

**Personal AI-Powered Conversation and Grammar-Checking Chatbot**

By

Su Yi Wen

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

**BACHELOR INFORMATION SYSTEMS (HONOURS)**

**BUSINESS INFORMATION SYSTEMS**

Faculty of Information and Communication Technology  
(Kampar Campus)

MAY 2023

## REPORT STATUS DECLARATION FORM

**Title:** PERSONAL AI-POWERED CONVERSATION AND GRAMMAR-CHECKING  
CHATBOT

**Academic Session:** May 2023

I SU YI WEN

declare that I allow this Final Year Project Report to be kept in

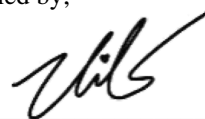
Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,

yiwen

(Author's signature)



(Supervisor's signature)

**Address:**

NO 22, JALAN INDAH 1/23

TAMAN UNIVERSITY INDAH

43300 SERI KEMBANGAN, SELANGOR

Aun Yichiet

Supervisor's name

**Date:** 14<sup>th</sup> September 2023

**Date:** 14 Sep 2023

<b>Universiti Tunku Abdul Rahman</b>			
Form Title : <b>Sample of Submission Sheet for FYP/Dissertation/Thesis</b>			
Form Number: <b>FM-IAD-004</b>	Rev No.: <b>0</b>	Effective Date: <b>21 JUNE 2011</b>	Page No.: <b>1 of 1</b>

**FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY**

**UNIVERSITI TUNKU ABDUL RAHMAN**

Date: 14<sup>th</sup> September 2023

**SUBMISSION OF FINAL YEAR PROJECT /DISSERTATION/THESIS**

It is hereby certified that SU YI WEN (ID No: 20ACB06456) has completed this final year project entitled "PERSONAL AI-POWERED CONVERSATION AND GRAMMAR-CHECKING CHATBOT" under the supervision of DR AUN YICHIET from the Department of COMPUTER AND COMMUNICATION TECHNOLOGY, Faculty of INFORMATION AND COMMUNICATION TECHNOLOGY.

I understand that University will upload softcopy of my final year project in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,

  
\_\_\_\_\_

(SU YI WEN)

## DECLARATION OF ORIGINALITY

I declare that this report entitled “**Personal AI-Powered Conversation and Grammar-Checking Chatbot**” is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : *yiwen*

Name : Su Yi Wen

Date : 14<sup>th</sup> September 2023

## ACKNOWLEDGEMENTS

I would like to express my sincere thanks and appreciation to my supervisors, Dr Aun Yichiet who has given me this bright opportunity to engage in an **AI Chatbot project**. It is my first step to establish a career in **building chatbot** field. I am immensely thankful to you. Lastly, I would like to express my appreciation to my parents and family for their unwavering love, support, and consistent encouragement during the entire journey.

# **ABSTRACT**

For many individuals who are learning the English language, it can be challenging to find a suitable practice partner, which can significantly hinder their progress. Additionally, research has shown that even if language learners are able to find someone with a comparable language proficiency level to practice with, it may not necessarily result in effective learning as they may be unable to identify and correct each other's errors. To address these issues, this project involves the creation of a Personal AI-Powered Conversation and Grammar-Checking Chatbot that provides English language learners with an opportunity to engage in conversations with a chatbot and improve their grammar skills with the assistance of a grammar checker and a paraphrased function. Additionally, a report that visualizes users' activities while using this project and their learning progress will be generated for their benefit.

# CONTENT TABLE

<b>DECLARATION OF ORIGINALITY .....</b>	<b>IV</b>
<b>ACKNOWLEDGEMENTS .....</b>	<b>V</b>
<b>ABSTRACT.....</b>	<b>VI</b>
<b>CONTENT TABLE .....</b>	<b>VII</b>
<b>LIST OF FIGURES .....</b>	<b>X</b>
<b>LIST OF TABLES .....</b>	<b>XVI</b>
<b>CHAPTER 1 INTRODUCTION.....</b>	<b>1</b>
1.1 Problem Statement and Motivation .....	3
1.1.1 Learner is Lacking English Learning Environment.....	3
1.1.2 Lacking interaction, grammar-checker and paraphraser in existing language learning app.....	4
1.1.3 lacking report in existing language learning app for user to keep track their progress .....	4
1.2 Project Objectives .....	5
1.3 Project Scope and Direction.....	7
1.4 Contributions.....	8
<b>CHAPTER 2 LITERATURE REVIEWS.....</b>	<b>9</b>
2.1 Overview of Common Chatbot Uses in Different Industries.....	9
2.2 Previous Research on English Learning in Malaysia .....	10
2.3 Previous Research of Chatbot in language learning .....	11
2.4 Todays' Language Learning Chatbot.....	13
2.5 Todays' Conversational AI Chatbot .....	19
2.6 Review on Grammar Analysis .....	23
2.7 Review on paraphrasing.....	25
2.8 Research on Weekly Grammarly Insights Reports .....	27
2.9 Research on Tone analysis.....	30

2.10 Reason to develop AI-conversational Chatbot in English language learning.....	32
--	----

**CHAPTER 3 PROPOSED METHOD/APPROACH .....34**

3.1 System Requirement .....	34
3.1.1 Hardware.....	34
3.2 System Design .....	34
3.2.1 System Architecture.....	34
3.2.2 Integration of Grammarly Text Editor SDK and APIs .....	48
3.3 English Practice Chatbot Report design and features .....	50
3.3.1 How the Message per day and month calculated.....	52
3.3.2 How the Sentiment Distribution calculated. ....	52
3.3.3 How the Unique Words/Vocabulary calculated. ....	53
3.3.4 How the Grammar Analysis calculated. ....	53
3.3.5 How the Tone calculated. ....	54
3.4 Overview of the English Practice Chatbot Report 's user interface.....	56
3.5 How the English Practice Chatbot Report uses learning analytics .....	58
3.6 Timeline .....	58
3.7 Future Development Plan/ Future Work.....	58

**CHAPTER 4 IMPLEMENTATION.....60**

4.1 Overview of the development process for English Practice Chatbot .....	60
4.1.1 Development tools and technologies used for English Practice Chatbot.....	60
4.2 Integration of Open AI API, Grammarly SDK and paraphraser tool .....	61
4.2.1 how the chatbot integrates Open AI API .....	61
4.2.2 How the chatbot integrates Grammarly Text Editor SDK.....	62
4.2.3 How the chatbot integrates Smodin paraphraser API.....	63
4.3 English Practice Chatbot Report implementation.....	63
4.3.1. Overview of how the English Practice Chatbot Report was built. ....	63
4.3.2. Python Libraries Imported .....	64
4.3.2.1 Streamlit.....	64
4.3.2.2 Streamlit Lottie .....	64
4.3.2.3 JSON.....	65
4.3.2.4 Pandas .....	65
4.3.2.5 NumPy .....	65
4.3.2.6 Matplotlib.....	66
4.3.2.7 Plotly .....	66
4.3.2.8 NLTK (Natural Language Toolkit).....	66
4.3.2.9 WordCloud.....	67



4.3.2.10 LanguageTool Python .....	67
4.3.2.11 Datetime .....	67
4.3.2.12 Base64 .....	68
4.3.2.13 TextBlob .....	68
4.3.3. Components in English Practice Chatbot Report.....	68
4.3.3.1 preprocess_text .....	68
4.3.3.2 Message per Day & Month .....	70
4.3.3.3 Sentiment Distribution .....	72
4.3.3.4 Unique Words/ Vocabulary .....	73
4.3.3.5 Grammar Error .....	74
4.3.3.6 Tone Distribution .....	75
4.4 Chatbot training and Testing.....	77
4.5 Smodin Paraphraser Testing .....	81
4.6 Grammarly Text Editor SDK Testing.....	82
4.7 English Practice Chatbot Report training and Testing.....	84
4.8 Deployment.....	94
4.9 User Testing .....	94
4.10 Enhancement.....	103
<b>CHAPTER 5 CONCLUSION.....</b>	<b>108</b>
<b>REFERENCES.....</b>	<b>110</b>
<b>APPENDIX.....</b>	<b>A</b>
<b>POSTER.....</b>	<b>K</b>
<b>PLAGIARISM CHECK RESULT.....</b>	<b>L</b>
<b>FYP 2 CHECKLIST .....</b>	<b>Q</b>

## LIST OF FIGURES

<b>Figure Number</b>	<b>Title</b>	<b>Page</b>
<i>Figure 2.4.1</i>	Follow and Read Mode	13
<i>Figure 2.4.2</i>	Read and Answer Mode	13
<i>Figure 2.4.3</i>	Eggbun learning process interface	14
<i>Figure 2.4.4</i>	Chatting mode in Andy	15
<i>Figure 2.4.5</i>	Chatting mode in Oral Craft	16
<i>Figure 2.5.1</i>	Chat GPT	20
Figure 2.6.1	Simple sign system	24
<i>Figure 2.6.2</i>	Example of a system for Thing	24
<i>Figure 2.8.1</i>	Grammarly Productivity	27
<i>Figure 2.8.2</i>	Grammarly Mastery	27
<i>Figure 2.8.3</i>	Grammarly Vocabulary	28
<i>Figure 2.8.4</i>	Grammarly Top3 Mistakes	28
<i>Figure 3.2.1</i>	Register Screen	35
<i>Figure 3.2.2</i>	Login Screen	35
<i>Figure 3.2.3</i>	Forget Password Screen	36
<i>Figure 3.2.4</i>	User email	36
<i>Figure 3.2.5</i>	Reset password screen	37
<i>Figure 3.2.6</i>	Home Screen	37
<i>Figure 3.2.6</i>	Profile Screen	37
<i>Figure 3.2.7</i>	About Screen	38

<i>Figure 3.2.8</i>	Start chatting	38
<i>Figure 3.2.9</i>	Chatting Screen	39
<i>Figure 3.2.10</i>	Correcting Grammar Screen	39
<i>Figure 3.2.11</i>	After Correcting Grammar Screen	40
<i>Figure 3.2.12</i>	Chatbot Reply Screen	40
<i>Figure 3.2.13</i>	Example Session_1	41
<i>Figure 3.2.14</i>	Example Session_2	41
<i>Figure 3.2.14</i>	Start New Session	42
<i>Figure 3.2.15</i>	More Features Icon	42
<i>Figure 3.2.16</i>	More Features Screen	42
<i>Figure 3.2.17</i>	Selected Session's Chat History	43
<i>Figure 3.2.18</i>	Edit Session Name Screen	43
<i>Figure 3.2.19</i>	Session Name has been updated	44
<i>Figure 3.2.20</i>	Retrieve Paraphrased Sentences	44
<i>Figure 3.2.21</i>	Keyword search function	44
<i>Figure 3.2.22</i>	Retrieve Chat Messages Screen	45
<i>Figure 3.2.23</i>	.csv file created	45
<i>Figure 3.2.24</i>	Content of .csv file	45
<i>Figure 3.2.25</i>	English Practice Chatbot Report	46
<i>Figure 3.2.26</i>	To print/export English Practice Chatbot Report	46
<i>Figure 3.2.27</i>	Grammarly button	49

<i>Figure 3.3.1</i>	<i>Textblob process</i>	54
Figure 3.4.1	Upload interface of English Practice Chatbot Report	56
Figure 3.4.2	Waiting Screen	56
Figure 3.4.3	Result Screen	57
Figure 3.6.1	Gantt Chart	58
Figure 4.2.1	Integrating Open AI API	61
Figure 4.2.2	Install Text Editor SDK	62
Figure 4.2.3	Add Grammarly Text Editor	62
Figure 4.2.4	Integrating Paraphraser API	63
Figure 4.3.1	import streamlit	64
Figure 4.3.2	import streamlit_lottie	65
Figure 4.3.3	import json	65
Figure 4.3.4	import pandas	65
Figure 4.3.5	import numpy	65
Figure 4.3.6	import matplotlib.pyplot	66
Figure 4.3.7	import plotly.express	66
Figure 4.3.8	import nltk	66
Figure 4.3.9	import wordcloud	67
Figure 4.3.10	import language_tool_python	67
Figure 4.3.11	import datetime	68
Figure 4.3.10	import base64	68

Figure 4.3.11	import textblob	68
Figure 4.3.12	preprocess_text(text)	68
Figure 4.3.3.2(a)	message per day & month	70
Figure 4.3.3.2(b)	message per month	71
Figure 4.3.3.3(a)	to plot sentiment distribution (in main())	72
Figure 4.3.3.3(b)	lines in language_proficiency()	72
Figure 4.3.3.3(c)	calculate_sentiment_score(text)	73
Figure 4.3.3.4(a)	to show the unique words/ vocabulary (in main())	73
Figure 4.3.3.4(b)	lines in language_proficiency()	74
Figure 4.3.3.5(a)	import language_tool_python	74
Figure 4.3.3.5(b)	defining tool	74
Figure 4.3.3.5 (c)	calculate_grammar_error	74
Figure 4.3.3.6	detect_tone()	75
Figure 4.4.1	Test_1	78
Figure 4.4.2	Test_2	79
Figure 4.4.3	Test_3	80
Figure 4.4.4	Test_4	80
Figure 4.4.5	Test_5	81
Figure 4.6.1	Test_1	82
Figure 4.6.2	Test_2	82

Figure 4.6.3	Test_3	83
Figure 4.6.4	Test_4	83
Figure 4.7.1	snapshot of input collected	84
Figure 4.7.2	section 1 of the report	84
Figure 4.7.3(a)	section 2 of the report	85
Figure 4.7.3(b)	section 2 of the report	85
Figure 4.7.3(c)	section 2 of the report	86
Figure 4.7.4(a)	section 3 of the report	86
Figure 4.7.4(b)	section 3 of the report	87
Figure 4.7.4(c)	section 3 of the report	87
Figure 4.7.5 (a)	section 4 of the report	87
Figure 4.7.5 (b)	section 4 of the report	89
Figure 4.7.5 (c)	section 4 of the report	90
Figure 4.7.6 (a)	section 5 of the report	90
Figure 4.7.6 (b)	section 5 of the report	91
Figure 4.7.6 (c)	section 5 of the report	91
Figure 4.7.7 (a)	section 6 of the report	92
Figure 4.7.7 (b)	section 6 of the report	93
Figure 4.7.7 (c)	section 6 of the report	93
Figure 4.9.1	Survey Question 1	94
Figure 4.9.2	Survey Question 2	95

Figure 4.9.3	Survey Question 3	95
Figure 4.9.4	Survey Question 4	96
Figure 4.9.5	Survey Question 5	96
Figure 4.9.6	Survey Question 6	97
Figure 4.9.7	Survey Question 7	98
Figure 4.9.8	Survey Question 8	99
Figure 4.9.9	Survey Question 9 (a)	100
Figure 4.9.10	Survey Question 9 (b)	101
Figure 4.9.11	Survey Question 9 (c)	101
Figure 4.9.12	Survey Question 9 (d)	102
Figure 4.10.1	generateRandomTopic()	103
Figure 4.10.2	Random Topic Button	104
Figure 4.10.3	Random Topic Generated	104
Figure 4.10.4	startSpeechRecognition()	105
Figure 4.10.5	Microphone Button	106
Figure 4.10.6	Speaking mode started	106
Figure 4.10.7	Speaking done	106

## LIST OF TABLES

<b>Table Number</b>	<b>Title</b>	<b>Page</b>
Table 2.4.1	Summary of language learning platform	17
Table 2.4.2	Summary of entertainment purpose AI-chatbot application	22
Table 3.1	Specifications of laptop	34



## CHAPTER 1 Introduction

The term "Chatbot" is a widely discussed topic across various fields globally, including online business, IT, education, and healthcare. Many studies have been conducted to understand the definition of a chatbot, as well as its benefits and limitations. Chatbots can be classified into different types, including rule-based chatbots (also known as click bots), AI-powered chatbots, and application-oriented chatbots, which are a blend of rule-based and intelligent conversational systems. However, it may be more beneficial to differentiate chatbots based on their use cases and the user's perspective, rather than strictly on technical attributes.[1] The popularity of conversational AI chatbots is rapidly increasing, mainly due to advancements in natural language processing and deep learning. These technologies enable chatbots to replicate human conversation and provide a more personalized experience for users. As a result, conversational AI chatbots are being increasingly adopted across various fields, including e-commerce, customer service, education, and healthcare.[2]

In November 2022, the introduction of Chat-GPT represented a significant breakthrough in the field of chatbots and AI. The emergence of these technologies has captured widespread attention, particularly among those without an IT background. The development of Chat-GPT has enabled individuals to experience the capabilities of AI in a more tangible way. As a result of this growing interest, numerous research studies and news articles have been conducted to explore the impact of chatbots and AI on various industries and fields.

In this project, the focus will be on using Conversational AI-Chatbot for English language learning. The recent growth of chatbots in educational contexts, particularly in language learning, has prompted an exploration of their potential use in this field. For instance, the language learning online software sector has experienced substantial expansion, with various applications such as Duolingo, Mondly, and Rosetta Stone attracting massive user populations. As such, this project aims to investigate how a Conversational AI-Chatbot can be utilized in language learning and to evaluate its effectiveness in improving language proficiency.[3] There are three key factors that contribute to the effectiveness of chatbots in language learning. Firstly, chatbots are accessible 24/7, providing learners with the flexibility to practice their language skills

## CHAPTER 1

at any time. Secondly, they offer learners access to a broad range of language information that may not be available from human language partners, particularly in cases where learners and their peers are at similar proficiency levels. Finally, chatbots can serve as an engaging language practice partner who can converse with learners endlessly, providing them with valuable opportunities to practice the target language consistently.[4] There is a growing interest in using chatbots as language learning assistants, and not just limited to popular language learning applications like Duolingo and Mondly. Other messaging bots such as Replika, Personal AI, Simsimi, Anima, etc. are also being used by people for this purpose.[5] these chatbots are gaining attention due to their ability to converse with students using natural language, making the language learning process more interactive and engaging. [6] Interaction plays a crucial role in language learning, as it involves engagement with peers, teachers, and other experts. This experience enables students to receive understandable feedback on their performance, gain new insights, and refine their output.[7] In today's technological age, the development of conversational AI chatbots has made it possible for learners to interact in a new way. AI chatbots are capable of conversing with users within a specific domain by providing intelligent responses through natural language processing. With the chatbot providing both textual and audio inputs, learners can enhance their reading and listening skills. According to scholars, this practice can eventually lead to an improvement in foreign language speaking and writing abilities.[9]

## 1.1 Problem Statement and Motivation

### 1.1.1 Learner is Lacking English Learning Environment

In the 21st century, the English language has become an integral part of the educational curriculum in many countries where it is taught as a second or foreign language.[8] According to the Malaysian Education Blueprint, the operational English proficiency level among students is significantly lower. This is evident from the 2011 SPM English paper, where only 28% of students were able to achieve the minimum credit required as per the Cambridge 1119 requirements.[9] Foreign language (FL) learners often struggle with expressing themselves clearly and fluently in both written and spoken forms, which is the most challenging issue they face.[10] In addition, current reports suggest that social activists remain concerned about the weak command of English among Malaysians and have called for action to bring the country's human resources up to world standards. Therefore, it is necessary to investigate the challenges faced by learners in order to achieve this goal. **The lack of an English learning environment** is the most challenging obstacle to overcome. Learners who do not have access to English speakers miss out on the opportunity to develop their language skills through meaningful conversations, which can lead to a decline in their motivation to learn the language due to its limited practical value.[11] According to Dr. Nor Fadzleen Sa'don, a senior English language teacher at SMK Uda 2, students may miss opportunities to improve their language skills and overcome psychological barriers. This is due to the fear of making mistakes when speaking, which can cause learners to be less talkative. As a result, weaker students may rely on their teacher or classmates to do most of the talking, which can hinder their ability to develop their language proficiency since interaction is a crucial aspect of language learning.[12] Additionally, while **students may engage in foreign language practice with peers at a similar proficiency level**, they may require additional support to advance their language skills. Tan Sri Muhyiddin Yassin, the former Deputy Prime Minister, emphasized the importance of Malaysians mastering the English language, as it plays a critical role in nation-building.[9] In short, it can be inferred from these statements that learners are struggling with expressing themselves clearly and fluently in the English language. Meanwhile, teachers are facing difficulties in teaching English due to learners' reluctance to participate in speaking activities and limited time allocated in the current school timetable for language learning.

