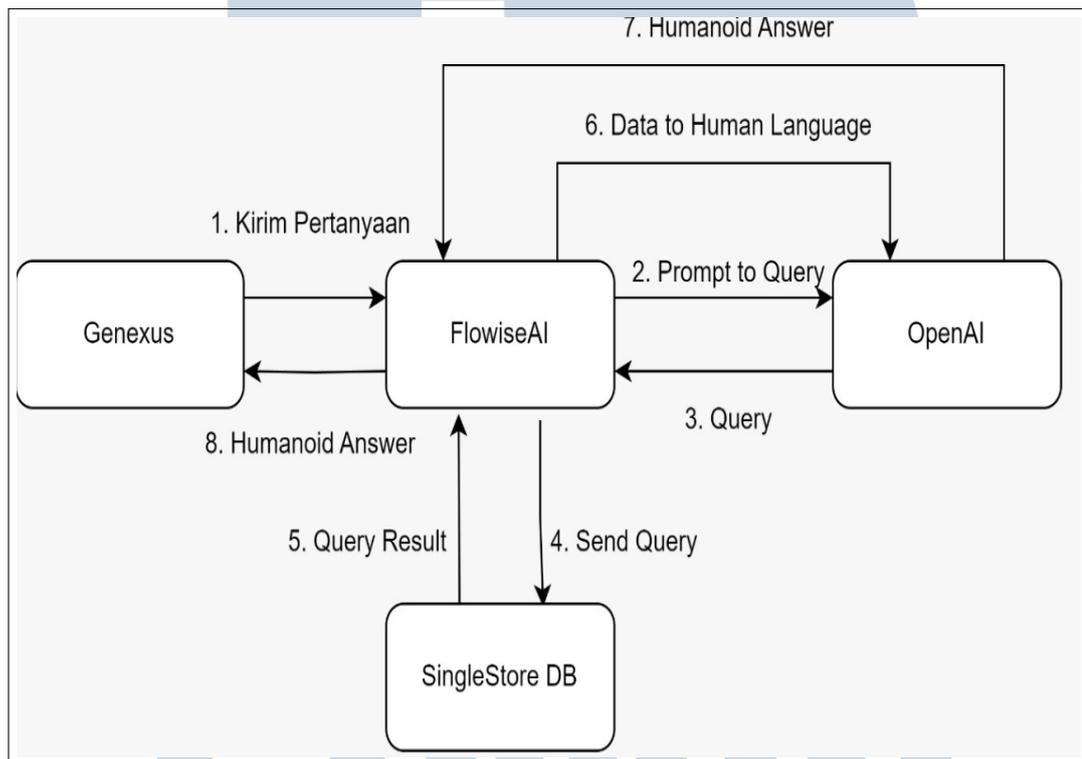


Gambar 3.5. Transaction di Genexus



Gambar 3.6. Webpanels di Genexus

Setelah mengikuti semua video tutorial untuk Training Core Genexus dibuat sebuah Knowledge Base (KB) baru untuk projek knowledge management system berbasis AI. Ada 2 fitur utama yang akan dibuat untuk projek tersebut, yaitu fitur untuk user mengupload file pdf, data KTP, dan BPKB kedalam database, dan fitur yang memperbolehkan user untuk bertanya tentang data-data yang sudah diupload. Konsep dari Knowledge Management System tersebut divisualisasikan dengan gambar 3.7

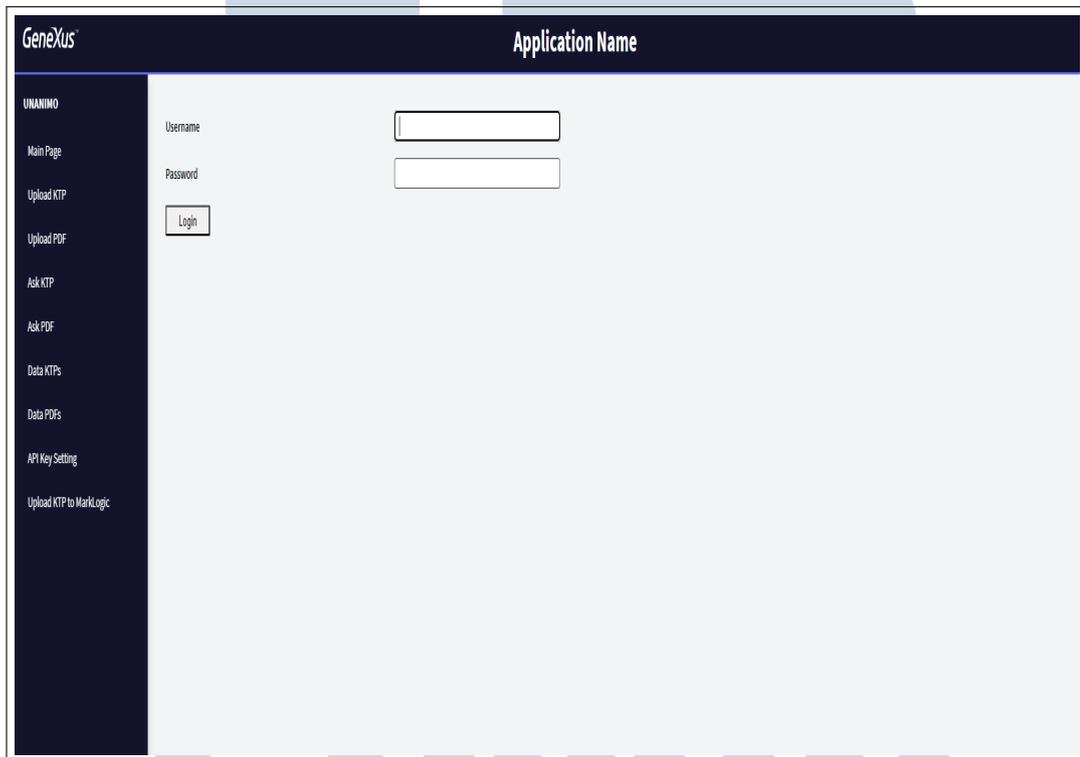


Gambar 3.7. Konsep desain Knowledge Management System

User bisa menuliskan pertanyaan beserta dengan prompt yang ingin digunakan melalui web application Genexus. Genexus akan memproses string kedalam http request POST yang diarahkan kepada Flowise. String pertanyaan dan prompt tersebut akan diproses oleh Flowise dan dikirimkan ke OpenAI yang akan mengubah string tersebut ke bentuk query yang sesuai dan mengembalikannya ke Flowise. Query yang diterima oleh Flowise akan dikirimkan ke SingleStore yang melakukan vector searching berdasarkan query kedalam database. Query yang dihasilkan oleh SingleStore akan diberikan ke Flowise yang akan mengoperkannya ke OpenAI untuk memproses hasil query, hasil query pun akan diubah ke bahasa yang digunakan oleh user untuk mengirim pertanyaan lalu dikembalikan ke Flowise.

Flowise pun akan mengirimkan jawaban yang diterima dari OpenAI kembali ke Genexus yang kemudian akan memperlihatkan jawaban tersebut.

