UM Hackathon 2025

Submission Date: 12/4/2025

Domain 3

Task 1

DAX Assistant – Handsfree

Group Vegetables

Team Member
LAM YOKE YU
GOE JIE YING
TAN YI YA
TEH RU QIAN

1.0 Introduction

Currently, Grab drivers, whether delivering food or providing ride-hailing services, interact with AI through text to ask questions. This is not only inefficient but also poses a serious safety risk. To address this, Grab is developing a voice-based AI assistant that allows drivers to get help without needing to pick up their phones and type. This system will greatly enhance convenience, enabling drivers to receive quick answers while working more efficiently and safely, making it a true win-win solution.

2.0 Problem Statement

During the development process of the voice-centric interface, several important factors need to be considered to ensure that drivers have a good user experience:

- **Audio Condition on the Road:** Background noises such as vehicle engine sounds and weather conditions can interfere with voice recognition. The AI must accurately interpret user instructions despite these challenges while also delivering clear and efficient feedback.
- **Speech Pattern Complexities:** Malaysia's diverse driver community speaks multiple languages and dialects. Beyond the three main languages (Malay, English, and Chinese), the AI should also adapt to various dialects and mixed-language speech patterns (e.g., *bahasa rojak*) to ensure inclusivity and accessibility.

3.0 Solution

The solution to overcome the problem above is to build a robust voice interaction system that enables reliable driver–assistant communication in challenging audio environments.

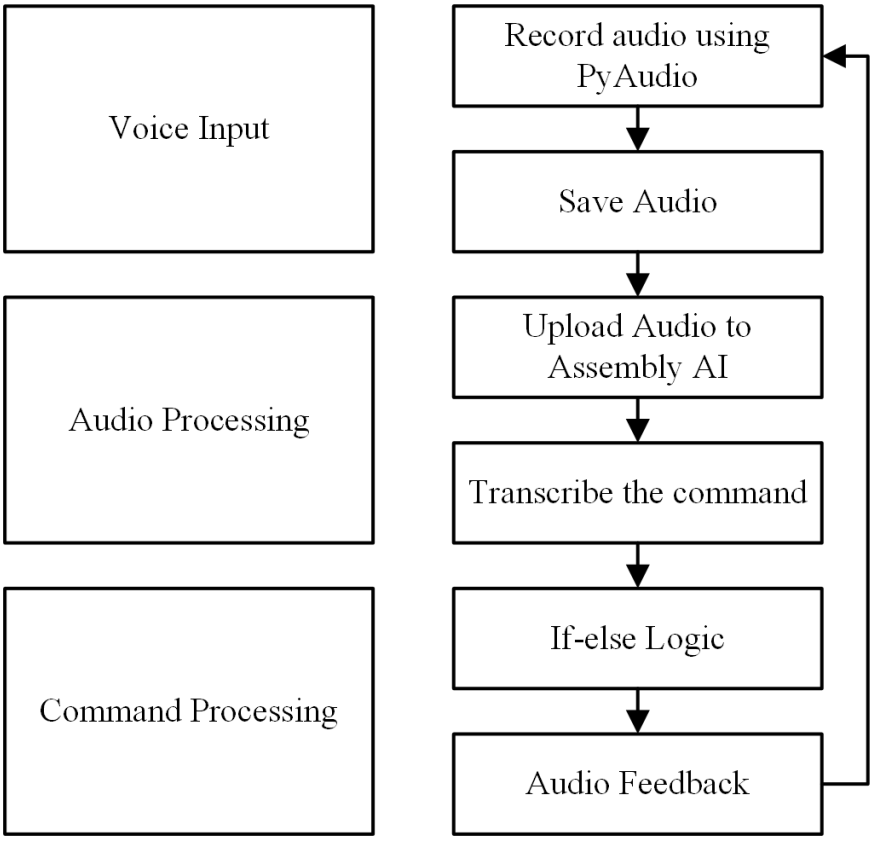
3.1 Overview

The AI-powered voice assistance:

- Supports **different languages** commonly used by Malaysian drivers, such as Regional Malay, Malaysian English, Malaysian Chinese, *bahasa rojak*, and more.
- Can be automatically initiated by a prompt, eliminating complex wake-up steps and enabling zero-touch activation.
- Responds to driver queries via voice prompts.
- Notifies drivers when a new order arrives and allows them to accept or decline using voice commands.
- Provides real-time updates on order status.

After implementing these functions, the assistant will enhance accessibility by supporting multiple languages, optimize driver decision-making, and minimize distractions.

3.2 Architecture



Architecture Components of the Prototype

Voice Input

- Components: `record_mic()`
- Captures audio from the microphone using the `pyaudio` library.
- The program listens for a predefined time of 5 seconds.
- The captured audio is saved as a `.wav` file with a timestamp as the filename.