### **1.1.2 Lacking interaction, grammar-checker and paraphraser in existing language learning app**

Language learning involves interaction with others, and learners have adopted various methods to learn through interaction, including conversing with their teachers or peers and using chatbot applications. However, current language learning applications like Duolingo and Mondly are designed to teach in a game-like format, rather than through conversation, resulting in inadequate interactive features. Learners are only required to select the correct answer from multiple choices, which does not provide enough opportunities to express themselves. Research indicates that many individuals struggle to express themselves verbally or in writing, despite being able to read, write, and comprehend English. This is partly due to the lack of English communication practice. [13] Language learners can internalize their newly acquired knowledge through conversation practice, which involves combining several cognitive processes to produce oral communication. A study suggests that conversation practice acts as a link between syntax, punctuation, and vocabulary, strengthening the connections among these linguistic features as learners create sentences.[14] Conversational AI chatbots, such as Replika and Anima, have offered a platform for English language learners to engage in daily conversation and practice their English. However, since these chatbots were primarily designed to function as AI companions, they do not include grammar checking or vocabulary teaching features. This limitation has led users to struggle with determining whether the sentences they generate are grammatically correct or not.

### **1.1.3 lacking report in existing language learning app for user to keep track their progress**

Ensuring the effective support of English learners (ELs) requires a personalized, intentional, and evidence-based approach. By utilizing progress monitoring, educators can identify the specific academic and language needs of their EL students and then design and implement instructional programs that foster optimal language acquisition and content mastery. [15]This emphasizes the importance of providing users with a progress report to track their learning journey. Unfortunately, popular language learning applications like Duolingo, Andy, and Eggbun currently lack this crucial feature.

### 1.2 Project Objectives

The aim of this project is to address the limitations of current English language learning applications by integrating conversational AI-chatbot technology, such as Replica, into the learning environment. The primary **objective is to create an interactive English learning environment that allows users to practice their language skills by engaging in conversations with a chatbot with real-time grammar checker and paraphraser, rather than relying solely on traditional lessons.** To achieve this, the project will focus on developing a chatbot that utilizes the Grammarly Text Editor SDK to provide real-time grammar suggestions to users. This will enable users to receive immediate feedback and correct their mistakes while conversing with the chatbot. Furthermore, the chatbot will use the Open-AI API to facilitate natural and fluent conversation with users. In addition to grammar suggestions, the chatbot will incorporate a paraphraser API to assist users in learning alternative ways of expressing their thoughts and ideas in English, which is not typically available in existing English learning platforms. This feature will help users learn from their mistakes and gain confidence in their language skills.

Besides, this project also aims to **provide an English Practice Chatbot Report for user to visualize their activity and the learning progress** to address the problem of lacking report in existing language learning app for user to keep track their progress. This will be done using Streamlit. The concept for the report came from Grammarly's weekly report, which contained metrics for vocabulary, activity (total word count for the week and how frequently you used Grammarly compared to other users), top grammar errors, and tone.[16] This report will present metrics such as daily and monthly message frequency, overall and monthly sentiment distribution, usage of distinct words, identification of grammar mistakes, and distribution of tones The "Message per Day and Month" metric provides a glimpse into consistent practice, aiding users in gauging their commitment. The "Sentiment Distribution" metrics offer an emotional insight, enabling users to explore their language connection. Tracking "Unique Words/Vocabulary" showcases language expansion. "Grammar Error" reveals prevalent mistakes, facilitating learning.

Overall, this project seeks to offer users a personalized and interactive English learning experience while overcoming the limitations of current language learning

## CHAPTER 1

applications. By utilizing conversational AI-chatbot technology, this project has the potential to transform the way English language learners improve their language skills.

### 1.3 Project Scope and Direction

The project aims to develop an innovative conversational chatbot that addresses the limitations of current language learning apps, such as Duolingo and Mondly. The chatbot will be specifically designed to facilitate interactive and engaging conversation between users and the AI bot. The main objective of the project is to provide users with a personalized and interactive English language learning experience, utilizing conversational AI-chatbot technology.

The chatbot will incorporate Grammarly Text Editor SDK and Open-AI APIs to provide users with real-time grammar correction and natural language processing, enabling users to practice their language skills through interactive conversation rather than multiple-choice questions. The chatbot will provide users with an immersive learning environment that simulates a conversation with a native speaker, allowing learners to gain confidence in their speaking, writing, and comprehension skills.

The chatbot's conversational interface will be designed to be user-friendly and engaging, using chatbot best practices to provide a natural and personalized learning experience. The chatbot will also include a paraphraser API, providing users with alternative ways of expressing their thoughts and ideas in English, helping learners to develop their language skills further. The chatbot's chat logs will be analyzed to identify user patterns and areas for improvement, enabling the chatbot to adapt and improve its responses.

The project's direction will be to develop a beta version of the chatbot and conduct user testing to validate its efficacy and user-friendliness. The beta testing will involve a diverse group of English language learners to ensure that the chatbot is inclusive and suitable for a broad range of users. The feedback from the beta testing will be used to refine the chatbot's functionality, improve its performance and user-friendliness, and add new features.

The project's final deliverable will be a fully functional and polished chatbot complemented by an integrated report to visualize the activity and the progress of user. The chatbot will provide a personalized and engaging English language learning experience, utilizing cutting-edge conversational AI-chatbot technology. The chatbot will be designed to be scalable and adaptable, enabling it to evolve and improve over time, based on user feedback and advances in AI technology.

## **1.4 Contributions**

The aim of this project is to create a novel chatbot that combines the features of language learning apps like Duolingo and Mondly with conversational chatbot apps such as Replika. The proposed chatbot will provide learners of English with an interactive platform for practicing their language skills through conversation, in contrast to the conventional method of selecting answers from multiple-choice questions.

The chatbot will provide a comfortable setting for learners to enhance their English skills, allowing them to practice anytime without worrying about feeling embarrassed or stressed. By including tools to correct grammar mistakes, learners can easily spot their errors and learn from them – something that might not be possible when practicing with peers who share a similar language level. Moreover, the chatbot's paraphrasing feature will let learners experiment with different ways of expressing their thoughts, aiding in the development of their language abilities. Lastly, a report will be constructed using user input to visually display user activity and progress. This report will present metrics such as daily and monthly message frequency, overall and monthly sentiment distribution, usage of distinct words, identification of grammar mistakes, and distribution of tones



## **CHAPTER 2 Literature Reviews**

### **2.1 Overview of Common Chatbot Uses in Different Industries**

A chatbot is a type of computer software that emulates and handles human dialogue, whether it be written or spoken, permitting individuals to engage with electronic devices as if they were conversing with an actual person. Chatbots can vary from a basic program that answers a single question to advanced digital assistants that acquire and analyse information to offer more customized and personalized responses.[17] Chatbots have found utility in various industries, including private schools and educational institutions. By placing a chatbot on their website, these institutions can offer assistance to students and parents seeking basic payment and registration information. Additionally, incoming calls can be directed to the chatbot, enabling faster service and freeing up phone lines. AI chatbots can also be utilized by regional airlines as a first-line of defence for handling incoming queries. They can quickly provide updates to customers regarding flight information, including boarding times and gate numbers. Furthermore, they can manage changes and cancellations to flights, as well as facilitate payments for upgrades and transfer fees.[18] Chatbots are being utilized in the healthcare industry to automate repetitive and lower-level tasks that representatives would typically perform. This automation allows healthcare professionals to focus their attention on more complex tasks, ultimately leading to improved patient care. Advancements in chatbot technology over the past decade have resulted in significant growth in the healthcare sector and other AI tools. Chatbots can detect symptoms, manage medications, and provide assistance with chronic health problems. They are also capable of guiding individuals with severe illnesses and assisting them in scheduling appointments with healthcare professionals.[19] From the use cases above, we able to conclude that chatbots are already widely used by humans in different industries and people are gaining benefit from it.

## 2.2 Previous Research on English Learning in Malaysia

Globalization has resulted in English becoming the dominant international language of the 21st century, with approximately 1.75 billion individuals worldwide possessing a reasonable level of proficiency in the language. In Malaysia, a country with a multicultural population and where the official language is Bahasa Malaysia, English (British) holds significant historical importance. Consequently, investing in English language skills has become a popular trend in Malaysia. However, numerous individuals have expressed disappointment in the communication abilities of local graduates in the Malaysian context.[20] Based on the 2016 global ranking of countries by English skills, countries such as Russia (40%), Japan (43%), Turkey (37%), Mongolia (42.2%), Kazakhstan (47.4%), Cambodia (39.4%), and Laos (38.6%) have been found to have a lower percentage of English proficiency. This is often attributed to factors such as limited exposure to an English-speaking environment and lack of sufficient opportunities for English language learning. [21] In a non-English-speaking environment, such as Japan, China, Korea, and Russia, English language learners may struggle to practice the language outside of the classroom as the predominant language used in society is their native tongue. On the other hand, a broken-English speaking environment refers to the development of dialects unique to a particular country or region, such as the Korean bamboo language, Malaysian Manglish, and Singapore's Singlish. For English learners in Malaysia, the environment outside of the classroom presents a significant challenge. Malaysia is a multicultural country with the government language Bahasa Malaysia, which is a combination of Tamil and Chinese Mandarin or Cantonese. While government agencies require communication in BM, people on the streets prefer to communicate in their "comfortable language," and some families speak their original language, such as Mandarin, Cantonese, Tamil, and Hindu. As a result, opportunities for pure English language practice and conversation are limited, resulting in a lower percentage of individuals who can communicate effectively in English. According to the 2016 world ranking of countries by English skills, countries with a non-English speaking environment and a broken English environment, such as Russia (40%), Japan (43%), Turkey (37%), Mongolia (42.2%), Kazakhstan (47.4%), Cambodia (39.4%), and Laos (38.6%), had the lowest percentage of English proficiency.[20]

### 2.3 Previous Research of Chatbot in language learning

The use of chatbots for language learning is not a new concept. These bots have become increasingly important in the field of education due to their interactive nature, as opposed to traditional e-learning systems. Some examples of chatbots designed for learning foreign languages include Cleverbot, Mondly, Andy, CSIEC, and Autotuter.[22] Autotuter is a chatbot that utilizes multidisciplinary approaches to enhance learning by facilitating conversations with learners. On the other hand, Mondly and Andy English Bot enable learners to choose from multiple responses to select the most suitable one, thus improving their language learning experience. [23] While the CSIEC system generates communicative responses based on user input and dialogue context, the CSIEC system generates communicative responses.[24]

Numerous studies have explored the potential of chatbots in language learning. Fryer and Carpenter identified six ways in which chatbots can assist learners. Firstly, learners may feel more comfortable speaking with a chatbot "tutor" than with a human tutor. Secondly, learners may be willing to review the same material repeatedly with a chatbot. Thirdly, chatbots often provide both text and synthesized speech, enabling students to practice both their reading and listening skills. Fourthly, chatbots are a novel and exciting learning tool for students. Fifthly, chatbots offer an opportunity for students to use a variety of language structures. Lastly, chatbots may provide learners with immediate feedback on their grammar and spelling. [25] At the same time, using chatbots in language learning has six benefits, according to Haristani, including lessened language anxiety, widespread availability, multimodal practise, novelty effect, a wealth of contextual vocabulary, and efficient feedback. [26]. Besides, research shows that conversation with a chatbot preceding classroom discussion increased the number of student contributions to the discussion.[27]

A strategy for promoting successful English language learning outcomes is to provide students with meaningful input in low-stress settings that includes messages they find engaging and allows them to construct sentences at their own pace. [28] Wang conducted a study on the experiences of ESL students who used a for-profit chatbot tutor. The findings suggest that chatbot technology can fulfill its promises of improving interactive English learning environments through communication.[29] A study

## CHAPTER 2

conducted on the use of chatbots such as ALICE and George for English language learning has revealed that these AI chatbots have significant potential for use both inside and outside of a language classroom. They can enable language learners to practice their language skills and build confidence in a personalized, stress-free manner at their own pace and performance.[29] Moreover, a study on the use of Cleverbot revealed that the majority of students found it enjoyable to use and felt more at ease conversing with the chatbot than with a teacher or fellow student.[26]

A college professor in China distributed the CSIEC (Computer Simulator in Educational Communication) chatbot system to her students for an introductory English course and requested their feedback. The majority of students treated the chatbot as a friend rather than a tutor, and it increased their desire to use technology for language learning. Some students also found it helpful to chat with the chatbot to practice their English, even though it couldn't correct their mistakes. However, issues with the chatbot's technology led to poor learning outcomes, such as redundant or unrelated responses. Despite these issues, the CSIEC improved students' educational opportunities, and emotional states and experiences played a significant role in their learning. The professor believes that the CSIEC can meet the rising demand for language practice among users.[30]

While there is considerable research suggesting that AI chatbots can be helpful for language learning, there are also documented drawbacks. The novelty effect of using chatbots has led to concerns that they may not be the ideal long-term instructor, as student interest in chatbot conversations tends to decline over time. As a result, students may need additional resources and support to maintain their motivation and engagement with chatbot-based language learning.[31] For example, comparing speaking tasks with a chatbot and a human partner showed that students were much less interested in the chatbot task than the human partner task due to the novelty effect. [31] Furthermore, a study reported that chatbots often provide irrelevant responses, and there are concerns regarding the accuracy rate of the combined categories of grammar and meaning. [32] In addition, one study also found that chatbots are only helpful for advanced or very keen language students. [26]

## 2.4 Today's Language Learning Chatbot



**Figure 2.4.1 Follow and Read Mode**     **Figure 2.4.2 Read and Answer Mode**

With the advancements in technology, individuals now have the option to learn using chatbots, allowing for self-directed learning rather than attending traditional courses. The proliferation of language-learning applications in the market is a testament to this shift. Unlike traditional methods, software and applications offer affordability, convenience, interactivity, and flexibility, making learning both practical and enjoyable.[33] Duolingo was the first free language-learning app that could compete with expensive paid programs, offering self-paced assignments to help learners build a foundational knowledge of multiple languages or improve their proficiency in a language they already know.[34] There are numerous activities available for users to learn through gamification on Duolingo. The platform awards experience points (XP) to learners for their progress, including completing individual courses, placement tests, checkpoint quizzes, skill practice, test outs, and stories in various languages. Duolingo has become the most popular language-learning website and educational app, boasting more than 500 million users. Since its launch eleven years ago, the platform has grown significantly. Prior to the outbreak of the 2020 pandemic, there were already 300 million users.[33] From this data, we can observe that a large proportion of individuals these days are open to learning through chatbot applications.[33] Despite its fascinating features, Duolingo has received criticism from many users who remain

## CHAPTER 2

dissatisfied with its effectiveness in language learning. Even after studying and maintaining their streak for days, many users have found that they are only able to speak the language at an intermediate level. This is because the app lacks opportunities for interpersonal communication and speaking practice, which are essential for achieving fluency in a language.[35] This user feedback once again demonstrates the value of interaction in language learning.



**Figure 2.4.3 Eggbun learning process interface**

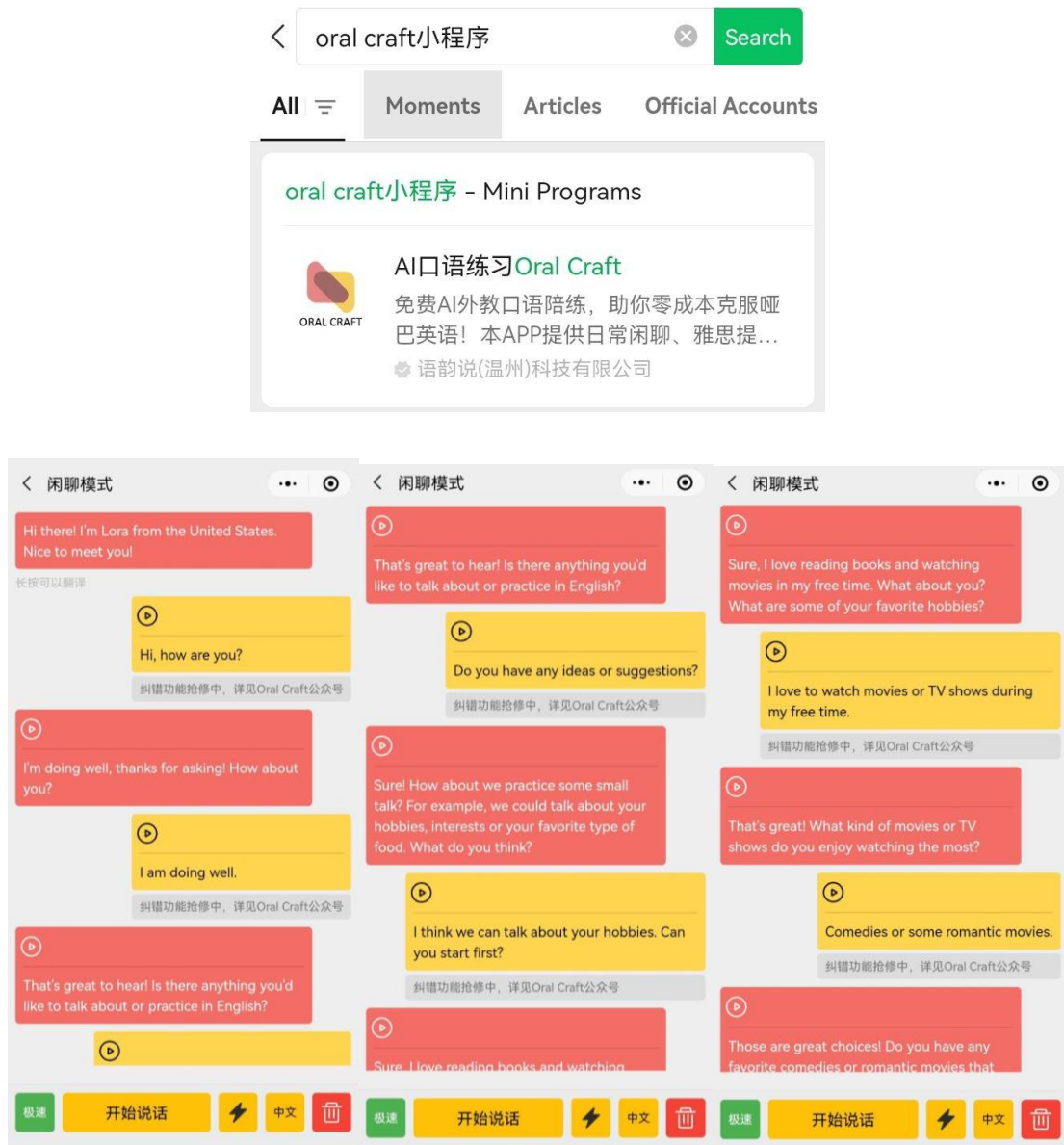
Eggbun is a language learning application for English, French, Japanese, Bahasa Indonesia, and Mandarin speakers who want to learn Korean. The app features a cute character named Lanny, who engages in conversations with learners and teaches them the Korean language. The course curriculum begins with an alphabet overview and then progresses to lessons on borrowed words, numerals, pronunciation, classroom vocabulary, greetings, self-introductions, markers, conjugation, real-life conversations, and travel. Each session includes various levels and quizzes to assist learners in their progress, along with audio and word lists. Eggbun is a popular language learning app that offers a unique learning experience to its users.[36] One noteworthy feature of Eggbun, shown in Figure 2.4.3, is that everything is conversational, including the vocabulary and phrases featured in the courses and the teaching style, which is beneficial for getting learners familiar with that aspect of language learning. However,

the lessons are pre-planned, and learners have no control over what they learn in any given class, nor can they choose which terminology is taught or ask clarifying questions. Moreover, similar to Duolingo, when learners make errors, they must retype their answer until it is correct, and there are no real penalties or ways to ask questions when they answer incorrectly or need clarification. This approach may encourage guessing rather than understanding, depending on how learners approach it.[37]



**Figure 2.4.4 Chatting mode in Andy**

In addition to language learning chatbots, there are also AI-powered chatbots available for language learning such as Andy. Recognizing that interaction is a crucial component in language learning, the development team utilized AI technology to create Andy, an application that relies on a virtual character, a robot named "Bot" with artificial intelligence, to initiate communication with the learner in simplified English. The virtual character gradually advances the learner's language proficiency until they become comfortable speaking in English and can engage in educational conversations with real people.[38] Despite the effort put in by the development team of Andy to integrate AI technology to facilitate language learning, there is a crucial aspect that has been overlooked - the grammar component. While the application offers grammar lessons, it lacks a grammar-checking feature to verify the accuracy of the learner's sentences during the conversation. As a result, the learner may be unable to learn from their mistakes or determine whether they are applying the knowledge correctly or not.