Mengikuti dari konsep yang digambarkan oleh gambar 3.7, hal pertama yang dilakukan adalah menghubungkan KB baru dengan database singlestore yang sudah dihost oleh perusahaan. Berikutnya adalah membuat webpanels-webpanels LoginPage, dan MainPage yang menunjukkan fitur-fitur yang tersedia didalam web application.



Gambar 3.8. Login Page

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3.9. Main Page

Dari Main Page user dapat menavigasi ke fitur yang ingin diakses, yaitu untuk mengupload KTP, PDF, dan BPKB atau untuk bertanya tentang PDF, KTP, dan BPKB yang sudah diupload kedalam SingleStore.

UMN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

Upload your KTP here

ADD FILES START UPLOAD CANCEL UPLOAD

KTP.jpg 322.05 KB

Generate

Province: PROVINSI JAWA BARAT

Regency: KABUPATEN BANDUNG

NIK: 3204271503960004

Name: MUHAMMAD RIZWAN

Birthplace: BANDUNG

Birth Date: 1996-09-14T17:00:00.000Z

Gender: LAKI-LAKI

Blood: [Dropdown]

Street: KP. BABAKAN DINA

RT/RW: 001/015

Kelurahan: CIKUNTA

Kecamatan: CICALENGKA

Religion: ISLAM

Marital Status: KAWIN

Work: KARYAWAN SWASTA

Nationality Name: WNI

Created At: 2021-02-14T17:00:00.000Z

Created In: BANDUNG

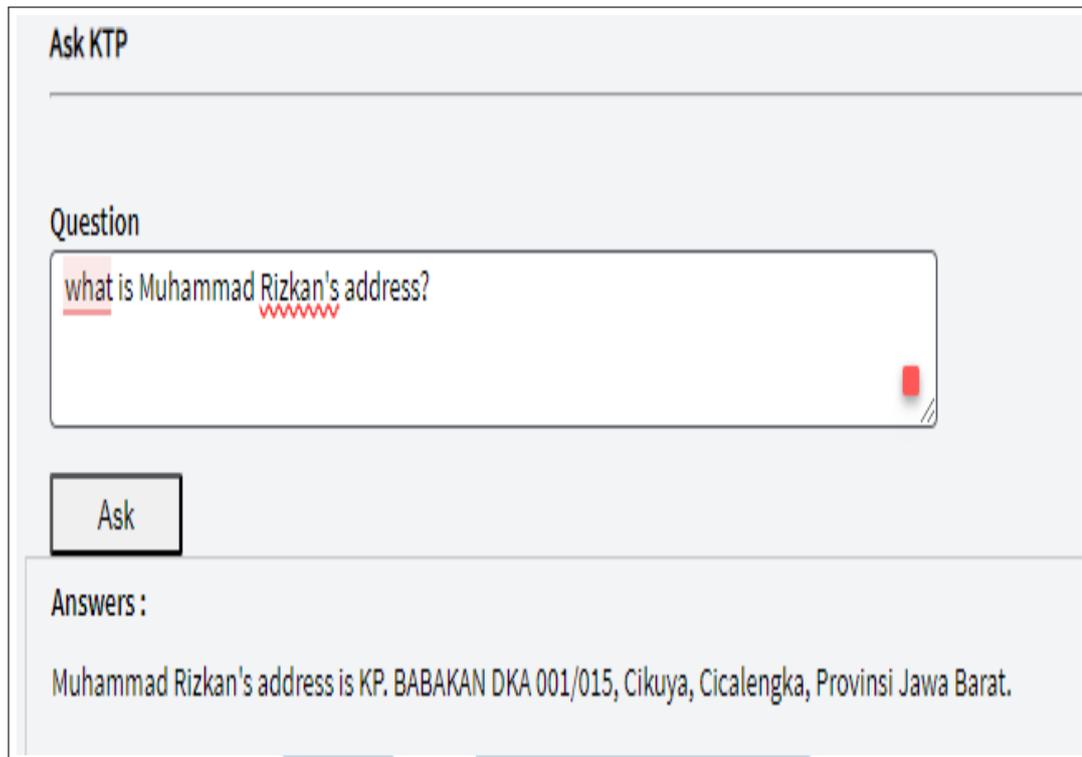
Submit

Ask about KTP

Gambar 3.10. Webpanel KTPUploader

Fungsi Webpanel KTPUploader adalah sebagai tempat user untuk mengupload KTP yang dimiliki kedalam database SingleStore. Webpanel KTPUploader dihubungkan dengan API OCR, dan saat user menekan tombol generate setelah mengupload foto KTP, foto tersebut akan dikirimkan ke API OCR dalam bentuk base64string yang akan mengekstrak teks didalam foto, semua karakter yang terekstrak akan diubah kedalam bentuk json. respon json tersebut akan disimpan kedalam Structured Data Type bernama KTPResponse yang kemudian akan ditampilkan kepada user didalam field-field yang bisa diisi. Jika user memilih untuk submit ktp, maka informasi KTP tersebut akan tersimpan kedalam database SingleStore tanpa foto KTP originalnya. User juga bisa menekan tombol Ask about KTP yang akan memindahkan user ke webpanel AskKTP.

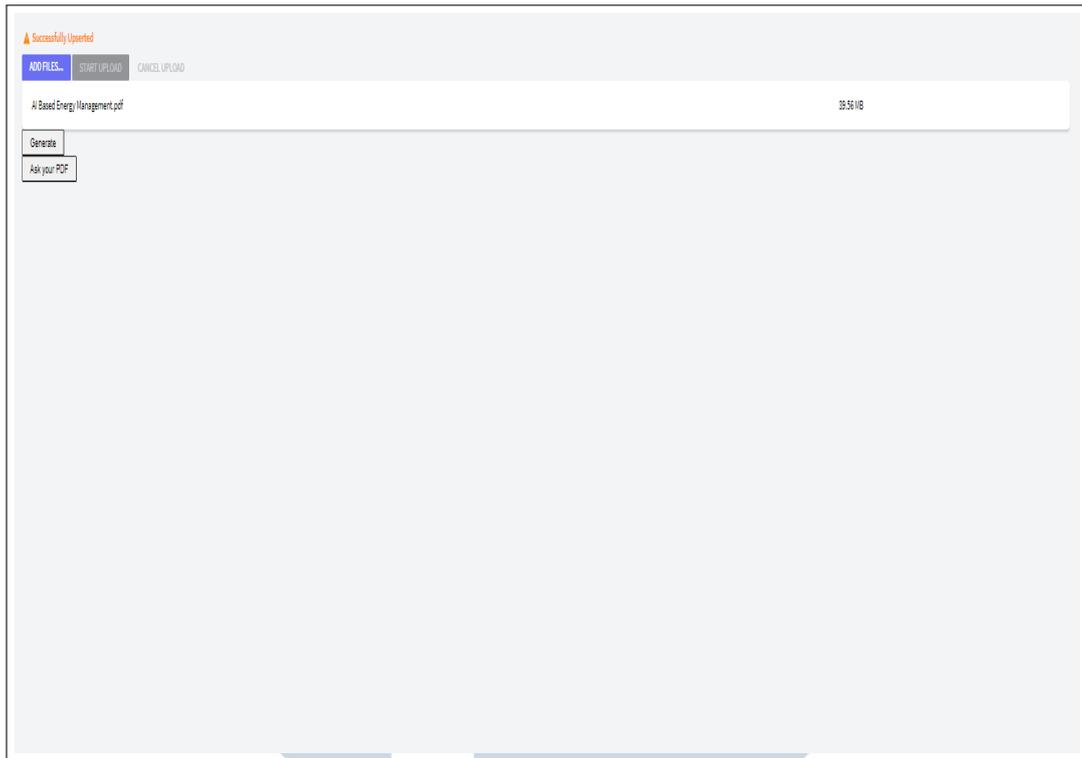
AskKTP berfungsi sebagai interface untuk user bertanya terkait dengan KTP yang sudah diupload. Saat ditekan tombol Ask, maka Genexus akan mengirimkan string pertanyaan yang ada didalam textbox ke aplikasi Flowise melalui http request API yang akan memberikan jawaban sesuai dengan data yang terupload dari KTP.