Audio Processing

- Components: `upload()`, `transcribe()`, `get_transcription_result_url()`, `save_transcript()`
- The recorded file is uploaded to AssemblyAI API using a POST request for transcription.
- The `transcribe()` function is called to start the transcription process, and the script polls the API until the transcription is completed.
- Once the transcription is completed, it is saved to a text file and returned.

Command Processing and Feedback

- Components: `talk()`

- The transcribed text from the recorded audio is used as a command to control the assistant. Based on the transcribed command, the program performs specific actions.
- The program provides audio feedback to the user using text-to-speech (TTS) through the `pyttsx3` library.

Key Libraries Used

- `PyAudio`: Captures live audio from the microphone.
- `wave`: Saves the recorded audio to `.wav` format.
- `requests`: Handles uploading of the audio file and makes API requests to AssemblyAI for transcription.
- `pyttsx3`: Provides text-to-speech functionality for the assistant to speak back to the user.

Other Libraries

- `pywhatkit`: Handles playing songs on YouTube by searching for the song name.
- `wikipedia`: Retrieves summary information for user queries.
- `pyjokes`: Provides random jokes to the user.

4.0 Data Utilization

Data Collection

Audio is captured from the microphone using the `pyaudio` library and the audio is stored in a `.wav` file in the `data/` directory, using a timestamp as the filename for unique identification.

Data Upload

The recorded audio file is then uploaded to a third-party service (AssemblyAI) for transcription. The system uses a POST request to send the `.wav` file data in chunks.

After the transcription process is completed, the system retrieves the text from the API response and logs the text to the `recording_records.txt` file.

Data Analysis

The text is then processed using string matching to identify specific keywords (such as “play,” “stop,” or “time”). Based on these keywords, the system executes the corresponding action.

Data Output

Using the `pyttsx3` library, the system converts text responses (such as the current time, a joke, or other feedback) into speech. The responses are also logged to the `recording_records.txt` file.

5.0 Personal Strategies

Learning

As we are new to AI technologies, we spent a significant amount of time familiarising ourselves with the technologies. We utilised YouTube and GitHub examples to learn about the technologies. It was a rewarding experience to learn different types of libraries and technologies and get to build a minimal functional prototype.

Challenges

As we are learning with YouTube, we followed the steps introduced in the video but it came to a place where we would need to pay for the real-time speech recognition services. We finally decided to use the basic plan to record the audio in a 5 seconds interval.

In situations where we faced situations that we cannot handle, we would use generative AI to get solutions. We would also use the World Wide Web and YouTube to search of alternative solutions.

7.0 Challenges & Future Enhancements

Although the AI assistant is well-equipped to handle the diverse roles of drivers in different situations, we can further improve it by addressing its potential challenges and enhancing its accessibility in the future.

7.1 Challenges

One of the main challenges is the assistant's current limitations. As technology continues to improve, the assistant may not be able to meet the changing needs of drivers, which could impact its effectiveness in the long term.

Besides, integrating the voice assistant smoothly with existing Grab systems will require significant time and effort. Drivers may also face difficulties adapting to voice AI systems, especially if they are not accustomed to using voice-based commands.

7.2 Future Enhancements

- Improve error handling for API failures, network issues, or microphone issues.
- Implement continuous listening in the background rather than using the fixed 5-second interval.
- Implement Natural Language Processing to support more commands
- Continuously update the voice model to support more languages and regional dialects. Additionally, enhance the assistant’s capabilities by adding features like an accident detector, which can automatically alert emergency services when it detects dangerous situations through sound.

- Use modular architecture and APIs for easy integration and updates, minimizing disruptions during system updates.
- Provide a comprehensive beginner tutorial to help drivers quickly get up to speed with the system. Also, implement personalization features to tailor the assistant's responses and functionality to individual driver preferences, creating a more intuitive and user-friendly experience.

8.0 Demonastation

Video Link:
https://drive.google.com/file/d/1IGRRFYTYixSpK0zIhcSIVo-OBi95VtyO/view?usp=sharing

Content Printed on Console for the above Demo:

```
Listening...
Recording saved to 20250412_175323.wav
20250412_175323.wav
Waiting 1 seconds...
What is the time now?
Transription saved!!
what is the time now?
05:53 PM
Listening...
Recording saved to 20250412_175337.wav
20250412_175337.wav
Find university.
Transription saved!!
find university.
Listening...
Recording saved to 20250412_175358.wav
20250412_175358.wav
Waiting 1 seconds...
Play football.
Transription saved!!
play football.
Listening...
Recording saved to 20250412_175411.wav
20250412_175411.wav
Waiting 1 seconds...

Transription saved!!
```

Content Logged on `recording_records.txt` file

```
20250412_175323: What is the time now?
Bot: Current time is 05:53 PM
20250412_175337: Find university.
Bot: A university (from Latin universitas 'a whole') is an institution of tertiary education and research which awards academic degrees in several academic disciplines.
20250412_175358: Play football.
Bot: Playing football.
20250412_175411:
Bot: What did you say?
```