*Figure 2.4.5 Chatting mode in Oral Craft*

On March 1st, 2023, WeChat introduced a new mini program named Oral Craft which serves as a free AI tutor designed for English language learners to practice their conversational skills. The program offers four modes, including Small Talk mode, Specialized Practice mode, IELTS Improvement mode, and Text Adventure Game mode. The Small Talk mode, as shown in the accompanying image, involves a conversation between the user and a chatbot named Lora. While users can engage in a conversation with Lora, the program does not provide any grammar checker or paraphrasing function, leaving users unable to learn from their mistakes and rely solely on the bot's responses for their learning.



## CHAPTER 2

By reviewing the three platform, Duolingo, Eggbun, Andy, Oral Craft. A summary of these three platforms has shown in the table 1.

<b>App/ Chatbot name</b>	<b>Purpose</b>	<b>Strength</b>	<b>Weaknesses</b>
<b>Duolingo</b>	Language learning	<ul style="list-style-type: none"> <li>- help you learn grammar and vocabulary through interactive spaced repetition.</li> <li>- Learning through playing</li> <li>- More than 30 language available to learn by using English</li> <li>- Available 24/7</li> </ul>	<ul style="list-style-type: none"> <li>- Not able to practice what have learn by having conversation with the bot</li> <li>- Some sentences would not normally be used in natural speech</li> <li>- No grammar-checking feature</li> <li>- No paraphrase feature</li> </ul>
<b>Eggbun</b>	Language Learning (Korean)	<ul style="list-style-type: none"> <li>- The artificial intelligence chatbot acts like a tutor. Lanny, as she's called, focuses on providing you with guidance and direction in learning your target language. The interaction offers text-based dialogue along with audio examples</li> <li>- Available 24/7</li> </ul>	<ul style="list-style-type: none"> <li>- Not able to practice what have learn by having conversation with the bot</li> <li>- Only able to learn Korean</li> <li>- No grammar-checking feature</li> <li>- No paraphrase feature</li> </ul>
<b>Andy</b>	Language learning	<ul style="list-style-type: none"> <li>- Study language by actually using it in a conversation. Take grammar lessons and learn new words every day!</li> <li>- Available 24/7</li> </ul>	<ul style="list-style-type: none"> <li>- No grammar-checking feature</li> <li>- No paraphrase feature</li> </ul>

<b>Oral Craft</b>	Language learning (English)	<ul style="list-style-type: none"> <li>- Available 24/7</li> <li>- User able to learn through conversation</li> <li>- Translator are provided</li> </ul>	<ul style="list-style-type: none"> <li>- No grammar checking feature</li> <li>- No paraphrase feature</li> </ul>
-------------------	-----------------------------	--	--

***Table 2.4.1 – Summary of language learning platform***

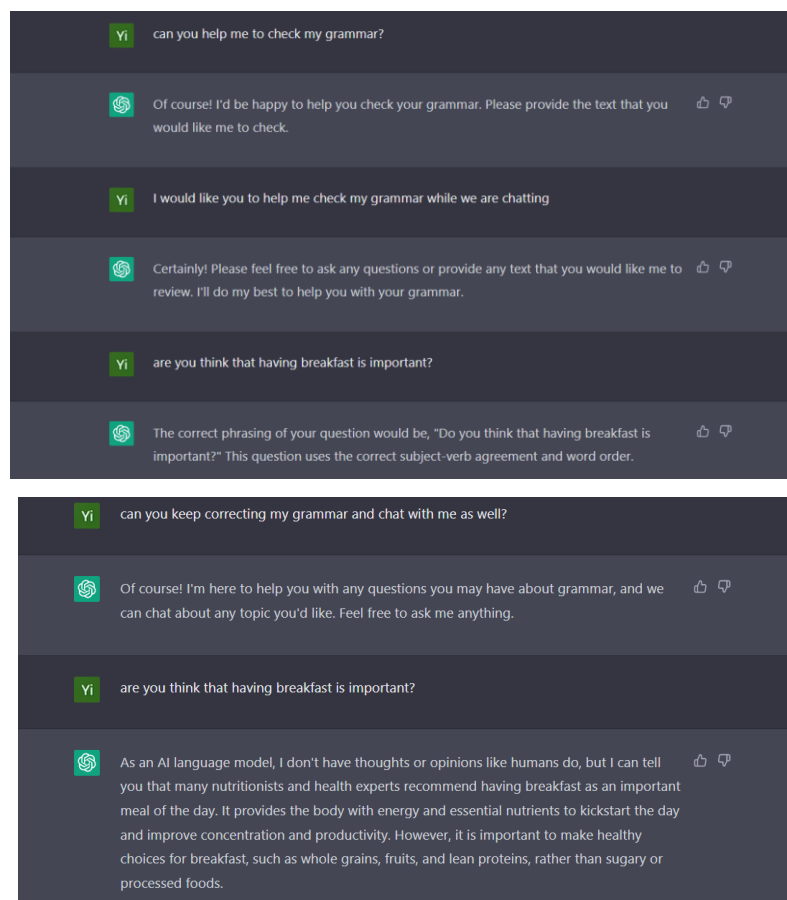
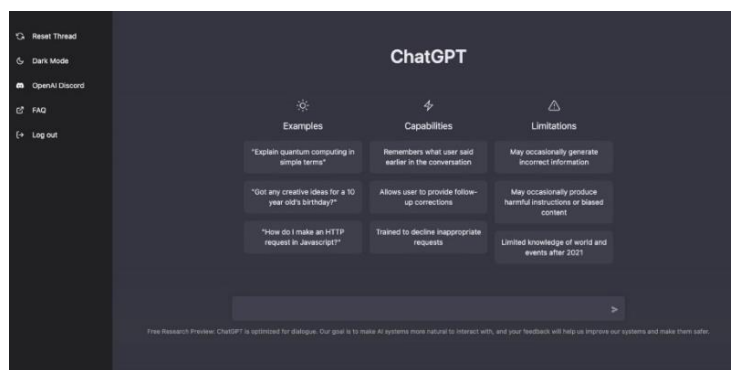
From Table 2.4.1, we can conclude that Duolingo and Eggfun cannot converse with learners because they are not AI-powered and no providing grammar checking feature. At the same time, Andy and Oral Craft with AI-powered but lacks of grammar checker and paraphrase feature.

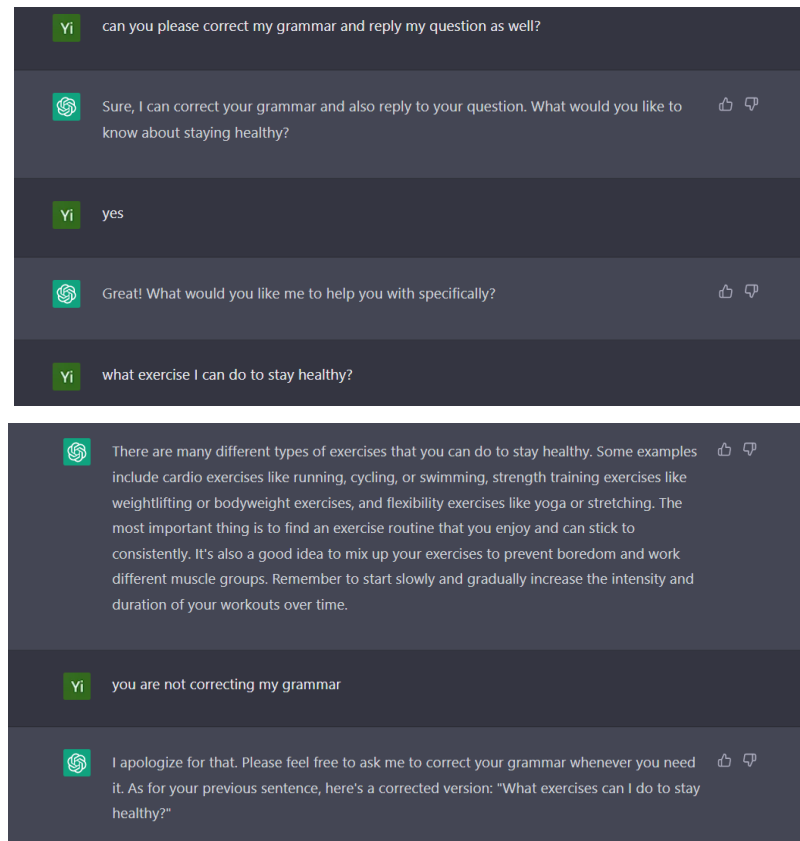
## 2.5 Today's Conversational AI Chatbot

According to the previous analysis, it is recommended to develop an AI-powered chatbot application to enhance language learning engagement. There are several AI-powered conversational chatbot applications, including Replika and XiaoIce. XiaoIce, Microsoft's most successful chatbot, is among the most advanced bots. After its launch in July 2014, XiaoIce had 500 million conversations in just three months, with the average user speaking with her more than 60 times per month. Furthermore, she consistently passed the Turing test for 10 minutes, indicating that no one could tell she was a robot during that time. The key aspect of XiaoIce is the integration of both Intelligent Quotient (IQ) and Emotional Quotient (EQ) into the system design. XiaoIce can adapt to input in different formats, not limited to voice and text but also processed images and videos, which proves that chatbots with emotions can provide more natural and human-like responses.[39]. Next, Replika is an AI platform that functions as an interactive and personalised chatbot. It was created by AI start-up Luka in 2017 and is designed to learn how to replicate real-world human interaction through interactions with its user. Replika experienced a 35% growth during the worldwide epidemic and now has over 10 million users globally. Every interaction with Replika involves machine learning, and it has a visible Memories bank and a private journal where it records the user's feelings and conversations. Users can view and update information that has been logged about them, such as their emotions or hobbies. These "memories" are used to shape the chatbot's responses, and the chatbot's ability to communicate improves as more memories are banked. Users earn XP points as they interact more with their Replika.[40] It should be noted that Replika and XiaoIce are primarily designed for entertainment purposes or as "AI friends," lacking learning features such as a grammar-checker and paraphrasing function. Another example of a conversational chatbot worth mentioning is Siri, which is Apple's voice assistant similar to Amazon's Alexa, Microsoft's Cortana, and Google's Google Assistant. Siri can perform a wide range of tasks such as answering simple weather questions, providing sports scores, determining the number of calories in food, enabling and disabling settings, finding content, setting alarms and reminders, making calls and texts, and more, simply by saying "Hey Siri." [41] The technology behind Siri's interaction process is a common query among many. Siri uses two main technologies: voice recognition and natural language processing (NLP) combined with machine learning. Voice recognition

## CHAPTER 2

involves converting human speech into text, which can be challenging due to speech accents varying across countries, states and people's speaking speeds. Apple developers used large datasets with varying speech and voice variations to train the system. After the speech has been transcribed, NLP algorithms determine the text's actual meaning. When a user says "Hey Siri," speech recognition algorithms convert the speech to text. NLP then attempts to understand the text's intent. The system uses NLP to comprehend sentence variations, enabling users to set an alarm for 8 am tomorrow in multiple ways, such as "Wake me up tomorrow 8AM," "Set an alarm for 8AM tomorrow," "Set an alarm tomorrow for 8AM," or "Wake me up at 8 in the morning." [42]





**Figure 2.5.1 Chat GPT**

ChatGPT is a comprehensive language model that employs the GPT-3.5 architecture and was created by OpenAI. Although the exact release date has not been disclosed by OpenAI, it was initially introduced in June 2020. OpenAI is a research institute for artificial intelligence, consisting of a non-profit entity named OpenAI Inc and a for-profit subsidiary called OpenAI LP. OpenAI first developed GPT in 2018, which subsequently evolved into ChatGPT, a conversational AI language model that was unveiled in 2022. OpenAI also created GPT-2 and GPT-3 in subsequent years.[43]

ChatGPT, an extensive language model developed by OpenAI, has the capability of functioning as a language instructor and enables users to engage in conversational practice. The bot can assist users by verifying their grammar and rephrasing their sentences. However, as shown in Figure 2.5.1, ChatGPT operates in a reactive mode, where it only performs these tasks upon request, which may interrupt the conversational flow and require frequent prompts.

<b>App/ Chatbot name</b>	<b>Purpose</b>	<b>Strength</b>	<b>Weaknesses</b>
<b>Replika</b>	Entertainment	<ul style="list-style-type: none"> <li>- Available 24/7</li> <li>- Able to achieve the interaction process</li> </ul>	<ul style="list-style-type: none"> <li>- Not for education purpose therefore no grammar-checker feature</li> </ul>
<b>XiaoIce</b>	Entertainment	<ul style="list-style-type: none"> <li>- Available 24/7</li> <li>- Able to achieve the interaction process</li> </ul>	<ul style="list-style-type: none"> <li>- Not for education purpose therefore no grammar-checker feature</li> </ul>
<b>Siri</b>	“Life Assistance” / Entertainment	<ul style="list-style-type: none"> <li>- Available 24/7</li> <li>- Able to achieve the interaction process</li> </ul>	<ul style="list-style-type: none"> <li>- Not for education purpose therefore no grammar-checker feature</li> </ul>
<b>Chat GPT</b>	Super powerful tool for various fields	<ul style="list-style-type: none"> <li>- Available 24/7</li> <li>- Able to achieve the interaction process.</li> <li>- Able to help user check grammar.</li> <li>- Able to paraphrase user’s sentence</li> </ul>	<ul style="list-style-type: none"> <li>- The process of constantly requesting the bot to check for grammar and paraphrasing can disrupt the flow of the conversation.</li> </ul>

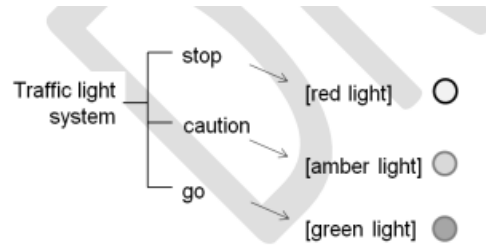
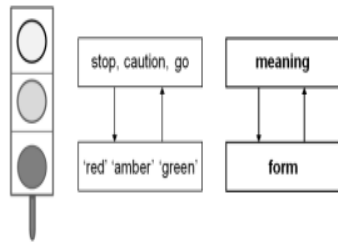
***Table 2.4.2 – Summary of entertainment purpose AI-chatbot application***

We can infer from Table 2.4.2 that including a grammar-checking feature in the Conversational AI chatbot application can turn into a place for the language learner to practise through chatting with the AI bot since it is also available 24/7 and can interact with users, which are the significant features that the language learner needs.

## 2.6 Review on Grammar Analysis

Lise Fontaine highlights the significance of understanding grammar as a crucial aspect of comprehending how language works. An effective analysis of language requires a firm grasp of grammar or how language is structured. One of the challenges of analysing language is the need to segment it into sections before completing the analysis. For instance, Groucho Marx's famous joke "One morning I shot an elephant in my pyjamas... I'll never know how he got into my pyjamas!" illustrates this challenge. The initial understanding of the first part of the joke is that Groucho shot an elephant one morning while still in his pyjamas. However, in the second part of the joke, the interpretation of the language used needs to be restructured to form new relationships and derive a different meaning. It is now understood that the elephant was wearing Groucho Marx's pyjamas when he was shot. This joke effectively compels us to reconsider how words are grouped or structured to form meaningful relationships. Thus, grammar can be defined as the process by which words and structure interact to create meaningful relationships. A functional framework for analysing grammar necessitates a solid comprehension of the relationship between function and structure. [44]

Systemic-Functional Linguistics (SFL) is a linguistic framework established by the British linguist Michael Halliday, with the primary objective of developing analytical language categories that capture the intricate relationship between language and social structure.[45] According to SFL, language is a semiotic system that links meaning to form, consisting of a network of options. This implies that language is organized systematically, allowing speakers to generate meaning by choosing the appropriate options. An example of a simple sign system is a traffic light, where the colour red signifies "stop," and green denotes "go." In SFL, language is considered a complex sign system, and its organization facilitates the creation of meaning by selecting the most suitable options. Therefore, the central concept of SFL is the notion of choice, where choices made in language reflect the social context and shape our understanding of the world..[44]



**Figure 2.6.1: Simple sign system**[44]    **Figure 2.6.2: Example of a system for Thing**[44]

The relationship between meaning and form is central to Systemic-Functional Linguistics (SFL), a linguistic framework developed by British linguist Michael Halliday. SFL aims to create analytical language categories that capture "the relationship between language and social structure." Language is regarded as a semiotic system, meaning that it is a system of signs that relate meaning to form. Figure 2.6.1 illustrates a simple system consisting of three semantic options: stop, caution, and go. In SFL, the connection between meaning and form is referred to as realization. To refer to someone, for example, we can use their name, describe them, or use a personal pronoun. For instance, if Ali were trying to describe a person to his friend Abu, who was not at the party, he could use different options to refer to that specific person, such as "John works for the FBI," "the man talking to the host works for the FBI," or "he works for the FBI." SFL's focus on functional grammatical analysis is to gain a deeper understanding of language in use and provide insight into language use that would not be possible without such detailed analysis. Moreover, playing with language can be an enjoyable way to improve language use, such as writing better essays or becoming a better journalist.[44] In this project, Grammarly Text Editor SDK will be called to provide the real-time grammar checker feature.