Gambar 3.11. Webpanel AskKTP

Webpanel Uploader berikutnya adalah PDFUploader. PDFUploader bekerja berbeda dari KTPUploader karena PDFUploader tidak terhubung dengan API OCR, melainkan langsung terhubung ke Flowise untuk proses upsert file kedalam SingleStore. API OCR tidak digunakan dalam webpanel PDFUploader karena jumlah entry yang dimasukkan kedalam SingleStore akan melebihi limit token response dari OpenAI. Jadi PDFUploader langsung dihubungkan dengan chatflow Conversational QA Retrieval Chain PDF dalam Flowise sebab chunking menggunakan modul recursive character text splitter menghasilkan entry database yang lebih sedikit dan jawaban yang lebih akurat. User juga bisa menekan tombol Ask your PDF untuk berpindah ke webpanel AskPDF

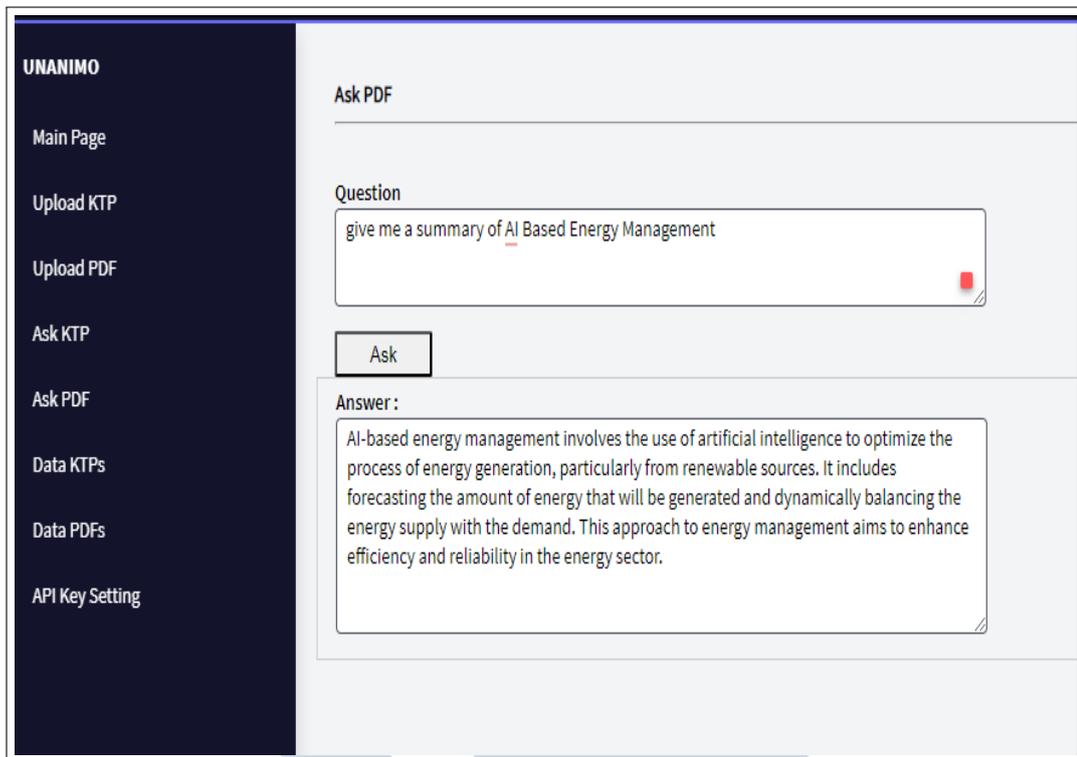
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.12. Webpanel PDF Uploader

AskPDF bekerja layaknya webpanel AskKTP. String pertanyaan dan prompt yang diberikan oleh user akan diproses oleh Flowise setelah user menekan tombol Ask yang akan memanggil API dari aplikasi Flowise. Didalam Flowise, LLM OpenAI akan mengakses vector store database SingleStore yang sudah menyimpan vector dan meta data pdf menggunakan OpenAI embeddings, dan melakukan vector similarity search, setelah dilakukan vector similarity search maka Flowise akan mengembalikan respon string jawaban kembali ke Genexus yang akan menampilkan jawaban tersebut ke user.

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A



Gambar 3.13. Webpanel AskPDF

Webpanel uploader yang terakhir adalah BPKB Uploader, BPKBUploader itu sendiri terdiri dari 3 webpanel yang berbeda, masing-masing untuk lembar BPKB pemilik, lembar BPKB kendaraan, dan lembar BPKB registrasi. Alasan BPKBUploader memiliki 3 webpanel untuk bekerja adalah karena SDT yang digunakan tidak bisa menampung 3 respon json yang berbeda dari API OCR. saat user mengupload lembar BPKB pemilik ke API OCR, data yang terekstrak akan tersimpan kedalam 1 SDT namun saat user mengupload lembar BPKB kendaraan ke API OCR maka data yang sudah tersimpan sebelumnya akan mengalami overwrite dan data tersebut akan hilang. Solusi yang dibuat adalah agar user mengupload lembar pertama yang akan discan oleh OCR, dan kemudian data disimpan kedalam database saat user menekan tombol save. Saat user menekan tombol next user akan berpindah ke webpanel untuk mengupload lembar BPKB kendaraan, dan saat data dari lembar BPKB kendaraan di save data tersebut akan digunakan untuk mengupdate dan melengkapi entry data lembar pemilik yang sudah diupload sebelumnya.

Upload BPKB Identitas Pemilik

ADD FILES... START UPLOAD CANCEL UPLOAD

003.png

Generate

BPKBId: N-20194260

Address: J. KAPTEN ARYA GG.II NO.04 RT 002 KW 003 KEL KARANGANYAR PP WA ISP MW W 1G A. MANTRA MAANKEC INDRAMAYU KAB. INDRAMAYU

Card Id: 3212234310810007

Owner Name: KUNASIH IN DRIYANI

Occupation: MENGURUS KUMAH TANGGA

Save Next

Gambar 3.14. Webpanel upload BPKB pemilik

Saat user menekan tombol next user akan berpindah ke webpanel untuk mengupload lembar BPKB kendaraan, dan saat data dari lembar BPKB kendaraan di save data tersebut akan digunakan untuk mengupdate dan melengkapi entry data lembar pemilik yang sudah diupload sebelumnya. Menekan tombol next akan memindahkan user ke webpanel terakhir untuk mengupload BPKB yaitu BPKB persyaratan registrasi pertama.

U M M N
 UNIVERSITAS
 MULTIMEDIA
 NUSANTARA

▲ BPKB Upload is Successful

Upload BPKB Identitas Kendaraan

ADD FILES... START UPLOAD CANCEL UPLOAD

001.png

Generate

Reg Id	<input type="text" value="L4763 PKA"/>
Plate Color	<input type="text" value="MERAH"/>
Brand	<input type="text" value="YAMAHA"/>
Type	<input type="text" value="LDP NON ABS"/>
Category	<input type="text" value="SPD. MOTOR"/>
Model	<input type="text" value="SOLO"/>
Year	<input type="text" value="287"/>
CC	<input type="text" value="885 CC"/>
Color	<input type="text" value="MERAH"/>
Vin	<input type="text" value="MH3SGSI20HKSBI820"/>
Engine Id	<input type="text" value="GSELEOSSSI21/7"/>
Fuel	<input type="text" value="BENSIN"/>
Axis	<input type="text" value="0"/>
Wheel	<input type="text" value="2 (DUA)"/>

Save

Next

Gambar 3.15. Webpanel upload BPKB kendaraan

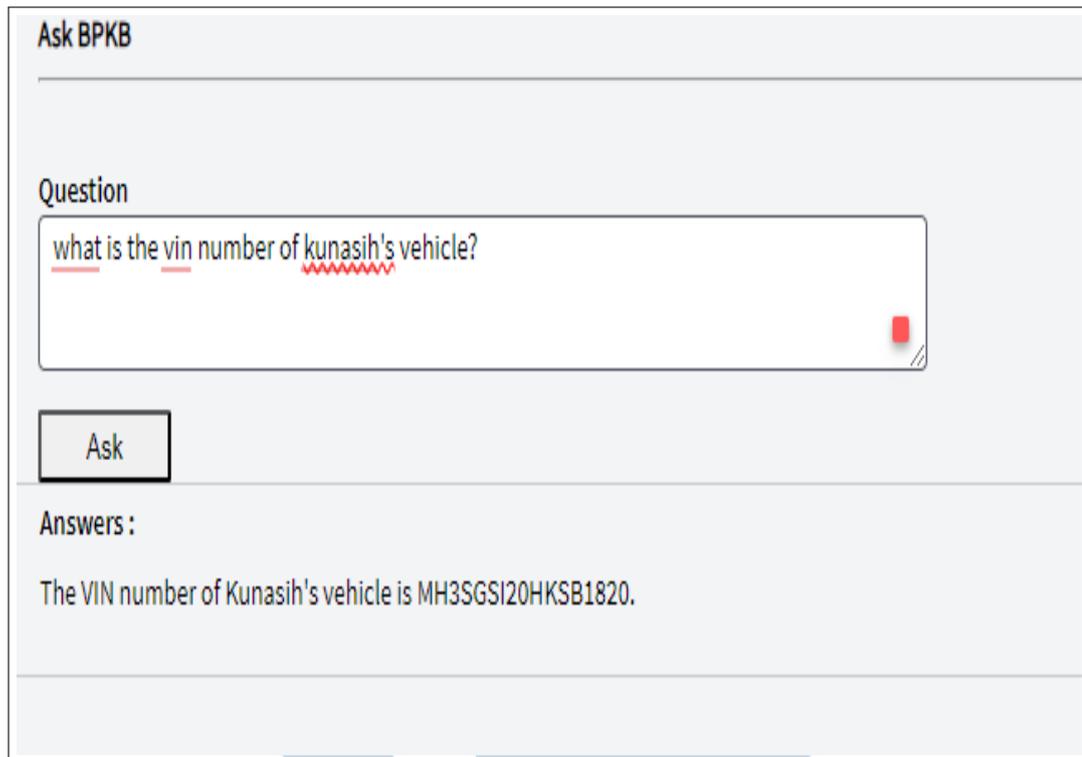
User akan melakukan hal yang sama seperti 2 webpanel sebelumnya, yaitu mengupload BPKB persyaratan registrasi yang akan discan oleh OCR dan mengenerate data yang terekstrak ke dalam field-field yang ditampilkan. Menekan tombol save akan mengupdate entry data terbaru untuk terakhir kali dan user bisa menekan tombol Ask your BPKB yang akan memindahkan user ke webpanel AskBPKB.

UIN
UNIVERSITAS
MULTIMEDIA
NUSANTARA

Gambar 3.16. Webpanel upload BPKB persyaratan registrasi pertama

AskBPKB bekerja seperti webpanel lain yang digunakan untuk bertanya terkait dengan data tersebut, setelah user memasukkan pertanyaan yang ingin dijawab Genexus akan mengirimkan string pertanyaan tersebut ke Flowise yang akan mengakses database SingleStore. Hasil yang ditemukan dari SingleStore akan diberikan kepada LLM OpenAI yang akan mengembalikan jawaban berupa kalimat ke Flowise dan diberikan ke Genexus yang akan menampilkan jawaban tersebut ke user.

UIN
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.17. Webpanel AskBPKB

Setelah proyek ini selesai, dilakukan perekaman video demo yang menunjukkan fitur-fitur dan hasil kerja yang sudah dilakukan, serta menunjukkan kemampuan sistem. Ada juga dilakukan eksplorasi fitur-fitur LangChain untuk membandingkan performa AI melalui Flowise dengan AI melalui LangChain namun perusahaan memutuskan untuk tetap menggunakan Flowise karena kemudahan dalam membuat dan mengubah chatflow sesuai kebutuhan.

Proyek *knowledge management system* juga masih sedang dikembangkan lagi menggunakan database yang berbeda yaitu MarkLogic, namun masih belum ada hasil konkrit yang bisa diperlihatkan.

Ada juga dilakukan tes untuk mengukur akurasi dari AI untuk pertanyaan terkait KTP, PDF dan BPKB. Tes dilakukan sebanyak 10 kali untuk ketiga topik menggunakan dataset KTP, BPKB dan PDF yang disediakan oleh perusahaan, dan menggunakan bahasa Inggris dan bahasa Indonesia. PDF yang digunakan bisa diakses dengan link <https://anyflip.com/lyxku/maxi/basic>.