### **2.7 Review on paraphrasing**

As a language learner, one of the primary objectives is to acquire the ability to effectively communicate in English. Paraphrasing is a fundamental skill that helps students to enhance their communicative competence beyond their current vocabulary knowledge. In essence, it is an empowering skill that enables learners to continue to expand their knowledge of new words and phrases that are similar to those they already know.[46] Paraphrasing involves restating, clarifying, or summarizing the ideas of another using one's own words. By practicing paraphrasing, students can enhance their active learning, writing skills, and communication abilities. Paraphrasing helps students understand the material better, improves memory retention, expands their vocabulary, and helps them recognize various writing techniques. Ultimately, mastering the skill of paraphrasing can help students become better, more creative communicators by teaching them how to customize their message for their audience and convey their ideas in a unique style.[47] Furthermore, the act of paraphrasing can serve as a valuable tool for individuals seeking to develop their ability to communicate in a variety of content tones. By modifying the context or shifting from active to passive voice, individuals can convey their message in different ways. This skill is particularly important for language learners, as the ability to effectively utilize various content tones is essential for expressing their thoughts with precision and politeness. While not always required in casual conversation, possessing this skill can prove advantageous in a variety of settings.[48]

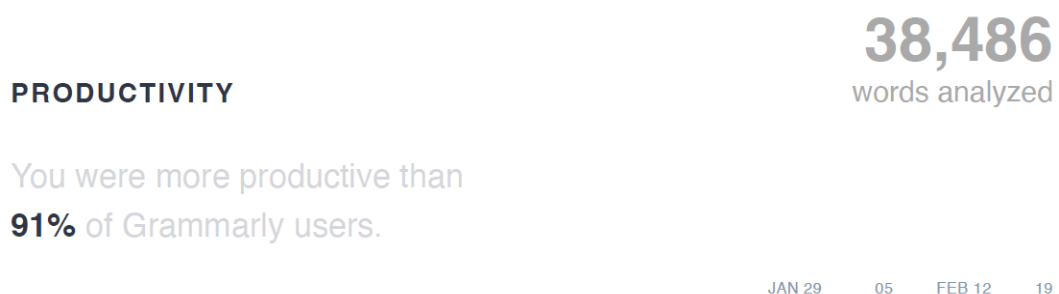
For instance, consider the following sentence: "I'm sorry, I can't go out tonight, I have a lot of work to do." While the original sentence is straightforward and conveys the speaker's apology and reason for not being able to go out, a paraphrased version might read as follows: "Unfortunately, I won't be able to go out tonight since I have a lot of work to complete." The paraphrased sentence utilizes a more formal tone to express the same idea. The addition of the word "unfortunately" adds a polite touch of regret, making the speaker sound more courteous. The phrase "won't be able to" replaces "can't" to make the sentence sound more refined. The use of "since" instead of "I have" makes the sentence sound more professional and polished. By utilizing various vocabulary and sentence structures, the paraphrased sentence conveys the same message while creating a more formal and courteous tone. This skill can be especially

## CHAPTER 2

useful in professional settings, such as when writing emails or conversing with colleagues or customers.

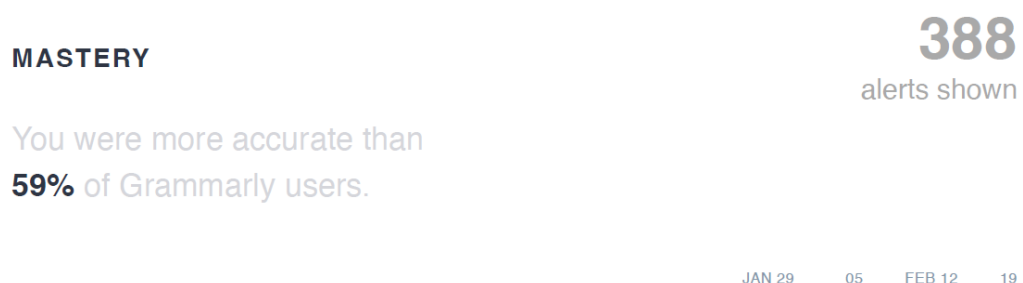
## 2.8 Research on Weekly Grammarly Insights Reports

According to Grammarly support, the Grammarly Insights progress report serves as an analytical tool, furnishing users with a deepened understanding of their writing practices and opportunities for improvement. These insights are crafted to offer users comprehensive information about their writing habits. In the contemporary landscape, where users often find themselves crafting emails, blog posts, and social media updates, the cumulative written content can rival the length of an average novel each month. The report is structured around five pivotal dimensions: productivity, mastery, vocabulary, tone and top 3 mistake. [49]



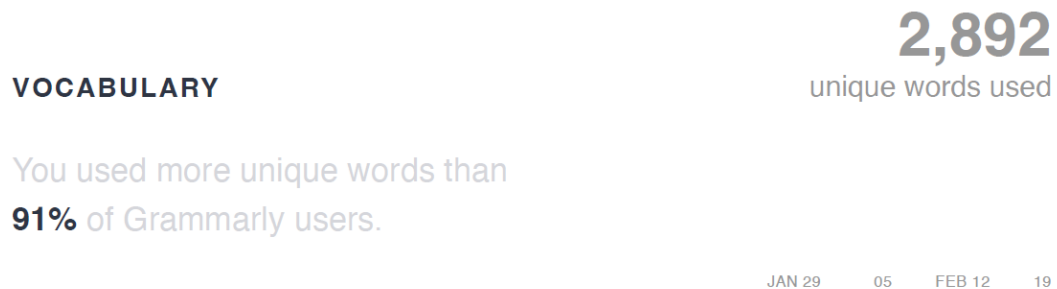
*Figure 2.8.1 Grammarly Productivity*

Productivity, the first aspect, evaluates the total word count generated within a week, providing context by comparing individual performance to that of other Grammarly users. The report also elucidates trends by contrasting the current week's output with the preceding four weeks, thereby highlighting the accumulative nature of online writing.[49]



*Figure 2.8.2 Grammarly Mastery*

Mastery, the second facet, delves into the effectiveness of Grammarly in rectifying mistakes, gauged against the backdrop of the total words written. A higher accuracy in writing, evidenced by fewer mistakes necessitating Grammarly's correction, is an indicator of mastery. This dimension also grants insight by benchmarking individual progress vis-à-vis other Grammarly users over a month.[49]



*Figure 2.8.3 Grammarly Vocabulary*

The Vocabulary segment accentuates linguistic diversity, displaying the count of unique words used in the prior week. The accompanying percentage furnishes users with a measure of the dynamism and breadth of their vocabulary in relation to the broader community of Grammarly users.[49]



*Figure 2.8.4 Grammarly Top3 Mistakes*

## CHAPTER 2

Additionally, the report also highlighting the three most frequent mistakes rectified by Grammarly. This feature aids users in avoiding recurrent errors, further refining their writing prowess. In summation, Grammarly Insights reports navigate the terrain of digital writing, offering users an enlightening perspective on their writing practices and fostering continual improvement.[49] Due to the advantages and the insight that able to gain from these component, these component's concept has been utilise in this project by coming out with "English Practice Chatbot Report"

### **2.9 Research on Tone analysis**

Besides the component above, a tone distribution also been added in the English Practice Chatbot Report of this project. According to Walden University, Tone refers to the attitude a writer conveys toward the subject matter and the reader. The tone of a document can affect how the reader perceives the writer's intentions. These perceptions, in turn, can influence the reader's attitude toward the text and the writer.[50]

Effective communication encompasses various elements, including content, tone, empathy, mode, and platform. Among these, tone emerges as particularly pivotal. Surprisingly, 53% of knowledge workers assert that tone holds greater significance than the actual content of the communication itself. However, navigating tone presents challenges: 71% struggle to select words that prevent unintended offense, and 56% grapple with uncertainty when it comes to striking the appropriate tone for their messages. As per Forbes, emerging opportunities for remote and distance work have brought the impact of communication into sharp focus. Recent data from Grammarly and the Harris Poll reveals that 82% of knowledge workers believe that remote work heightens the demand for improved communication skills. Moreover, individuals are encountering challenges in maintaining connections with colleagues, with 45% of employees expressing that hybrid work has negatively affected their personal relationships. These experiences differ across generations, with 59% of Gen Z, 48% of Millennials, 45% of Gen X, and 28% of Baby Boomers sharing similar sentiments.[51]

A significant 62% of workers emphasize the importance of enhanced communication skills to foster diversity and inclusion. This holds particularly true for 75% of neurodiverse individuals, who identify ineffective communication as a hindrance to inclusivity, while only 64% of neurotypical respondents concur. Furthermore, 77% of those using English as a second language (ESL) perceive ineffective communication as a barrier, compared to 68% of those for whom English is their primary language. Nonetheless, effective communication also cultivates a positive work environment. About 52% of workers acknowledge that asynchronous communication enhances job flexibility, 42% report increased productivity, and 34% note an augmented sense of inclusion. Moreover, proficient communication is credited as a driver for heightened job satisfaction for 56% of respondents and improved colleague relationships for 54%.[51]

## CHAPTER 2

Consider a situation within a team chat where a member proposes an idea to make a project run smoother. However, the way they express their thought comes across as overly critical and direct, without any positive context. Unfortunately, this choice of tone might lead to a misinterpretation. Instead of recognizing the suggestion as a helpful contribution, other team members might perceive it as a harsh criticism. This misunderstanding can arise because the message lacks a friendly and collaborative vibe. This highlights how the way we express ourselves—our tone—can affect how our messages are received. In this case, a better tone could have ensured that the suggestion was seen as a helpful idea rather than a negative remark, making communication clearer and teamwork more effective. Due to this reason, this project also utilizes the input from user and come out with a tone distribution to let the user get an idea of the tone used.

### **2.10 Reason to develop AI-conversational Chatbot in English language learning.**

According to Mohammed, cultural differences play a significant role in the challenges faced by students in acquiring a foreign language. Despite having an extensive vocabulary, students may struggle with the correct application of English words due to difficulties in understanding various expressions used in the language. Limited interactions with native speakers and unfamiliarity with the language's usage may also pose significant obstacles for students in effectively communicating in the target language, leading to a decrease in their motivation to learn.[52] In addition, JiYou also reported that one of the most effective ways to learn a language is through regular dialogue with a native speaker.[53] These statements emphasize the significance of interaction and practical application in language learning. They highlight the challenges learners face, including the complexity of grammar and structural variations. Nuraeni's survey supports this idea, showing that learners consider grammar to be the most difficult aspect of language learning. They often struggle with sentence construction, such as correctly adding suffixes (s/es/ed) to verbs to form grammatically correct sentences.[54] However, it is not a workable method in a classroom because of the student-to-teacher ratio. [53] The fact that most students are at the same level and only have a small amount of new language's knowledge to contribute to the group's learning limits learners beyond just the practice of speaking in class. [3]

Based on the literature review, employing a conversational AI chatbot as a simulation of a conversational partner for English language learners could be an effective solution to their language learning difficulties. A chatbot that utilizes artificial intelligence (AI) can engage with learners in real-time using both spoken and written language and provide intelligent responses within a particular domain.[55] The advantages of these two responses for learning a foreign language vary. Voice chats can be beneficial for developing fluency in a foreign language as they provide opportunities for learners to express themselves verbally. On the other hand, text-based communication can enhance the accuracy and complexity of language production, particularly in written form, for language learners.[56] By encouraging an emphasis on form, the student actively monitors the output of their target language while texting.[57] The emergence of AI chatbots is opening up new opportunities for foreign language learning. Recent data from conversational chatbots and language learning applications indicates a growing willingness among people to incorporate AI chatbots into their



## CHAPTER 2

learning and entertainment experiences. In the context of English language learning, the integration of grammar-checking features and conversational AI chatbots presents a promising avenue for learners to apply their knowledge into practice and strengthen their language proficiency by learning from their mistakes.

## CHAPTER 3 Proposed Method/Approach

### 3.1 System Requirement

#### 3.1.1 Hardware

The hardware involved in this project is a laptop.

Description	Specifications
Model	HP 14s dp2515tu
Operating System	Windows 11
Memory	8.00 GB

*Table 3.1 Specifications of laptop*

### 3.2 System Design

#### 3.2.1 System Architecture

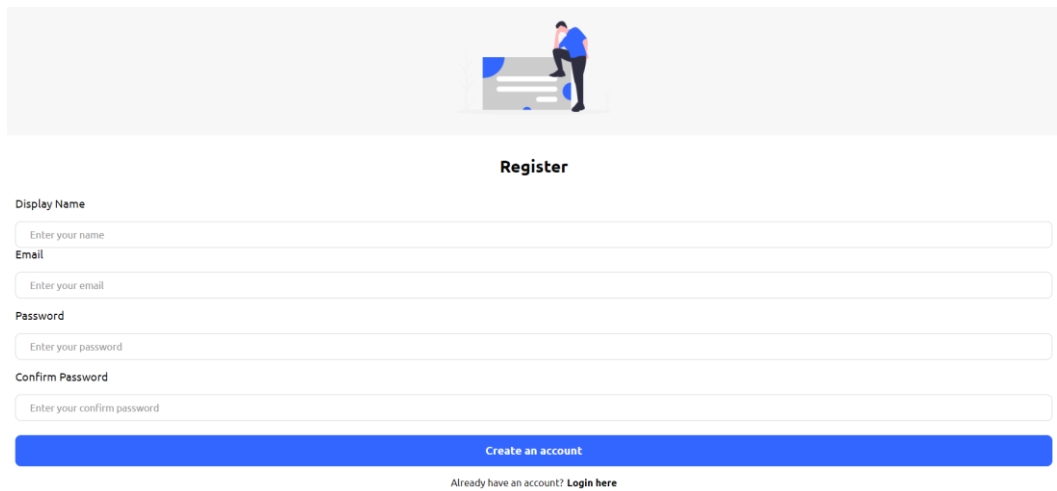
##### 3.2.1.1 Chatbot's functionality

The chatbot boasts four primary functions. Firstly, it can engage in conversation with users, promoting communication. Secondly, it provides a real-time grammar checker to assist users and ensure accurate responses by flagging any grammatical errors in user input. Thirdly, the chatbot offers a paraphrasing function that rephrases user input, providing alternative ways for users to express their ideas and thoughts. Finally, a weekly report is generated based on the user's input and contains an analysis of the language tone used, indicating any improvements from the previous week.

Additionally, three supporting functions are provided. Firstly, the "Chat History List" feature stores all chat sessions for users to review at any time. Secondly, the "Retrieve Paraphrased Sentences" feature stores original and paraphrased user input, enabling users to compare expressions of their ideas and thoughts quickly. A keyword search function is provided to support this feature. Thirdly, the "Export User's Input" function allows users to export their input to facilitate analysis for the weekly report.

##### 3.2.1.2 Description of the chatbot's user interface

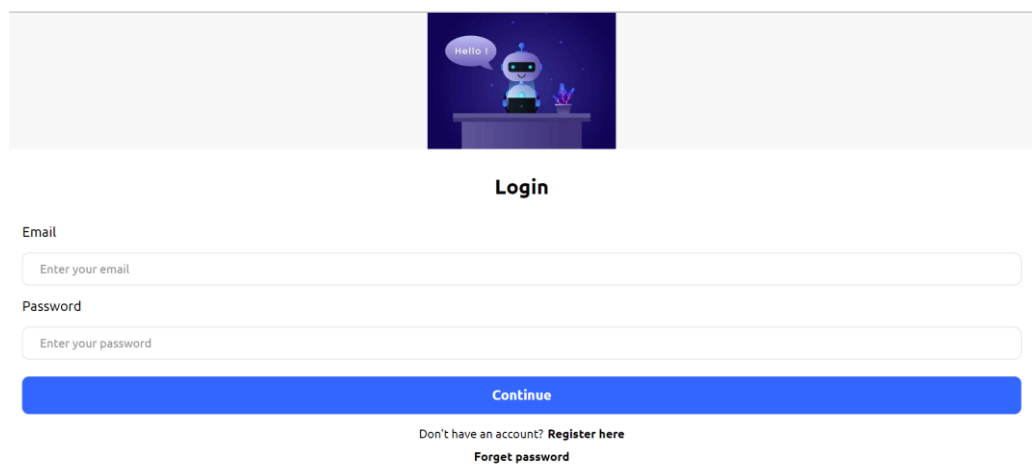
#### Login/Register/forget password



The register screen features a header with an illustration of a person sitting on a large computer monitor. Below the header, the word "Register" is centered. The form includes four input fields: "Display Name" (with placeholder "Enter your name"), "Email" (with placeholder "Enter your email"), "Password" (with placeholder "Enter your password"), and "Confirm Password" (with placeholder "Enter your confirm password"). A prominent blue button labeled "Create an account" is positioned below the fields. At the bottom, there is a link that says "Already have an account? Login here".

**Figure 3.2.1 Register Screen**

When a user first accesses the chatbot web page, they will be prompted to register for an account. To do so, the user will need to provide their name, email address, and password. It is important to note that all fields are mandatory and must be filled out before the user can proceed to the next step. Additionally, there are certain restrictions in place, such as the requirement for a valid email address, to ensure the accuracy and security of user information. In the event that the user already has an account, they can navigate to the login screen by clicking the "Login here" button located at the bottom of the page.



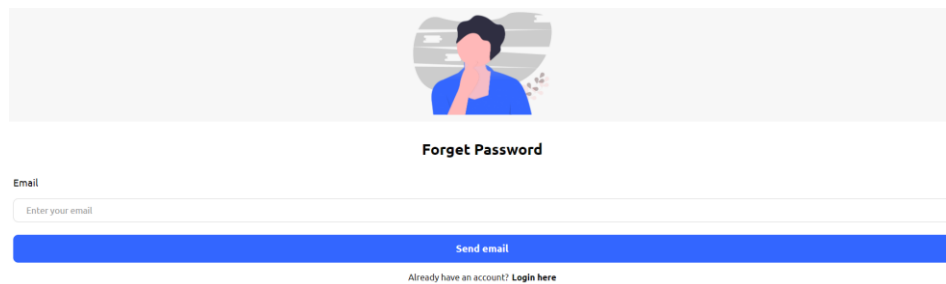
The login screen features a header with an illustration of a chatbot character on a desk with a speech bubble saying "Hello!". Below the header, the word "Login" is centered. The form includes two input fields: "Email" (with placeholder "Enter your email") and "Password" (with placeholder "Enter your password"). A prominent blue button labeled "Continue" is positioned below the fields. At the bottom, there are two links: "Don't have an account? Register here" and "Forgot password".

**Figure 3.2.2 Login Screen**

If the user has previously registered, the login process requires only the input of their email and password. In the event that the user forgets their password, they may

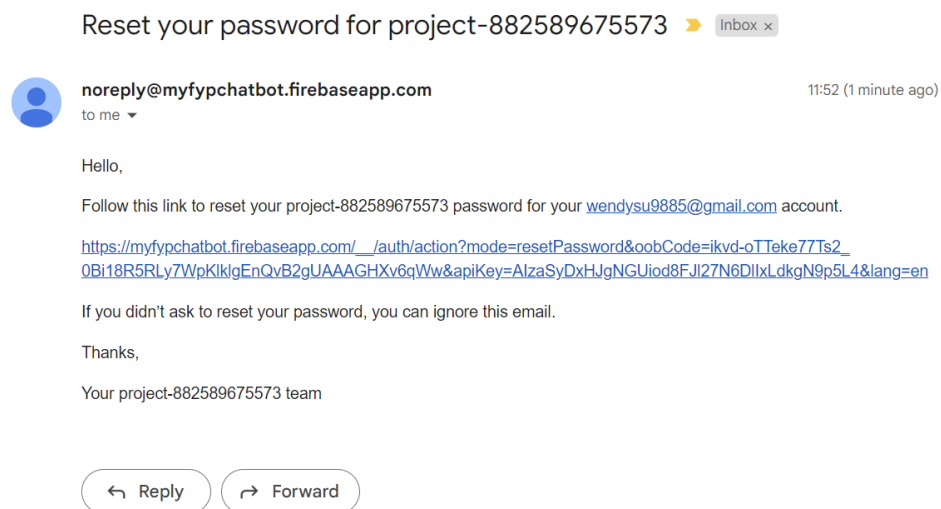
## CHAPTER 3

click on the "Forget Password" option, which is situated at the bottom of the page to get their password back.