Tabel 3.3. Tes akurasi AI untuk KTP

Percobaan Ke -	Pertanyaan	Respon	Benar / Salah
1	What is muhammad rizkan's address?	Muhammad Rizkan lives in KP. BABAKAN DKA, 001/015, CIKUYA, CICALENGKA	Benar
2	What is muhammad rizkan's job?	Muhammad rizkan is a private employee	Benar
3	Apa jenis kelamin muhammad rizkan?	Jenis kelamin Muhammad Rizkan adalah laki-laki	Benar
4	What is muhammad rizkan's nationality?	Muhammad Rizkan's nationality is Indonesian.	Benar
5	Apa agama muhammad rizkan?	Agama Muhammad Rizkan adalah Islam.	Benar
6	Kapan tanggal lahir muhammad rizkan	Tanggal lahir Muhammad Rizkan adalah 14 Maret 1996.	Benar
7	What is muhammad's blood type?	Muhammad's blood type is A.	Benar
8	Apa status nikah muhammad?	Status nikah Muhammad adalah belum menikah.	Benar
9	Apa lokasi pembuatan ktp muhammad?	Lokasi pembuatan KTP Muhammad adalah di Bandung.	Benar
10	kapan waktu pembuatan ktp muhammad	Waktu pembuatan KTP Muhammad adalah pada tanggal 14 Maret 1996.	Benara

Berdasarkan hasil dari 10 kali percobaan, akurasi AI untuk topik KTP adalah 100%, AI berhasil menjawab semua pertanyaan yang diberikan dengan benar.

Tabel 3.4. Tes akurasi AI untuk BPKB

Percobaan Ke -	Pertanyaan	Respon	Benar / Salah
1	Who is Kunasih Indriyani?	Kunasih Indriyani is an owner	Benar
2	Berapakah nomor id bpkb kunasih?	BPKBId Kunasih adalah N-20194260	Benar
3	Berapakah nomor registrasi kunasih?	Nomor registrasi Kunasih adalah L4763 PKA.	Benar
4	What is the vin number of kunasih vehicle	MH3SGSI20HKSB182B	Benar
5	Apa pekerjaan kunasih indriyani	Pekerjaan Kunasih adalah mengurus rumah tangga.	Benar
6	What is the vehicle that kunasih owns	Kunasih owns a 2020 YAMAHA IND. MOTOR MFG.	Benar
7	Kendaraan yang dimiliki kunasih memiliki berapa roda?	Kendaraan yang dimiliki Kunasih memiliki 2 roda.	Benar
8	apa brand kendaraan milik kunasih?	Brand kendaraan Kunasih adalah Yamaha.	Benar
9	What is the category of kunasih's vehicle?	The category of Kunasih's vehicle is SPD. MOTOR.	Benar
10	What is the model of kunasih's vehicle?	The model of Kunasih's vehicle is SOLO	Benar

Dari 10 kali percobaan, bisa dilihat bahwa hasilnya adalah AI untuk topik BPKB memiliki akurasi 100% dan berhasil menjawab semua pertanyaan dengan benar.

Tabel 3.5. Tes akurasi AI untuk PDF

No	Pertanyaan / prompt	Respon	Benar / Salah
1	Siapakah kancil dalam cerita ini?	Dalam cerita ini, Kancil adalah seekor tikus rusa yang cerdik dan licik yang berhasil mengalahkan hewan lain seperti Badak dan Banteng melalui kecerdasan dan kecerdikan.	Benar
2	apa peran serigala dalam cerita?	Serigala sering kali menjadi musuh Kancil yang mencoba menangkapnya, namun Kancil selalu berhasil mengelabui serigala dengan kecerdikannya	Benar
3	sebutkan protagonis dalam cerita ini	Protagonis dalam cerita Kancil yang cerdik adalah Kancil sendiri, Badak, Banteng, dan Serigala.	Salah
4	sebutkan antagonis dalam cerita ini	Antagonis dalam cerita Kancil yang cerdik tidak secara eksplisit disebutkan dalam konteks yang diberikan.	Salah
5	apa pesan moral yang bisa dipelajari dari cerita ini?	Cerita ini mengajarkan bahwa dengan kecerdasan dan kecerdikan, seseorang dapat mengatasi rintangan dan mengalahkan lawan yang lebih kuat secara fisik	Benar

U N I V E R S I T A S
M U L T I M E D I A
N U S A N T A R A

Tabel 3.5. Tes akurasi AI untuk PDF (lanjutan)

No	Pertanyaan / prompt	Respon	Benar / Salah
6	how does the story conclude?	The story concludes with Kancil outsmarting the other animals like Badak, Banteng, and Serigala using his intelligence and cunning nature. Kancil emerges victorious and teaches a lesson about wit and cleverness prevailing over physical strength.	Benar
7	how do other characters react to kancil's strategies and actions?	The other characters in the story react positively to Kancil's strategies and actions. When they see Kancil's cleverness and wit in action, they believe that Kancil will be able to outsmart their opponents like Badak and Serigala.	Benar
8	How does Kancil use his cleverness?	In the story, the character Kancil uses his cleverness to overcome obstacles by outsmarting stronger and more powerful opponents like Badak and Serigala.	Benar
9	give me an example of it	One example is when he tricks Serigala into falling into a trap by pretending to be asleep on the path. When Serigala approaches, thinking Kancil is an easy target, Kancil quickly jumps up and escapes, leaving Serigala trapped in the pitfall.	Benar
10	elemen budaya dan tradisi apa saja yang ada dalam cerita ini?	Dalam cerita Kancil yang cerdik, terdapat elemen budaya dan tradisi dalam hewan-hewan yang digambarkan melalui karakteristik yang sering digunakan dalam cerita rakyat atau fabel tradisional.	Benar

Sementara hasil dari 10 kali percobaan, akurasi AI untuk topik PDF adalah 80% karena mengalami kekeliruan antara protagonis dan antagonis. Sementara saat ditanyakan tentang antagonis dalam cerita, AI menjawab bahwa antagonis cerita tidak dinyatakan secara eksplisit. Alasan akurasi topik PDF tidak mencapai 100% adalah dibandingkan dengan KTP dan BPKB yang memiliki jawaban pasti, AI harus merangkum cerita dan memberikan respon berdasarkan konteks cerita yang diberikan.

Kinerja dari Knowledge Management System berbasis Artificial Intelligence

pun juga diukur oleh supervisor, Andi Brewster. Supervisor diberikan 4 pertanyaan dengan pilihan dalam bentuk *google form*, pilihan ini berupa skala dari 1 hingga 5 dan pilihan ganda. Berikut adalah pertanyaan pertanyaan yang ada didalam form.

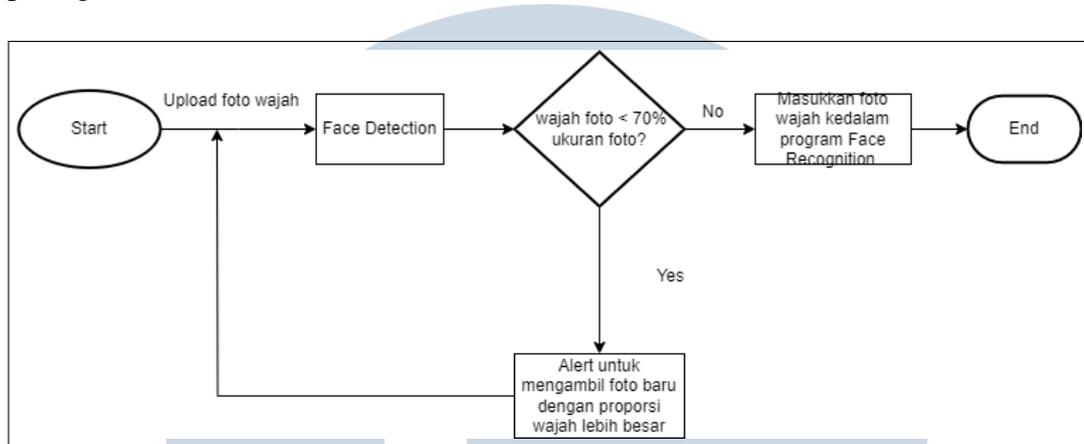
1. Seberapa menarik desain UI dari KMS berbasis AI? 1 (tidak menarik) - 5 (menarik) Jawaban: Supervisor menilai desain UI dengan nilai 3
2. Seberapa mudah user dapat melakukan navigasi antara menu-menu? 1 (sulit) - 5 (mudah) Jawaban: Supervisor memberikan nilai 4 untuk tingkat kemudahan dalam melakukan navigasi antara menu-menu
3. Seberapa singkat atau lama proses upload dan penyimpanan data? Jawaban: Supervisor memberikan *range* waktu dari 5 sampai 15 detik untuk mengupload dan menyimpan data.
4. Seberapa akuratkah AI dalam menjawab pertanyaan user? Jawaban: Supervisor menilai akurasi AI 80% yaitu AI dapat menjawab 4 pertanyaan dengan benar dari 5 pertanyaan yang diberikan.

3.3.2 Face Detection for Attendance

Untuk pembuatan program Face Detection for Attendance dilakukan untuk memperbaiki permasalahan sistem presensi wajah tamu di kantor Sinarmas APP. Requirement yang telah ditentukan harus bisa dilakukan oleh program adalah program bisa mendeteksi wajah dari foto, dan akan memberikan alert jika ukuran wajah yang terdeteksi memiliki proporsi wajah lebih kecil dari 70% ukuran foto.

Beberapa alat development yang digunakan untuk membuat program tersebut adalah Thonny IDE (Integrated Development Environment) dan juga library OpenCV. Setelah dilakukan instalasi untuk kedua hal tersebut dicari tutorial untuk membuat program *face detection* menggunakan OpenCV. Program ini bekerja dengan mengambil gambar yang diberikan dan melakukan *grayscale*, setelah gambar wajah sudah berupa *image grayscale* maka akan digunakan haarscascade classifier untuk mendeteksi wajah yang ada didalam foto. Jika wajah didalam foto berukuran setidaknya 70% dari ukuran foto yang dikirim, maka foto wajah akan disimpan kedalam program Face Recognition Sinarmas APP untuk identifikasi saat kunjungan tamu. Namun, jika wajah didalam foto memiliki ukuran yang lebih kecil dari 70% ukuran foto maka user akan menerima alert untuk mengirimkan foto baru.

Berikut adalah flowchart yang menunjukkan alur program face detection pada gambar 3.18



Gambar 3.18. Flowchart program Face Detection

Setelah program dibuat, program dimodifikasi untuk memunculkan bounding box hanya setelah ukuran wajah yang dideteksi memiliki ukuran setidaknya 70% dari ukuran foto yang dikirimkan, jika foto wajah memenuhi ukuran minimal maka akan dikembalikan respon jumlah wajah yang terdeteksi.

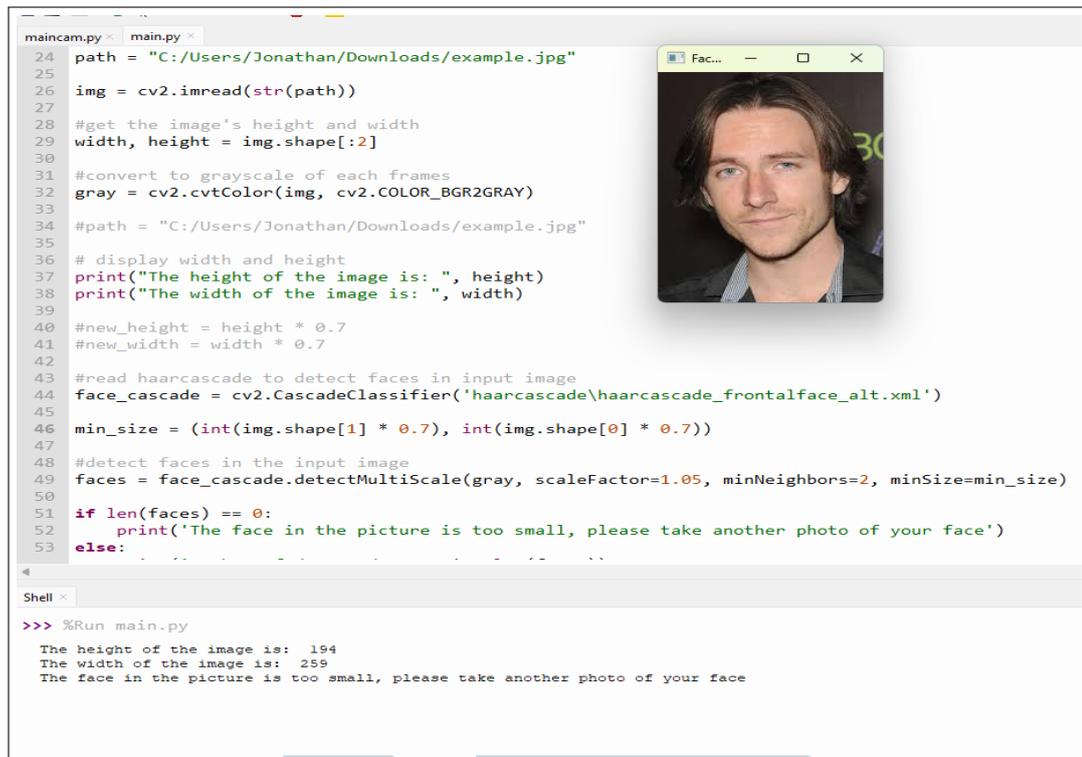
```

maincam.py | main.py
28 #get the image's height and width
29 width, height = img.shape[:2]
30
31 #convert to grayscale of each frames
32 gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
33
34 #path = "C:/Users/Jonathan/Downloads/example.jpg"
35
36 # display width and height
37 print("The height of the image is: ", height)
38 print("The width of the image is: ", width)
39
40 #new_height = height * 0.7
41 #new_width = width * 0.7
42
43 #read haarcascade to detect faces in input image
44 face_cascade = cv2.CascadeClassifier('haarcascade\haarcascade_frontalface_alt.xml')
45
46 min_size = (int(img.shape[1] * 0.7), int(img.shape[0] * 0.7))
47
48 #detect faces in the input image
49 faces = face_cascade.detectMultiScale(gray, scaleFactor=1.05, minNeighbors=2, minSize=min_size)
50
51 if len(faces) == 0:
52     print('The face in the picture is too small, please take another photo of your face')
53 else:
54     print('Number of detected Faces:' , len(faces))
55
56 #loop over all detected faces
57 for (x,y,w,h) in faces:
58     # to draw a rectangle around the detected face
  