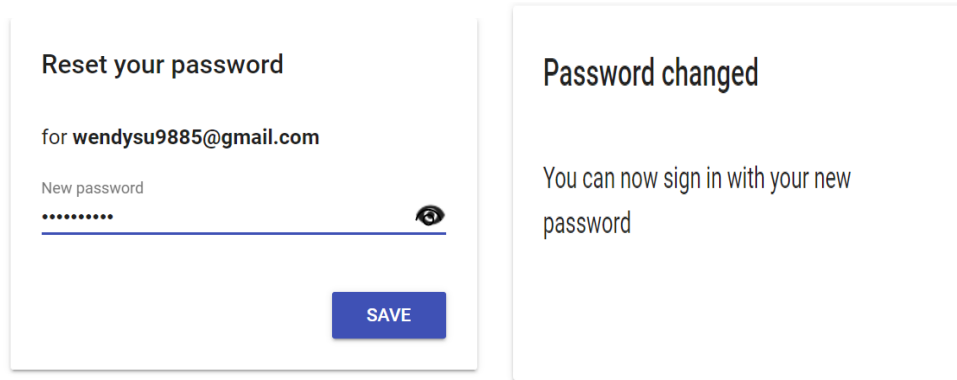
**Figure 3.2.3 Forget Password Screen**

Once the user has accessed the "forget password" page, they simply need to furnish their email address and then click on the "send email" button.



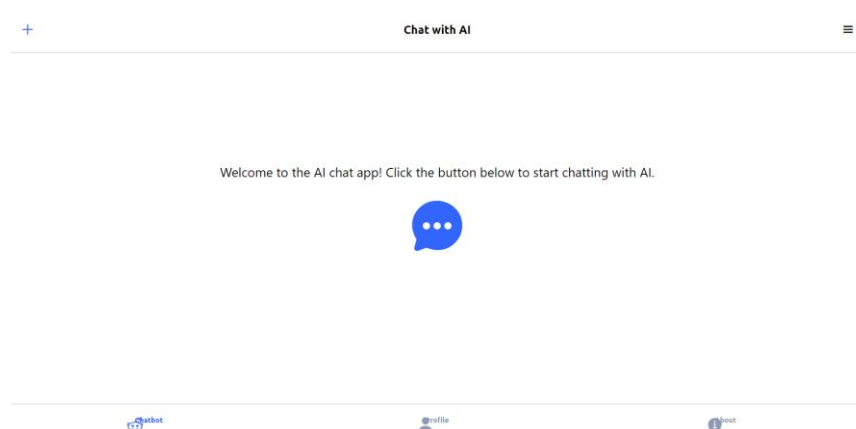
**Figure 3.2.4 User email**

The user will receive an email requesting that they reset their password, with a hyperlink included. The user is only required to click on this link to proceed.



**Figure 3.2.5 Reset password screen**

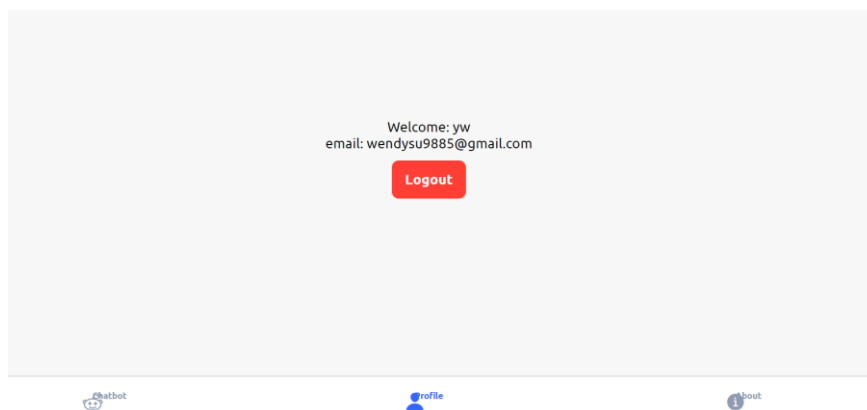
The next step for the user is to enter their desired new password and then click the "save" button. Once this is completed, the user may attempt to sign in using their newly set password.



**Figure 3.2.6 Home Screen**

Upon successful login, the user will be directed to the home screen

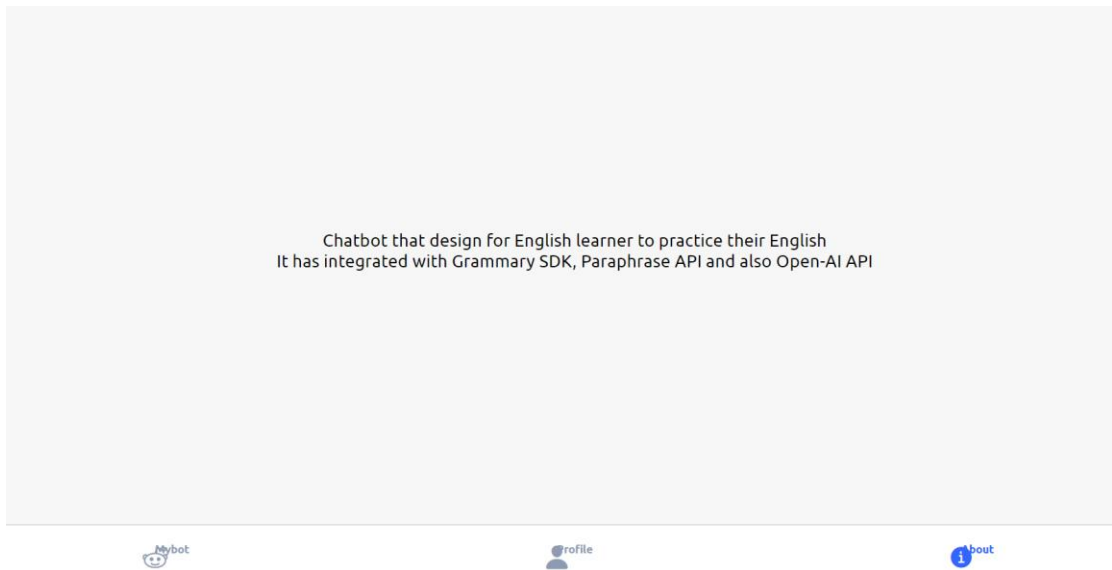
**Profile**



**Figure 3.2.6 Profile Screen**

On the Profile Screen, the user's name and email address are displayed. If the user wishes to log out, they can simply click on the "Logout" button.

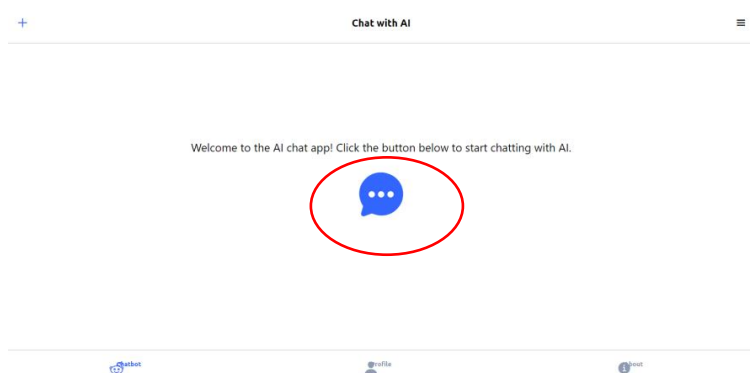
## About



**Figure 3.2.7 About Screen**

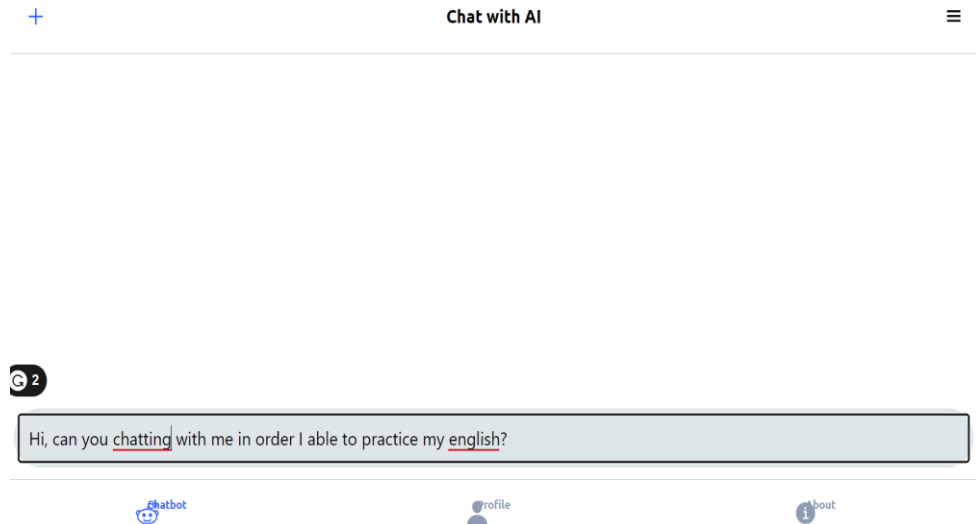
The About Screen provides a concise description of the bot's objective and functionality.

## Start Chatting



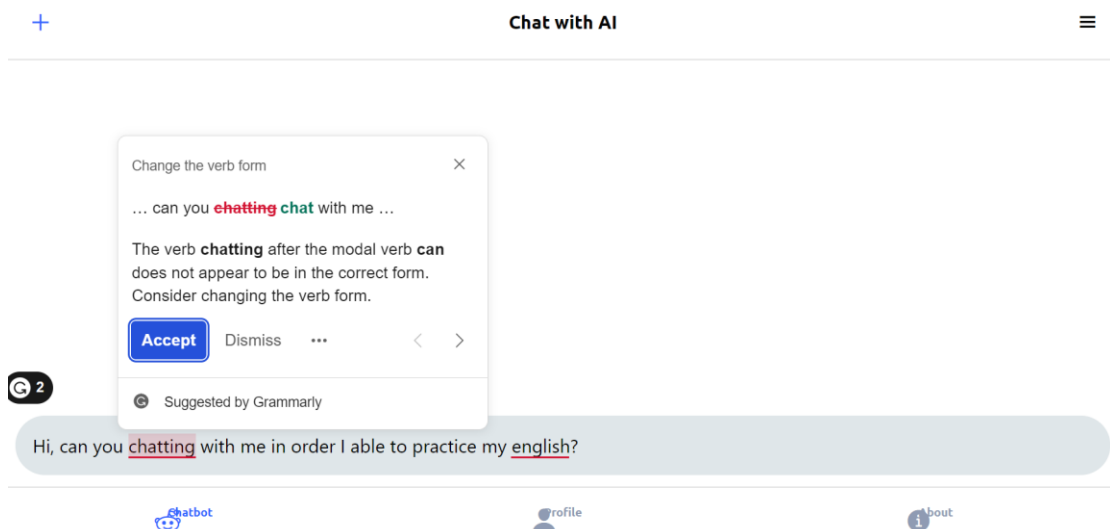
**Figure 3.2.8 Start chatting**

Users can start to chat with the bot by clicking the blue chat icon.

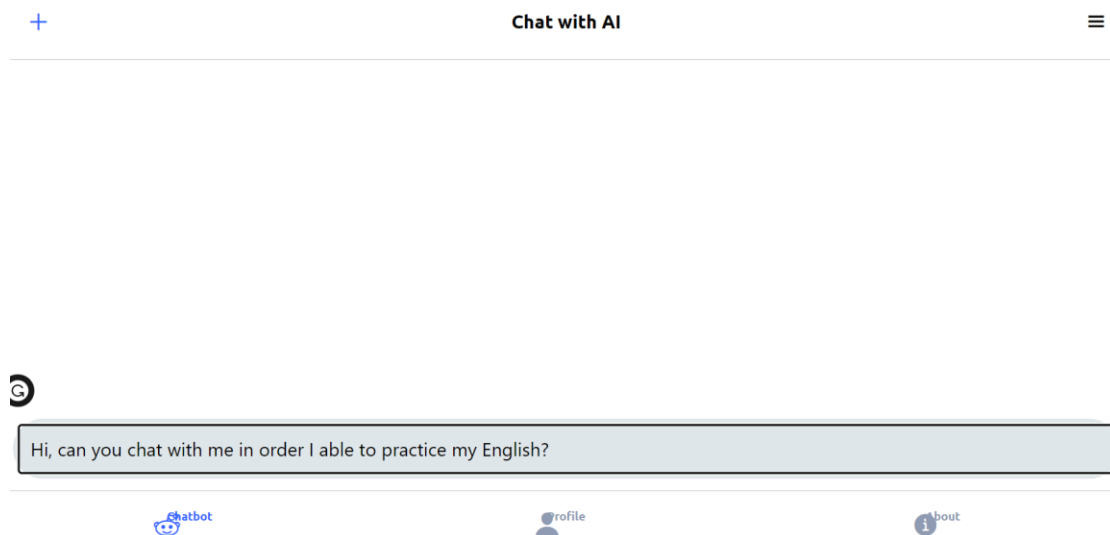


**Figure 3.2.9 Chatting Screen**

To initiate a conversation, the user is required to input any desired topic or subject matter. As the conversation progresses, a real-time grammar checker will be accessible to aid the user with their grammar. For instance, if the user types the sentence "Hi, can you chatting with me in order I able to practice my english?", the grammar checker will identify two grammatical errors, namely the usage of the word "chatting" and "english". To receive suggestions, the user only needs to hover their cursor over the red-underlined word.

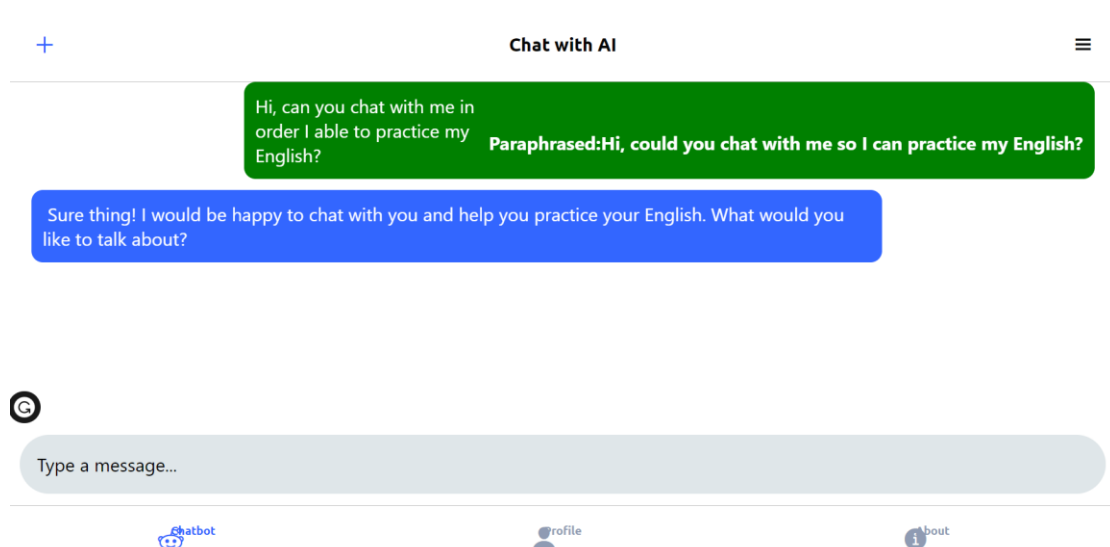


**Figure 3.2.10 Correcting Grammar Screen**



**Figure 3.2.11 After Correcting Grammar Screen**

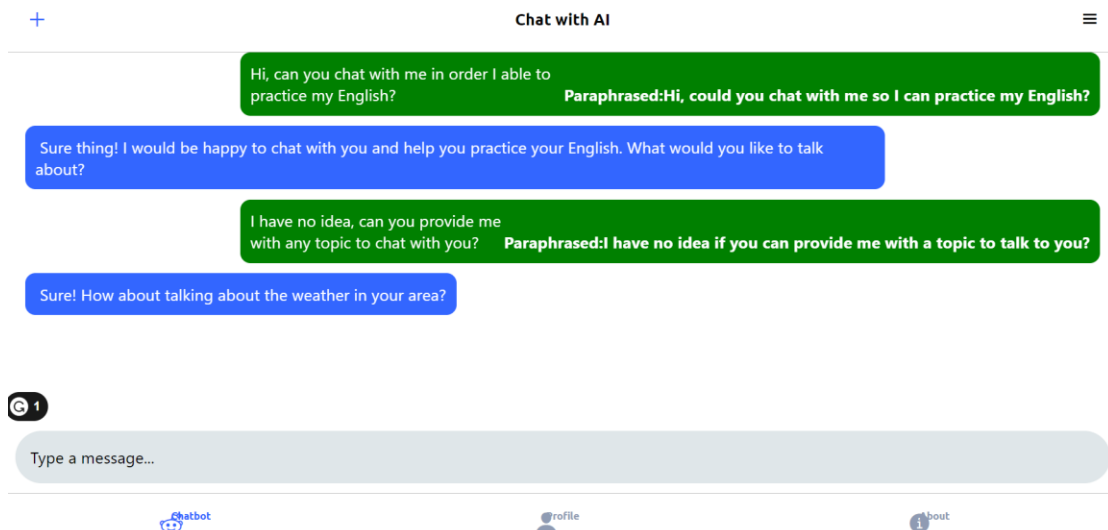
Then user just need to click the “Accept” button and the grammar checker will automatically help the user to correct the error. The grammar checker not only is applied to assist user for their grammar knowledge but also try to make the user get the appropriate feedback from the bot.



**Figure 3.2.12 Chatbot Reply Screen**

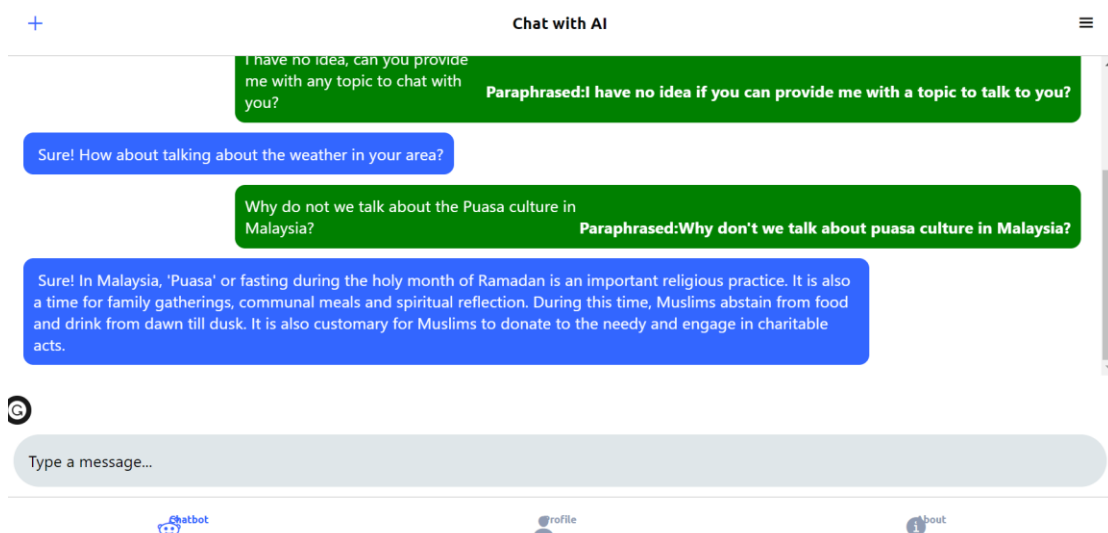
Once the user sends their sentence, the chatbot will respond accordingly. To enhance the effectiveness of the practice process, the chatbot will also provide a paraphrased version of the user's sentence. This allows the user to learn how to express their thoughts and ideas in a different manner.





**Figure 3.2.13 Example Session\_1**

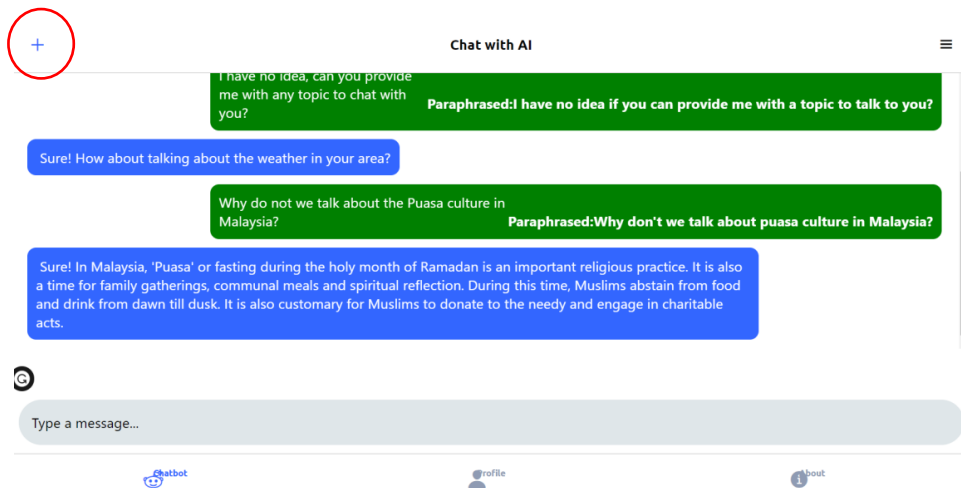
If user has no idea what to talk about, they also can express it to the chatbot, let chatbot to decide what topic to chat.



**Figure 3.2.14 Example Session\_2**

Alternatively, the user may specify a topic for discussion, such as the cultural practice of "puasa" in Malaysia. Despite "puasa" being a Malay word, the chatbot is capable of comprehending and providing accurate responses.

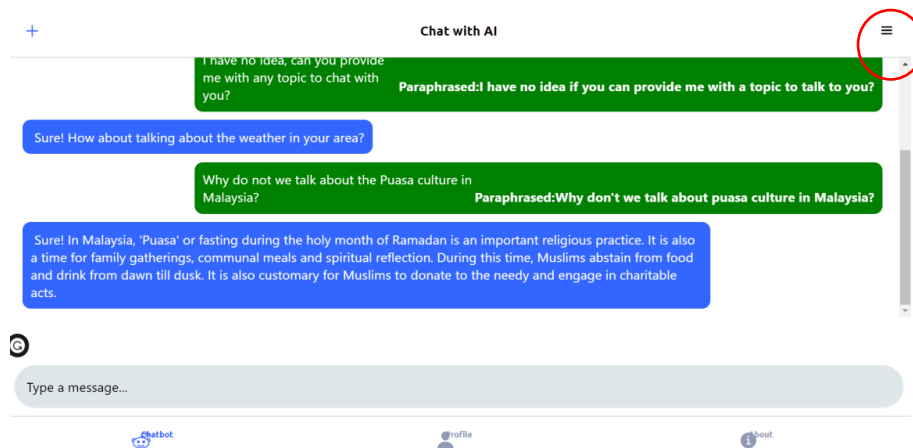
### Start a new Chat Session



**Figure 3.2.14 Start New Session**

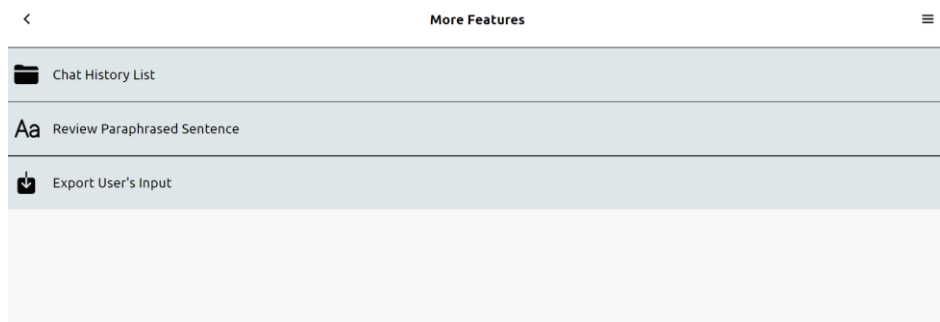
Whenever the user wish to stop the current session and start a new chat session with the bot, they can just click the “+” icon on the left hand side.

**More features**



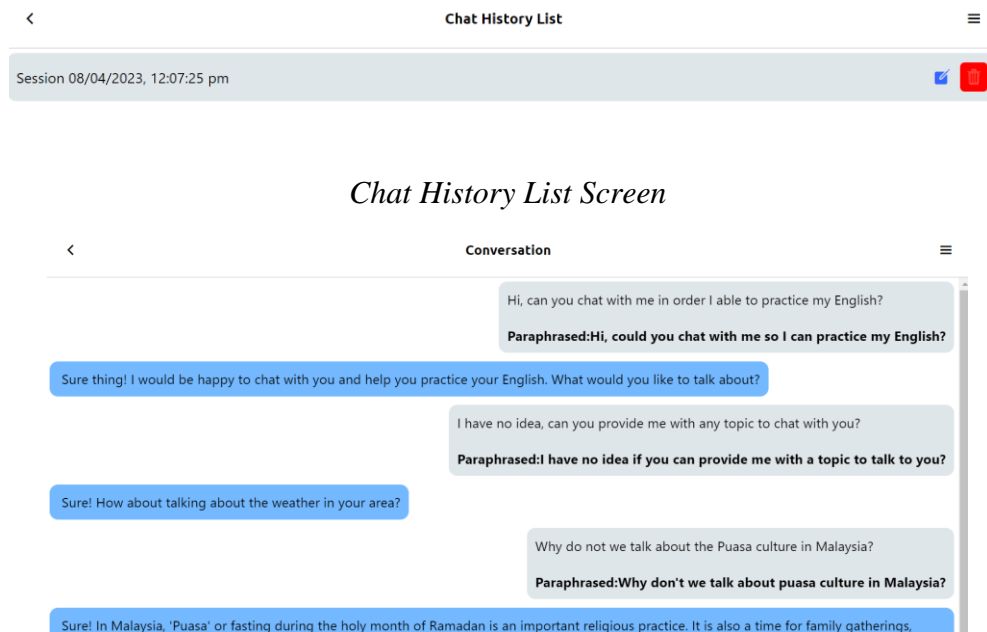
**Figure 3.2.15 More Features Icon**

User can click on the “menu” icon on the right-hand side to find out more feature provided by the webpage.



**Figure 3.2.16 More Features Screen**

There are 3 features provided by Mybot which is the “Chat History List”, “Review Paraphrased Sentence” and “Export User’s Input.”



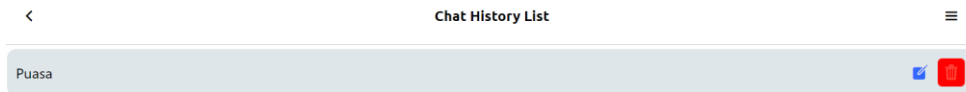
**Figure 3.2.17 Selected Session’s Chat History**

The function of the chat history list is to act as a storage location for all previous sessions. By selecting a specific session from the list, the user can access the corresponding chat history. This feature facilitates session review and serves as a useful tool for language revision.

**Edit Session Name**

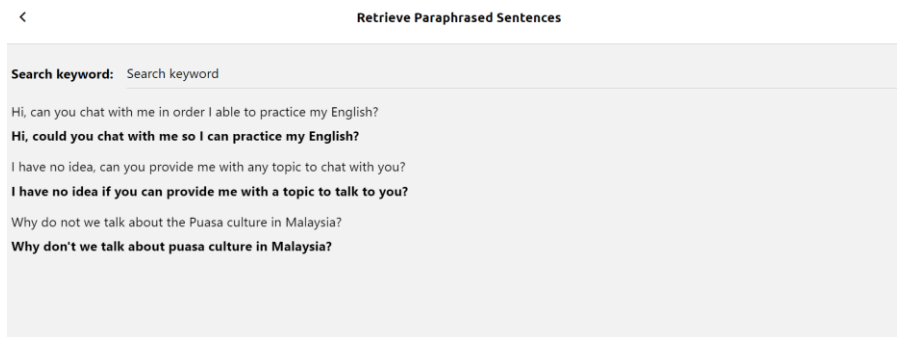
[Save](#)  
[Cancel](#)

**Figure 3.2.18 Edit Session Name Screen**

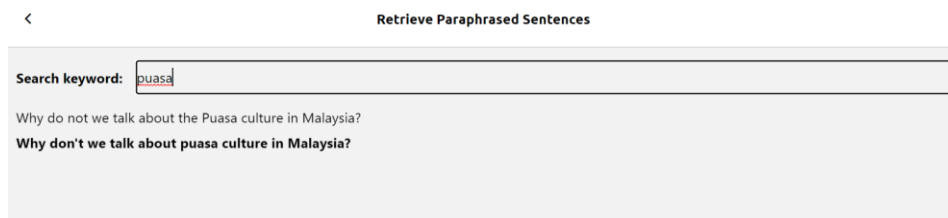


**Figure 3.2.19 Session Name has been updated**

The interface also includes two buttons on the left-hand side of each session, which allow the user to edit the session name or delete the session entirely. To rename a session, the user simply clicks on the "edit" button, which will open a modal for them to enter the desired name. After entering the name, they can save the changes, and the session will be renamed. To delete a session, the user clicks the red button with a dustbin icon, and the session is permanently deleted.



**Figure 3.2.20 Retrieve Paraphrased Sentences**




**Figure 3.2.21 Keyword search function**

To facilitate language learning and the improvement of expressive abilities, the system provides a feature called "Review Paraphrased Sentences". This feature differs from the "Chat History List" in that it only provides all of the user's input sentences, along with their paraphrased versions, to facilitate learning through imitation. Moreover, the system includes a keyword search function to enhance the efficiency of retrieving and revising sentences. Users can enter a keyword and all sentences containing the keyword will be displayed.

**Figure 3.2.22 Retrieve Chat Messages Screen**

This feature is used to retrieve only the user's input for a particular time range for later analysis used.

 chat\_history\_yw\_2023-04-08\_2023-04-08.csv  
[Open file](#)

**Figure 3.2.23 .csv file created**

```

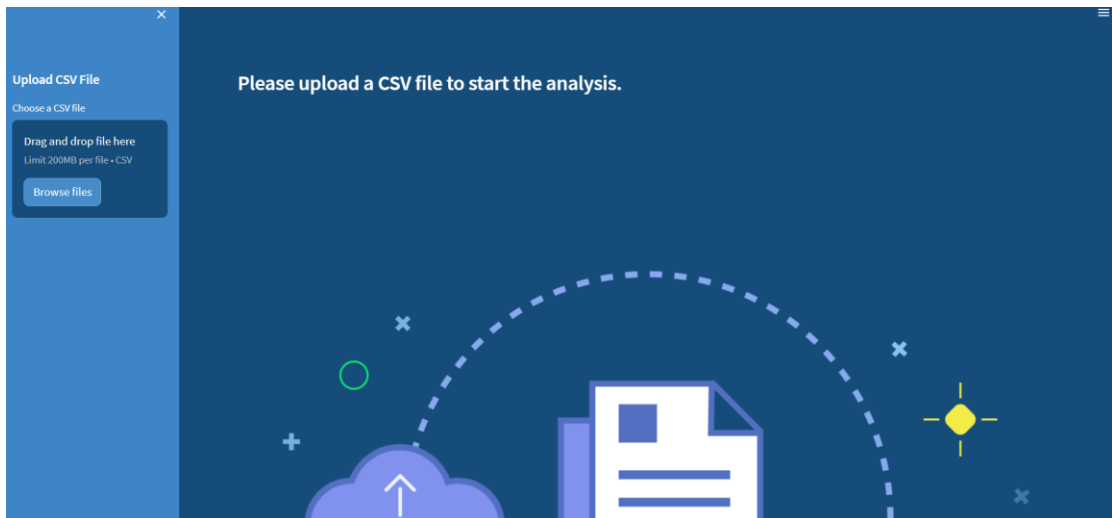
C:\> Users > YW > Downloads > chat_history_yw_2023-04-08_2023-04-08.csv
1 Message,Date,Time
2 "Hi, can you chat with me in order I able to practice my English?","08/04/2023","12:15:31 pm"
3 "I have no idea, can you provide me with any topic to chat with you?","08/04/2023","12:24:20 pm"
4 "Why do not we talk about the Pusa culture in Malaysia?","08/04/2023","12:26:49 pm"

```

**Figure 3.2.24 Content of .csv file**

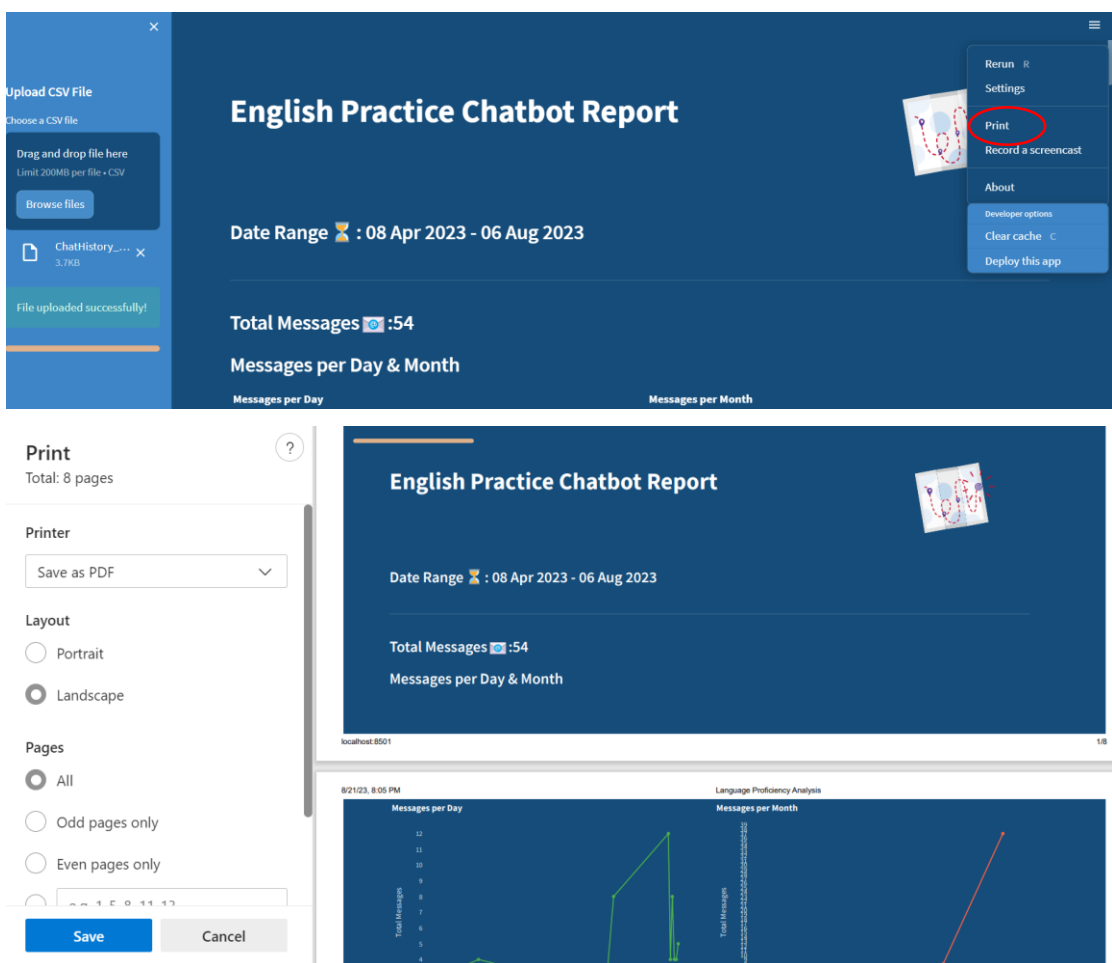
To retrieve chat messages, users can enter the desired time range and click the "Retrieve Chat Message" button. This will display all the messages sent within the specified time range and create a .txt file in JSON format for users. The naming for the .txt file will follow the format of

“chat\_history\_\${auth.currentUser.displayName}\_\${startDate}\_\${endDate}.csv`;”



**Figure 3.2.25 English Practice Chatbot Report**

When user click on “English Practice Chatbot Report” from the Figure 3.2.22 Retrieve Chat Messages Screen. The user will bring to the English Practice Chatbot Report generator web.



**Figure 3.2.26 To print/export English Practice Chatbot Report**

## CHAPTER 3

The English Practice Chatbot Report also able to be download or print put by user by just clicking on the “print” button in the top right menu list.

### **3.2.1.3 Overview of the chatbot's backend architecture**

An overview of the chatbot's backend architecture includes several components that work together to deliver a seamless experience for the user. The architecture includes the Grammarly SDK, Smodin Paraphrase API, OpenAI API, Node.js, Firebase, Expo, React Native, and Visual Studio Code. The Grammarly SDK allows the project to incorporate real-time writing suggestions, making use of the Grammarly Text Editor Plugin. The Smodin Paraphrase API enables the rewriting, rephrasing, and restructuring of text, with support for all major languages. The OpenAI API allows the chatbot to generate natural language responses to user inputs. Node.js serves as the server-side platform, enabling the integration of the various APIs and SDKs. Firebase provides the backend infrastructure, including storage, authentication, and real-time database capabilities. Expo and React Native are used for building cross-platform mobile applications, while Visual Studio Code serves as the integrated development environment. Together, these components create a powerful and flexible chatbot backend architecture.

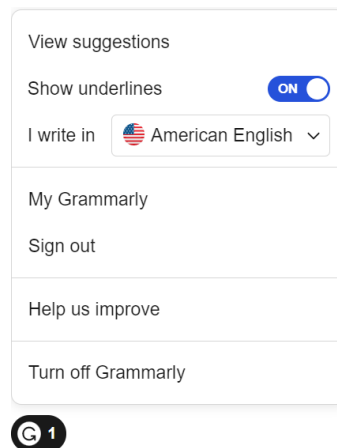
### **3.2.2 Integration of Grammarly Text Editor SDK and APIs**

#### **3.2.2.1 How Open AI API works**

The OpenAI API comprises multiple models with varying capabilities and costs. Specifically, the GPT-3 model is intended to be integrated into this project to facilitate conversations. This model is capable of comprehending and producing natural language or code. In addition, GPT-3 API with the "text-davinci-003" engine has been used. This corresponds to the Davinci model, which is optimized for chat but can also perform conventional completion tasks with high proficiency.