```

```

Shell
Please input the file path to the image:C:\Users\Jonathan\Downloads\example2.jpg
The height of the image is: 183
The width of the image is: 275
Number of detected Faces: 1
  
```

Gambar 3.19. Foto wajah sesuai ukuran minimum



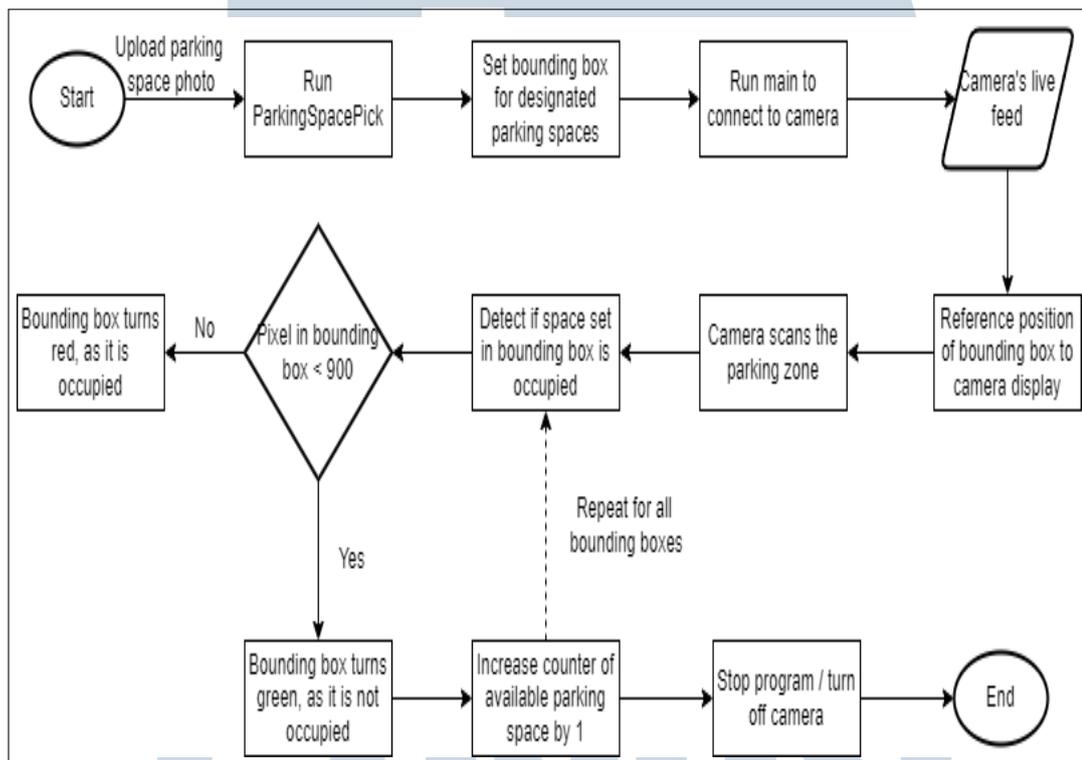
Gambar 3.20. Foto wajah lebih kecil dari ukuran minimal

Tapi jika wajah yang terdeteksi terlalu kecil dari ukuran minimal yang sudah ditentukan, maka akan muncul pesan yang meminta user untuk mengambil foto yang baru karena wajah yang terdeteksi terlalu kecil. Pada gambar kedua, *requirement* ukuran minimal wajah tidak terpenuhi, dan bisa dilihat tidak ada bounding box yang muncul disekitar wajah dan juga pesan yang meminta user untuk mengambil foto baru.

3.3.3 Parking Space Counter

Program *parking space counter* dibuat sebagai alternatif dari menggunakan sensor untuk menentukan berapa banyak kendaraan yang ada di tempat parkir serta jumlah tempat parkir yang tersedia. Penggunaan alternatif ini dilakukan karena harga sensor yang mahal untuk ditempatkan pada setiap tempat yang bisa di isi oleh kendaraan. Maka dibuat program *parking space counter* yang akan dijalankan pada kamera yang dipasangkan pada area tempat parkir, dan satu kamera tergantung pada penempatannya bisa mencakup area yang lebih luas dibandingkan satu sensor. *Requirement* dari program ini adalah user harus bisa menentukan area-area yang akan diklasifikasikan sebagai tempat parkir, program harus bisa mendeteksi apakah

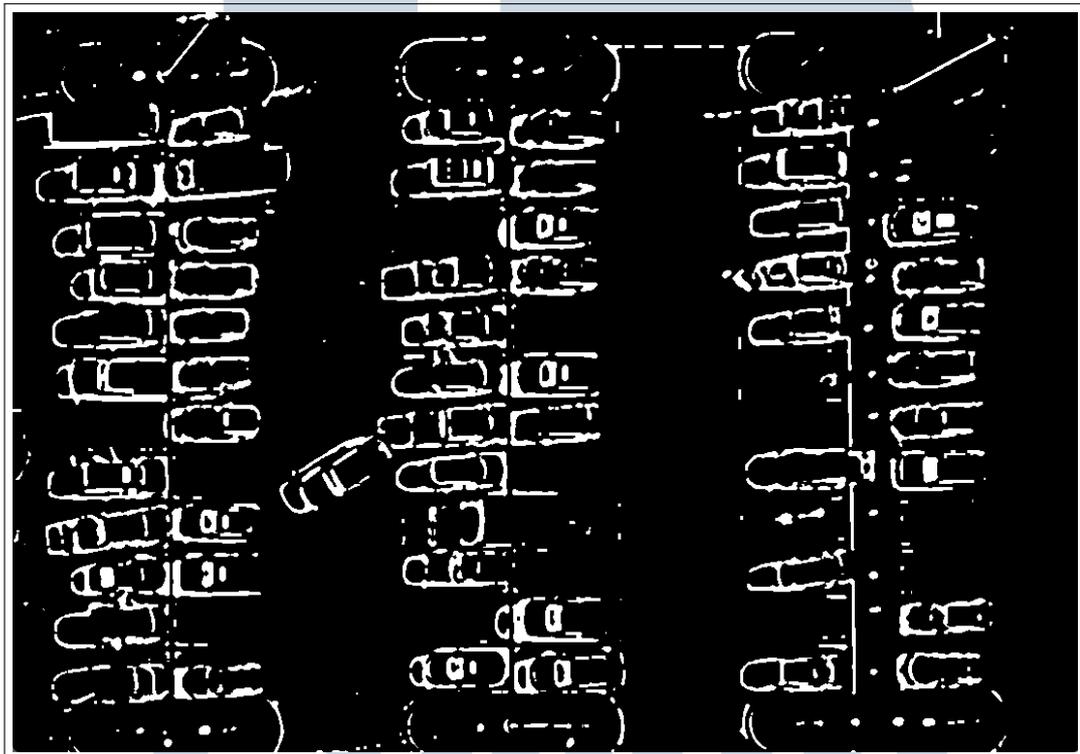
ada kendaraan yang menempati area parkir, dan program harus bisa menghitung jumlah tempat parkir yang ada secara keseluruhan dan jumlah tempat parkir yang masih tersedia. Dalam pembuatan program ini, tes dilakukan menggunakan foto dan video tempat parkir yang diberikan oleh perusahaan, namun foto dan video juga bisa didownload melalui link <https://www.computervision.zone/courses/parking-space-counter/> Gambar 3.21 menunjukkan flowchart program parking space counter.