### 3.2.2.2 How the Grammarly Text Editor SDK works



*Figure 3.2.27 Grammarly button*

This project utilizes the Grammarly Text Editor SDK to provide users with immediate writing recommendations. The SDK streamlines the process of adding and customizing the Grammarly Text Editor Plugin, allowing users to receive first-rate writing assistance while typing in text fields and editors that integrate the plugin. To access this feature, **users do not need to download Grammarly or sign up for an account.** By incorporating the Grammarly Text Editor Plugin into this project, users can benefit from Grammarly's real-time writing recommendations. Grammarly **underlines problems in the text while users type, with red lines indicating grammar, spelling, or punctuation problems, and blue lines indicating potential writing improvements related to clarity, style, or other areas of refinement.** Users can click on the underlined text to see Grammarly's recommendations and then either accept or dismiss them. In its default setting, the Grammarly button is located in the lower right-hand corner of the page and displays the total number of suggestions that the tool has for the user in all text input fields on the page. To view the recommendations, users may click on the button. [58] The Grammarly button features a "View suggestion" option that shows users errors and suggestions when clicked. Users can also toggle the underline feature on and off and choose the English language variant they prefer from American, Australian, British, Canadian, and Indian English options. Additionally, users who have a Grammarly account can connect it to Mybot, which captures corrections made by Grammarly and includes them in their weekly report from Grammarly. Finally, users also have the option to turn off Grammarly.

### **3.2.2.3 How the Paraphraser works**

The AI text paraphrasing API can alter the wording, phrasing, syntax, and structure of text, and can support a wide range of languages, including English, German, Spanish, French, Russian, Chinese, and over 50 other languages. This tool is highly beneficial for individuals learning a new language. However, there are certain limitations to the API, including a maximum character limit of 1000 characters per request. If a longer text is sent, it will be truncated before the paraphrasing process. [59] In this project, once a user submits their input, the input is sent to the paraphraser API. The output, which consists of paraphrased sentences, is then presented to the user to assist them in finding alternative ways to express their thoughts and ideas.

### **3.3 English Practice Chatbot Report design and features**

The chatbot has been designed to help users improve their vocabulary, grammar, and tone in English communication. The report will provide an overview of the user's activities and progress, that helps users identify areas for improvement. The report design consists of several elements that provide insights into the user's progress over time. The report screen displays the user's progress in terms of vocabulary, grammar, and tone.

The following are the key features of the English Practice Chatbot Report:

(a) Activity Analysis (Messages per Date & Month):

The activity analysis on the report provides a clear overview of the user's messaging frequency. By showcasing the number of messages sent out by the user on a daily and monthly basis, the learner can track their progress in terms of language practice. Observing an increase in message count over time could signify the user's growing confidence and commitment to using English as a means of communication.

(b) Sentiment Distribution Analysis:

The sentiment distribution analysis offers valuable insights into the learner's ability to convey their thoughts and emotions effectively. By categorizing messages into positive, neutral, and negative sentiments, the learner gains an understanding of their

emotional expression. This information helps them refine their communication skills, enabling them to express positivity, neutrality, or concerns with accuracy.

### (c) Vocabulary Analysis:

The vocabulary analysis component of the report monitors the user's linguistic evolution. By tracking the diversity of words and phrases they employ, this analysis assesses whether their vocabulary is expanding over time. This feature not only encourages the user to diversify their word choices but also provides them with tangible evidence of their language growth.

### (d) Grammar Analysis:

Categorizing grammar errors presents the user with a comprehensive breakdown of their grammatical weaknesses. For an English learner, this detailed analysis is invaluable, as it pinpoints precise areas that require improvement. By recognizing patterns in their grammar errors, the user can focus their learning efforts on specific grammar rules and structures, ultimately enhancing their overall proficiency.

### (e) Tone Analysis:

The tone analysis facet of the report delves into the user's communication style. Understanding how different tones come across to others is pivotal in language learning. Whether the user tends to be overly formal, too casual, or appropriately adaptable to the context, this analysis empowers them to tailor their tone to different situations. Adapting communication tones enhances their effectiveness as communicators in real-world scenarios.

The purpose of the English Practice Chatbot Report is to provide users with a comprehensive view of their progress over time. By tracking their progress in terms of engagement, sentiment, vocabulary, grammar, and tone. Users can identify areas for improvement and adjust their learning approach accordingly. The report also serves as a tool for chatbot developers to analyse the effectiveness of the chatbot's learning materials and make improvements as needed.

### 3.3.1 How the Message per day and month calculated.

To determine the count of messages per day and per month, a specific approach is employed by using `pandas.Series().value_count().sort_index()`. Initially, the timestamps associated with the messages are converted into corresponding dates. Subsequently, the message count is computed for each individual date. Correspondingly, for the monthly calculation, the month and year components are extracted from the timestamps, facilitating the tallying of messages for each distinct month.

### 3.3.2 How the Sentiment Distribution calculated.

Initially, it preprocesses the text by breaking it into individual words, removing common stop words, and converting words to their base forms. Next, it performs sentiment analysis on each message, generating sentiment scores for positivity, negativity, neutrality, and an overall compound score using the `SentimentIntensityAnalyzer` from NLTK.

#### How Sentiment Analysis Using `SentimentIntensityAnalyzer`: Behind the Scenes: [60]

Sentiment analysis, a vital component of Natural Language Processing (NLP), seeks to quantify the emotional tone of a given text. The NLTK library, well-known for its NLP capabilities, offers the `SentimentIntensityAnalyzer` class that enables the assessment of sentiment in text data through the assignment of a sentiment score. This score, often referred to as the "compound" score, is calculated by analyzing the words present in the text and their respective sentiment intensities.

#### 1. Initialization and Lexicon Utilization:

When a `SentimentIntensityAnalyzer` instance is created, it initializes an internal lexicon or dictionary. This lexicon contains a curated list of words along with their associated sentiment scores. These scores reflect the intensity of sentiment expressed by each word, typically ranging from -1 (indicating strong negativity) to 1 (indicating strong positivity).

#### 2. Tokenization and Polarity Assignment:

Upon providing a text for sentiment analysis, the `analyzer.polarity_scores(text)` method dissects the text into individual tokens, usually words. Each token is then matched against the lexicon to retrieve its sentiment score. This process involves assigning a polarity score to each token, signifying the extent of positivity or negativity conveyed by that word. The score represents the intensity of sentiment conveyed by the word.

### 3. Composition of Overall Sentiment Score:

To obtain an all-encompassing sentiment score for the entire text, the individual polarity scores of the tokens are combined. This calculation takes into consideration not only the intensity of sentiment attached to each word but also the context in which these words are used.

### 4. The Compound Score:

The resultant sentiment score, referred to as the "compound" score, encapsulates the overall sentiment of the text. This compound score is pivotal in gauging the text's emotional tone. A higher positive compound score suggests a predominantly positive sentiment, while a lower negative score indicates a predominantly negative sentiment. Scores that hover close to zero often indicate a more neutral sentiment within the text.

### **3.3.3 How the Unique Words/Vocabulary calculated.**

To compute the count of distinct words, a function called `preprocess_text()` was developed. This function serves to eliminate stopwords and tokenize words. Additionally, it involves the application of lemmatization—a procedure that aims to simplify words to their fundamental or root form, streamlining words with shared meanings into a cohesive representation. Consequently, the remaining words are identified as unique terms.

### **3.3.4 How the Grammar Analysis calculated.**

For conducting grammar analysis, the code employs the `LanguageTool` library sourced from `language_tool_python`, an open-source grammar tool. This library facilitates the detection of grammar errors and spelling mistakes present in the text using a Python

script. As a default configuration, `language_tool_python` automatically acquires a LanguageTool server .jar and operates it in the background to locally identify grammar errors. [61] In this project, total of 6 types of categories will be analysed by LanguageTool which are:

**Typos:** Typos are errors caused by mistyping or misspelling words, resulting in unintended words or incorrect spellings.

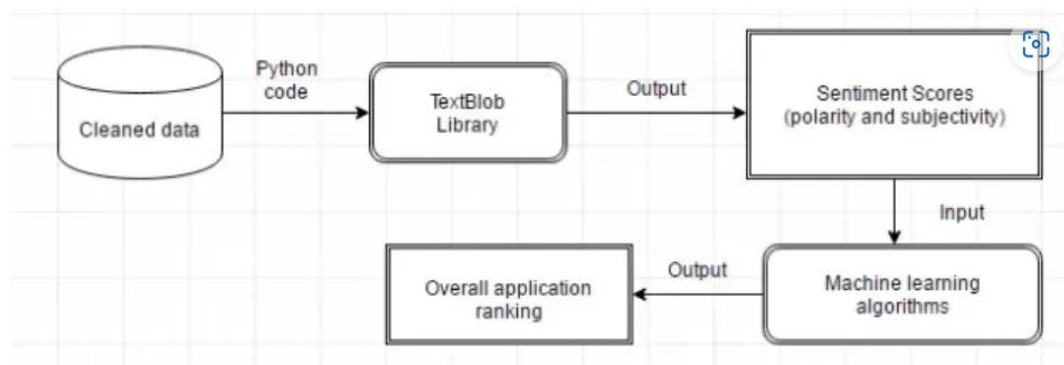
**Casing:** Casing errors occur when the wrong capitalization is used for words, such as using all uppercase or all lowercase letters inappropriately.

**Grammar:** Grammar errors refer to mistakes in the structure of sentences, including subject-verb agreement, verb tense, and word order.

**Style:** Style errors involve inconsistencies in writing style, such as using different formatting, phrasing, or expressions that are not in line with the desired writing style.

**Punctuation:** Punctuation errors occur when incorrect or missing punctuation marks are used, leading to unclear or improperly structured sentences.

### 3.3.5 How the Tone calculated.



*Figure 3.3.1 textblob process* [62]

It employs the TextBlob library to perform sentiment analysis and categorizes the tone based on polarity, subjectivity, word count, and formality of the message. For each message in the list of messages, the function creates a TextBlob object (blob) to analyze its sentiment. The initial tone is set to "Unknown". [63]

TextBlob's sentiment property provides sentiment analysis metrics. polarity ranges from -1 (negative) to 1 (positive), indicating the sentiment of the text. subjectivity ranges from 0 (objective) to 1 (subjective), representing how subjective the text is. If the subjectivity of the message is greater than or equal to 0.5, it checks the polarity to determine the tone. If the subjectivity is less than 0.5, it assigns a different set of tones.[62]

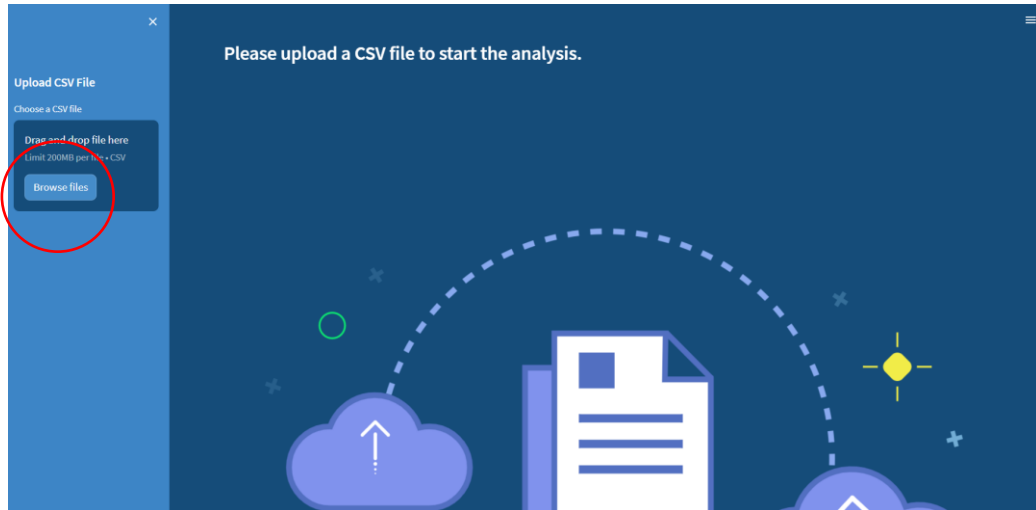
Within the subjective category, it further divided the tone based on polarity. For example, if a message was subjective (subjectivity  $\geq 0.5$ ) and had a polarity score greater than or equal to 0.3, classified it as "Confident." If the polarity was between 0.1 and 0.3, it was "Informative," and if it was less than or equal to -0.1, it was "Casual." If none of these conditions were met, you labelled it as "Neutral." Similarly, within the objective category (subjectivity  $< 0.5$ ), categorized messages based on their polarity. For instance, if a message was objective and had a polarity score greater than or equal to 0.3, you classified it as "Direct." If the polarity was between 0.1 and 0.3, it was "Curious," and if it was less than or equal to -0.1, it was "Assertive." If none of these conditions were met, it labelled it as "Neutral." In addition, it also considered formality by checking the word count and the number of pronouns (PRP) in a message. If a message had more than 10 words and a low count of pronouns (indicating fewer self-references), it labelled it as "Formal." Lastly, if a message didn't meet any of the conditions in the above steps, you assigned it a default tone of "Neutral."

The reason of TextBlob is used for tone analysis rather than SentimentIntensityAnalyzer from NLTK library is because TextBlob has a sentiment property that not only provides the polarity score (positive, negative, or neutral) but also includes a subjectivity score. This combination of sentiment and subjectivity information can be used to infer the tone of the text.

### 3.4 Overview of the English Practice Chatbot Report 's user interface

The report's user interface has been designed in a simple and straight forward way. User just need to upload the csv file by clicking the "Browse Files" button, then wait for the analysis result.

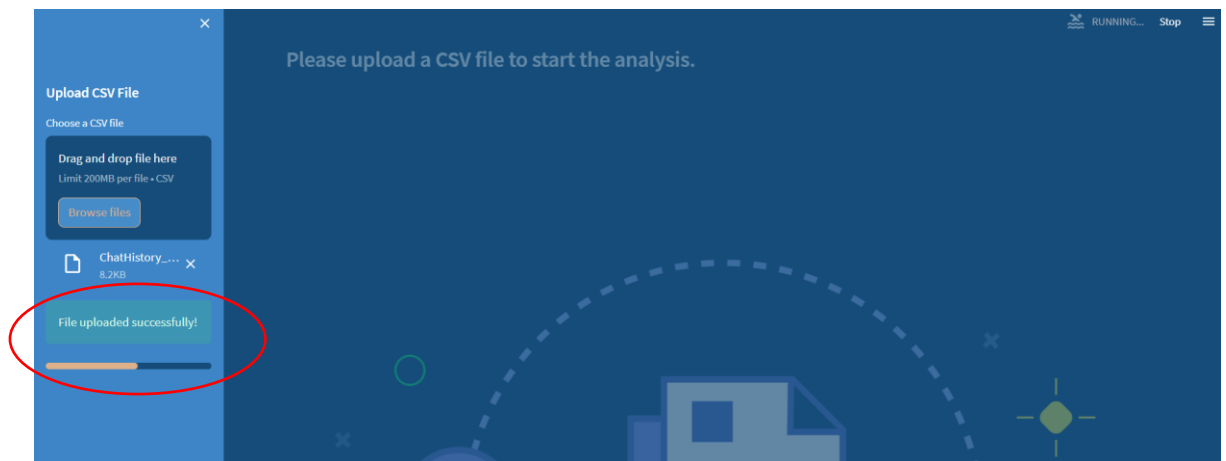
#### Uploading the input file (Downloaded from the Chatbot Interface)



*Figure 3.4.1 Upload interface of English Practice Chatbot Report*

User will need to upload the csv file by clicking the "browse file button".

#### Waiting Screen



*Figure 3.4.2 Waiting Screen*

Once the user has finished uploading the file, they will receive a message confirming the successful upload of the file. Additionally, a progress bar will be displayed to provide a visual representation of the analysis progress.

#### Result Screen





**Figure 3.4.3 Result Screen**

After done loading, the result will be shown to user.

### 3.5 How the English Practice Chatbot Report uses learning analytics

The report incorporates learning analytics to monitor the progress of users in learning English through the chatbot. Learning analytics involves gathering and examining data on the user's interactions with the chatbot to gain a deeper understanding of their learning process.

To analyse the user's communication skills and monitor their improvement over time, the report employs a range of metrics, including engagement, sentiment, vocabulary, grammar, and tone. For instance, the report can monitor the user's use of different words and phrases to ascertain if their vocabulary is growing and assess their response time to track their fluency progress.

By leveraging learning analytics, the report offers valuable insights into the user's learning process, allowing them to identify areas requiring improvement. It also provides encouragement and motivation by highlighting the user's accomplishments and progress. Furthermore, educators and language tutors can use the report to monitor their students' progress, provide tailored feedback, and offer guidance.

### 3.6 Timeline

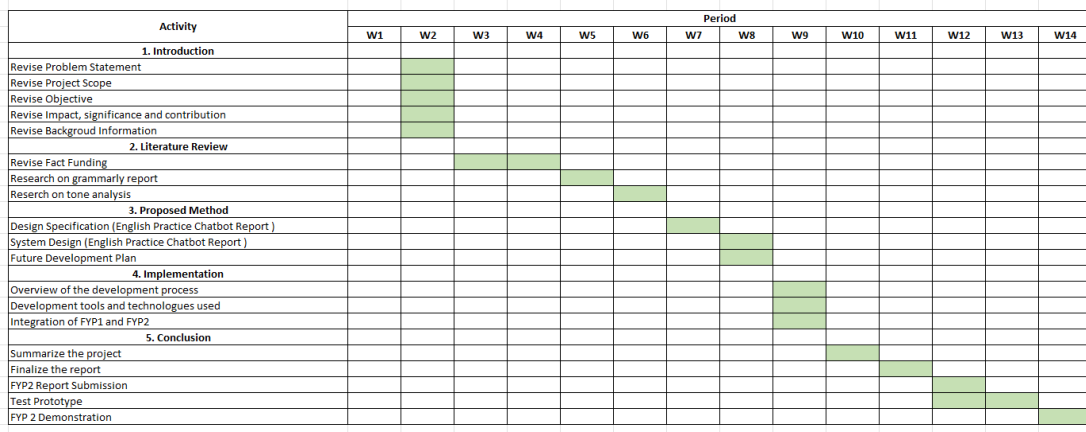


Figure 3.6.1 Gantt Chart

### 3.7 Future Development Plan/ Future Work

In line with the continuous dedication to delivering a thorough language learning encounter, this project has the potential of devising a novel functionality that will significantly transform user interaction with our English Practice Chatbot. This

endeavor is centered around the enhancement of listening abilities, both of which play pivotal roles in the acquisition of language skills.

### 1. Progress Monitoring Through Pronunciation:

With the integration of the speaking feature (added after user testing), the project is planned to take language learning a step further. Users' spoken inputs will be recorded and evaluated, allowing them to gauge their progress over time. The collected data will be presented in the user's personalized progress report. This visual representation of their pronunciation advancement will serve as motivation and a roadmap for their language learning endeavors.

### 2. Enhanced Listening Skills:

To bolster users' listening skills, interactive playback functionality also planned to be added. Users will have the option to replay both their own spoken input and the chatbot's responses. This side-by-side comparison will enable users to discern discrepancies between their pronunciation and the model response. By actively participating in this iterative process, users can fine-tune their speaking skills and attune their ears to native-like pronunciation.

## Chapter 4 Implementation

### 4.1 Overview of the development process for English Practice Chatbot

The chatbot project underwent various stages of development. At the outset, extensive research and analysis were conducted to understand the objectives and requirements of the project. Subsequently, a plan was formulated with a timeline that identified the project's milestones and tasks.

The development phase then commenced, involving the creation of the chatbot's frontend and backend components. React Native was used to develop the user interface. The backend was built using Node.js and Firebase, providing a serverless architecture that facilitated fast and efficient development and deployment. Additionally, the OpenAI API was integrated to enable natural language processing capabilities.