Gambar 3.21. Flowchart program Parking Space Counter

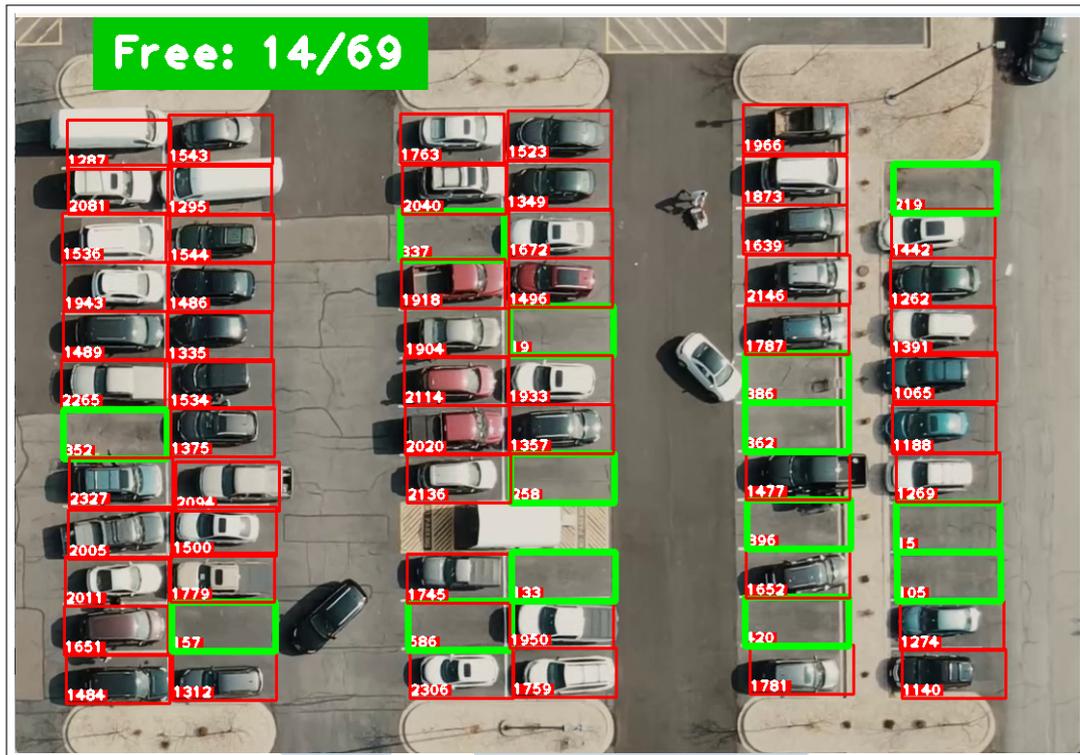
Untuk proyek *parking space counter*, dibuat 2 program yang berbeda. Program pertama adalah program *ParkingSpacePick* yang saat dijalankan akan mengakses foto tempat parkir yang akan dicakup dalam penglihatan kamera, foto yang ditampilkan kepada user bisa di klik untuk memunculkan *bounding box* yaitu kotak yang menandakan luas area deteksi untuk kotak tersebut. Jika ada kendaraan yang masuk kedalam area kotak tersebut, maka kotak akan menunjukkan bahwa area tersebut sudah ditempati oleh kendaraan. Program kedua bernama *main* menerima *live feed* dari kamera dan terhubung dengan program pertama, sehingga saat kamera mulai memantau area parkir, kamera akan bisa mengidentifikasi tempat-tempat mana saja yang bisa digunakan oleh kendaraan.

Setelah program main dijalankan *Live feed* kamera akan digrayscale untuk meningkatkan efisiensi pemrosesan gambar, dan dengan menggunakan *Gaussian Blur* dan *Adaptive Threshold*, video yang terekam oleh kamera akan berubah menjadi hitam putih. Warna putih tersebut adalah pixel objek yang terdeteksi oleh kamera, dan jumlah pixel tersebut akan digunakan untuk menentukan apakah terdapat sebuah mobil di area tersebut. Bisa dilihat pada gambar 3.21 yang menunjukkan lokasi-lokasi mobil dengan pixel berwarna putih.



Gambar 3.22. Image Threshold yang menunjukkan lokasi mobil

Setelah itu, program dimodifikasi untuk mendeteksi jumlah pixel yang ada di dalam *bounding box*. jika pixel yang terdeteksi di dalam *bounding box* berjumlah diatas dari 900, *bounding box* akan berubah menjadi warna merah untuk menandakan bahwa tempat parkir tersebut sudah dipakai. Namun jika pixel yang dideteksi lebih rendah dari 900 maka *bounding box* akan berubah menjadi warna hijau untuk menandakan tempat yang kosong. Berikutnya ditambahkan juga hitungan pixel pada *bounding box* dan juga *counter* yang akan menunjukkan total tempat parkir yang ada dan jumlah tempat parkir yang masih kosong.



Gambar 3.23. Program dijalankan pada video tempat parkir

3.4 Kendala dan Solusi yang Ditemukan

Selama proses kerja magang, ditemukan beberapa kendala yang menghambat pengerjaan projek. Kendala tersebut muncul dalam bidang teknis .

Beberapa kendala teknis yang dihadapi:

1. Bahasa pemrograman Genexus memiliki persamaan dengan bahasa pemrograman lainnya, namun karena syntax yang berbeda diperlukan penyesuaian terlebih dahulu.
2. AI yang digunakan cenderung menglaami halusinasi terutama saat jumlah data semakin besar atau karena dataset yang digunakan memiliki tuple data yang terlalu mirip dengan satu sama lain.
3. Requirement OpenAI API Key yang berbayar untuk menggunakan LLM OpenAI.
4. Reorganization database dalam Genexus yang sering menghasilkan fatal error.

5. Deployment Flowise menggunakan Render yang berbayar

Solusi-solusi yang ditemukan untuk menghadapi kendala teknis tersebut yaitu:

1. Mengikuti video tutorial Genexus, serta membaca documentation Genexus setiap menemukan error yang tidak dipahami.
2. Melakukan prompt engineering untuk embeddings dan LLM, serta mencari dataset yang memiliki data yang lebih unik dibandingkan satu sama lain.
3. Melakukan request OpenAI API Key kepada supervisor dan menerima approval oleh direktur
4. Memperhatikan foreign key dan struktur tabel yang berubah dan melakukan commit kedalam Team Development Genexus (seperti Github) setiap kali akan melakukan reorganization.
5. Mencari alternatif deployment menggunakan Railway yang menyediakan porsi kredit penggunaan untuk user gratis.