Various tools and technologies, such as Visual Studio Code and Expo, were employed throughout the development process, and comprehensive testing and debugging were conducted to ensure that the chatbot was functioning correctly and met the project requirements. The development process was iterative, with continuous feedback and improvements made to ensure that the final product met the project's objectives.

#### 4.1.1 Development tools and technologies used for English Practice Chatbot.

The chatbot project utilized both front-end and back-end technologies to achieve its functionality. **React Native**, a JavaScript framework was used for the front-end, along with **Expo**, a set of tools and services that simplify React Native application development. [64]**Node.js**, a JavaScript runtime environment for server-side applications, was utilized as the back end,[65] along with **Firebase**, a cloud-based platform that offers various backend services such as database and authentication.[66] To improve the chatbot's functionality, several APIs were integrated, including the **Open AI ChatGPT 3 API**. This API is based on the GPT-3 architecture and allows for the generation of high-quality human-like responses to user inputs. [67]The integration of this API allowed the chatbot to provide more engaging and realistic conversations with users, making it a valuable addition to the project. Besides, **Grammarly Text Editor SDK** for writing suggestions and the **AI text paraphrasing API from Smodin** for alternative expression discovery. Besides, various development tools were utilized

to streamline the development process, such as **Visual Studio Code** as the primary code editor, Git for version control, and multiple libraries and packages available on npm.

## 4.2 Integration of Open AI API, Grammarly SDK and paraphraser tool

### 4.2.1 how the chatbot integrates Open AI API

```

88     const openaiData = await fetch(
89       "https://api.openai.com/v1/engines/text-davinci-003/completions",
90       {
91         method: "POST",
92         headers: {
93           "Content-Type": "application/json",
94           Authorization:
95             "Bearer sk-InwpyLTLVJKV6Sdx33oT3BlbkFJmdxmYRvQpu2Z6MwpZcAG",
96         },
97         body: JSON.stringify({
98           prompt: `Conversation:\nUser: ${message}\nAI: `,
99
100          temperature: 0.87,
101          max_tokens: 500,
102          top_p: 0.78,
103          frequency_penalty: 0.5,
104          presence_penalty: 0.5,
105        }),
106       })
107     ).then((res) => res.json());
108
109     const botMessage = openaiData?.choices?.[0]?.text;

```

*Figure 4.2.1 Integrating Open AI API*

The provided code integrates the OpenAI API to facilitate the conversation process. The `fetch()` function sends a POST request to the OpenAI API's endpoint, which is `https://api.openai.com/v1/engines/text-davinci-003/completions`. The request includes the Authorization header, which contains an API key that grants access to the OpenAI API.

The body of the request is a JSON object that contains several parameters. The prompt parameter provides the context for the conversation and the message that the user has sent. The temperature parameter controls the randomness of the AI's response. The max\_tokens parameter determines the maximum number of tokens the AI should generate in its response. The top\_p, frequency\_penalty, and presence\_penalty parameters control the quality and relevance of the generated response.

The response from the OpenAI API is then parsed as JSON, and the botMessage variable is assigned the generated response from the API. If the response is null or undefined, then botMessage is also null or undefined.

#### 4.2.2 How the chatbot integrates Grammarly Text Editor SDK

To integrate the Grammarly Text Editor SDK, the following steps need to be taken:

First, create a new Grammarly for Developers app to obtain the client key.

```
npm install @grammarly/editor-sdk-react
```

*Figure 4.2.2 Install Text Editor SDK*

Next, install the Text Editor SDK.

```
13 import { GrammarlyEditorPlugin } from "@grammarly/editor-sdk-react";

202 <GrammarlyEditorPlugin clientId="client_KZgmWUDd869iTv6vo5p5qi">
203   <View style={styles.inputContainer}>
204     <TextInput
205       style={styles.textInput}
206       placeholder="Type a message..."
207       value={message}
208       onChangeText={(text) => setMessage(text)}
209       onSubmitEditing={sendMessage}
210       returnKeyType="send"
211     />
212   </View>
213 </GrammarlyEditorPlugin>
```

*Figure 4.2.3 Add Grammarly Text Editor*

Third, add the Grammarly Text Editor Plugin to the text editors where Grammarly's assistance is required which is the <TextInput> for this project.

### 4.2.3 How the chatbot integrates Smodin paraphraser API

```

62  const sendMessage = async () => {
63    const options = {
64      method: "POST",
65      headers: {
66        "content-type": "application/json",
67        "X-RapidAPI-Key": "7fffcada11mshfa3f1ef26ea5df7p161e88jsn1a960b875f94",
68        "X-RapidAPI-Host":
69          "rewriter-paraphraser-text-changer-multi-language.p.rapidapi.com",
70      },
71      body: JSON.stringify({
72
73        language: "en",
74        strength: 3,
75        text: message,
76      }),
77    };
78
79    const paraphraseData = await fetch(
80      "https://rewriter-paraphraser-text-changer-multi-language.p.rapidapi.com/rewrite",
81      options
82    ).then((response) => response.json());
83
84    const paraphrasedMessage = paraphraseData.rewrite;

```

*Figure 4.2.4 Integrating Paraphraser API*

This code integrates the Smodin Paraphraser API by sending a POST request to the API endpoint with specific parameters. The options object contains the request method (POST), headers, and the body. The headers object contains the content type and the API key required to access the Paraphraser API. The X-RapidAPI-Host header specifies the API host. The body object contains the language, strength, and text parameters. The language parameter specifies the input language. In this case, it is set to "en" for English. The strength parameter determines the level of paraphrasing. It is set to 3, which is a high level of paraphrasing. The text parameter is the input text to be paraphrased. The fetch function is used to send the POST request to the Paraphraser API endpoint. The await keyword is used to wait for the response from the API. The response is then converted to JSON format using the response.json() method. The paraphrased output is obtained from the paraphraseData object by accessing the rewrite property. Finally, the paraphrasedMessage variable contains the paraphrased output.

## 4.3 English Practice Chatbot Report implementation

### 4.3.1. Overview of how the English Practice Chatbot Report was built.

After the chatbot has been done, it proceeds to the development of English Practice Chatbot Report. As for the outset, again, extensive research and analysis were

conducted to understand the metric or and analysis that able to apply. Subsequently, a plan was formulated with a timeline that identified the project's milestones and tasks.

The development phase then commenced, involving the creation of the English Practice Chatbot Report frontend and backend components. **Streamlit** was used to develop the user interface which is a free and open-source framework tailored for the rapid development and sharing of aesthetically pleasing web applications dedicated to machine learning and data science. Streamlit is Python-based and is particularly well-suited for the needs of machine learning engineers.[68] The backend was built using python and importing various **Python libraries** and module for performing different task, such as natural language processing, data visualization, and more.

Various tools and technologies, such as **Visual Studio Code**, were employed throughout the development process, and comprehensive testing and debugging were conducted to ensure that the chatbot was functioning correctly and met the project requirements. The development process was iterative, with continuous feedback and improvements made to ensure that the final product met the project's objectives.

### 4.3.2. Python Libraries Imported

#### 4.3.2.1 Streamlit

```
import streamlit as st
```

*Figure 4.3.1 import streamlit*

Streamlit's open-source application framework presents the simplest avenue for data scientists and machine learning engineers to craft visually appealing and high-performing apps within a mere span of a few hours. This can be achieved entirely through the use of Python and comes at no cost.[69] Streamlit is a dynamic library for creating web applications optimized for data visualization and analysis. It transforms data scripts into interactive web interfaces, making it effortless to present analyses and insights to a broader audience. It has been utilized in this project to display the report for user.

#### 4.3.2.2 Streamlit Lottie



```
pip install streamlit-lottie
```

```
from streamlit_lottie import st_lottie
```

*Figure 4.3.2 import streamlit\_lottie*

Streamlit Lottie, facilitates the effortless incorporation of Lottie animations into Streamlit apps. Lottie animations, which are vector-oriented animations, elevate the visual allure and interaction of Streamlit's user interface.[70] In this project, Streamlit Lottie has been employed to introduce animations, enhancing the overall user interface experience.

#### 4.3.2.3 JSON

```
import json
```

*Figure 4.3.3 import json*

The JSON module plays a pivotal role in working with JSON (JavaScript Object Notation) data, a lightweight data interchange format.[71] It's utilized for loading Lottie animation files, contributing to the inclusion of dynamic visual elements in application.

#### 4.3.2.4 Pandas

```
import pandas as pd
```

*Figure 4.3.4 import pandas*

Pandas is a powerful library for data manipulation and analysis. It introduces DataFrames and Series to efficiently organize and manipulate data, proving invaluable in tasks such as filtering, sorting, reshaping, and analysis. [72] Throughout the project, Pandas enables functions like calculate\_unique\_words\_per\_month, calculate\_grammar\_errors\_per\_month, calculate\_month\_sentiment, calculate\_month\_tone, and calculate\_month\_message\_count.

#### 4.3.2.5 NumPy

```
import numpy as np
```

*Figure 4.3.5 import numpy*

NumPy is a fundamental package for numerical computations. It introduces arrays and matrices for handling numerical data, forming the bedrock for a range of scientific and mathematical operations.[73] In this project, NumPy plays a role in dealing with numerical computation.

#### 4.3.2.6 Matplotlib

```
import matplotlib.pyplot as plt
```

*Figure 4.3.6 import matplotlib.pyplot*

Matplotlib is a widely used library for crafting an array of visualizations, ranging from static to interactive. It equips with tools to generate plots, charts, and graphs, visually representing data in diverse formats. [74] Its influence extends to the creation of plots and charts within Streamlit applications.

#### 4.3.2.7 Plotly

```
import plotly.express as px
```

*Figure 4.3.7 import plotly.express*

Plotly is a specialized visualization library for interactive and dynamic visualizations. It empowers to create charts and graphs that users can engage with, offering capabilities such as zooming, panning, and hovering for additional information. [75]It has been used to plot different kind of chart in the report.

#### 4.3.2.8 NLTK (Natural Language Toolkit)

```
import nltk
from nltk.sentiment import SentimentIntensityAnalyzer
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
```

*Figure 4.3.8 import nltk*

The Natural Language Toolkit (NLTK) is a prominent open-source Python library designed for natural language processing (NLP). It offers an accessible interface to undertake a diverse range of NLP tasks, encompassing tokenization, stemming, lemmatization, parsing, and sentiment analysis. NLTK is a widely embraced tool among researchers, developers, and data scientists globally. It empowers them to create NLP applications and analyze textual data effectively. [76] Moreover, NLTK equips

with sentiment analysis capabilities for gauging emotional tones in text. In this project, NLTK serves a pivotal role in functions such as `preprocess_text`, `calculate_language_proficiency`, `calculate_sentiment_score`, `detect_tone`, and `calculate_month_sentiment`.

#### 4.3.2.9 WordCloud

```
from wordcloud import WordCloud
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

*Figure 4.3.9 import wordcloud*

The WordCloud library generates captivating visual representations known as word clouds. These graphics are constructed from text data, displaying words with varying font sizes based on frequency. WordCloud is valuable for visually summarizing common terms, facilitating quick insights into prevailing themes. To get started making a word cloud in Python, some packages such as pandas, numpy (libraries that is used for handling multi-dimensional arrays and matrices), wordcloud, and matplotlib (library that enables many other libraries to run and plot on its base, including seaborn or wordcloud) need to be installed. [77] Within the project, WordCloud is used to display the unique word used by the user.

#### 4.3.2.10 LanguageTool Python

```
from language_tool_python import LanguageTool
```

*Figure 4.3.10 import language\_tool\_python*

This library interfaces with the LanguageTool grammar and style checker API. It aids in identifying grammar errors and stylistic inconsistencies. It could check for spelling and grammar errors in over 20 languages such as English, German, French, Spanish and so on, thereby enhancing the quality of language analysis. By connecting to this API, the grammatical and stylistic accuracy of text can be refined. [61] Its contributions are evident in functions such as `calculate_language_proficiency`, `calculate_grammar_errors`, and `calculate_grammar_errors_per_month`.

#### 4.3.2.11 Datetime

```
from datetime import datetime
```

*Figure 4.3.11 import datetime*

The datetime module provides classes and functions for managing dates and times. It's a cornerstone for handling time-based data, facilitating conversions between various time formats and calculations involving time intervals. [78] Throughout the project, datetime ensures precise temporal handling in functions.

#### 4.3.2.12 Base64

```
import base64
```

*Figure 4.3.10 import base64*

The base64 module is employed for encoding and decoding binary data, a crucial process when handling files like image files. It's essential for efficiently transmitting binary data over text-based protocols.[79] It has been used to enable user to export the unique word into csv file.

#### 4.3.2.13 TextBlob

```
from textblob import TextBlob
```

*Figure 4.3.11 import textblob*

TextBlob is a versatile library designed for processing textual data. It offers tools for text classification, part-of-speech tagging, noun phrase extraction, and sentiment analysis. TextBlob simplifies complex language processing tasks and makes it easier to extract valuable insights from text.[60] Its impact is felt in the detect\_tone functions.

### 4.3.3. Components in English Practice Chatbot Report

#### 4.3.3.1 preprocess\_text

```
nltk.download('stopwords')
```

```
def preprocess_text(text):
    # Tokenize the text into words and filter out stopwords
    stop_words = set(stopwords.words('english'))
    words = word_tokenize(text)
    lemmatizer = WordNetLemmatizer()
    words = [lemmatizer.lemmatize(word.lower()) for word in words if word.isalpha() and word.lower() not in stop_words]
    return words
```

*Figure 4.3.12 preprocess\_text(text)*

`preprocess_text(text)` serves to prepare and refine a given text in the uploaded csv file for further analysis. This function encompasses several crucial text preprocessing steps, each contributing to the transformation of raw textual content into a more manageable and informative format.

The initial step involves the creation of a set of stopwords using the `stopwords.words('english')` function from the *Natural Language Toolkit (NLTK)* library. Stopwords encompass common words like "and," "the," "is," and others, which often carry limited semantic value and are typically omitted during text analysis. These stopwords will later be employed to filter out words that don't significantly contribute to the meaning of the text.

The subsequent operation is tokenization, facilitated by the `word_tokenize(text)` function from the *NLTK library*. Tokenization is the process of breaking down the input text into individual words or tokens. This transformation is pivotal for subsequent analyses, as it provides a basis for dissecting the text into its constituent linguistic units.

Following tokenization, the code introduces a lemmatizer using the `WordNetLemmatizer()` class from the *NLTK library*. Lemmatization is the process of reducing words to their base or root form, aiming to consolidate words with similar meanings into a unified representation. For example, Original Sentence: "She was jumping over the fences." Lemmatized Sentence: "She was jump over the fence.". This step enhances the effectiveness of subsequent analyses by simplifying words to their essential core.

The heart of the function resides in a list comprehension, which iterates through each tokenized word. For each word, two criteria are evaluated: first, whether the word consists solely of alphabetic characters (excluding punctuation and special characters), and second, whether the lowercase version of the word is not present in the set of stopwords. Words that satisfy these conditions are subjected to lemmatization, wherein they are transformed to their base form using the lemmatizer introduced earlier.

Finally, the function returns a list containing the lemmatized and cleaned words. These words can be further utilized in various analyses, such as sentiment analysis, topic modeling, or any task that requires a refined and standardized representation of

the original text. In essence, the `preprocess_text` function streamlines the text by removing stopwords, lemmatizing words, and producing a processed collection of words that retains essential linguistic content while discarding extraneous noise.

Let's explore how this function operates using the example text "Hello World, Nice to Meet you."

In the initial step, common English stopwords like "and," "to," and "the" are compiled into a set. Stopwords are words that typically convey limited significance in language analysis. Next, the example text is divided into individual words through a process known as tokenization, yielding ["Hello", "World", ",", "Nice", "to", "Meet", "you"]. Tokenization enables the text to be handled at the word level.

Following tokenization, a lemmatizer is employed to simplify words to their base form. The core of the function is a list comprehension, where each word undergoes scrutiny. Words that are exclusively composed of alphabetic characters and are not in the stopwords set pass the criteria. In this example, the word "to" is discarded due to being a stopword, while the lemmatized forms of "Nice" and "Meet" emerge as "nice" and "meet."

#### 4.3.3.2 Message per Day & Month

```
# Progress by Timestamp (Messages per Day)
st.subheader('Messages per Day & Month')
timestamps = language_proficiency['timestamps']
message_dates = [timestamp.date() for timestamp in timestamps]
message_count_per_day = pd.Series(message_dates).value_counts().sort_index()
```

*Figure 4.3.3.2(a) message per day & month*

To enable this analysis, the script assumes the presence of a dictionary named `language_proficiency`, which presumably contains the results of a language proficiency analysis. The line `timestamps = language_proficiency['timestamps']` retrieves the timestamp data from this dictionary. These timestamps are likely associated with the date and time when messages were sent.

To facilitate the per-day analysis, the next line `message_dates = [timestamp.date() for timestamp in timestamps]` plays a crucial role. It extracts only the

date component from each timestamp, creating a list of message dates. This step is fundamental in grouping and aggregating messages based on a daily timeframe.

The culmination of this analysis occurs in the line `message_count_per_day = pd.Series(message_dates).value_counts().sort_index()`. Here, the list of extracted message dates is transformed into a Pandas Series. The `value_counts()` function operates on this Series, meticulously calculating the frequency of messages for each date. This effectively results in a count of messages sent on each day. To ensure the presentation is organized chronologically, the `sort_index()` function is employed to arrange the dates in ascending order.

```
def calculate_month_message_count(messages):
    # Convert the 'Date' column to datetime format
    messages['Date'] = pd.to_datetime(messages['Date'], format='%d/%m/%Y')
    # Extract the month and year from the timestamps in the messages
    message_dates = [timestamp.date() for timestamp in messages['Date']]
    messages['Month'] = pd.Series([date.strftime('%Y-%m') for date in message_dates])
    # Count messages for each month
    message_counts_per_month = messages['Month'].value_counts()

    return message_counts_per_month
```

**Figure 4.3.3.2(b) message per month**

`calculate_month_message_count` is designed to analyse and quantify the distribution of messages over different months within a provided dataset of messages. The function takes a single argument, `messages`, which is expected to contain the message data. To begin, the code converts the 'Date' column within the messages dataset into the datetime format using the `pd.to_datetime()` function. This ensures that the date information is properly recognized and processed.

Next, the script extracts the date component from each timestamp in the 'Date' column using a list comprehension. This produces a list called `message_dates`, which contains the individual dates associated with the sent messages. Subsequently, a new column named 'Month' is added to the `message`'s dataset. This column is populated by converting each date in the `message_dates` list into a string representation using the '%Y-%m' format. This format displays the year and month of each message, such as '2023-08'. The final step involves calculating the message count for each unique month present in the 'Month' column. The `value_counts()` function is employed to tally the occurrences of each unique month, generating a Pandas Series named `message_counts_per_month`.

## 4.3.3.3 Sentiment Distribution

```

# Sentiment Distribution Plot
# Count sentiment categories
positive_count = len([score for score in language_proficiency['sentiment_scores'] if score > 0])
neutral_count = len([score for score in language_proficiency['sentiment_scores'] if score == 0])
negative_count = len([score for score in language_proficiency['sentiment_scores'] if score < 0])
labels = ['Positive', 'Neutral', 'Negative']
counts = [positive_count, neutral_count, negative_count]

# Calculate percentage for each sentiment category
total_count = sum(counts)
percentages = [(count / total_count) * 100 for count in counts]

# Create a DataFrame for the sentiment distribution
sentiment_df = pd.DataFrame({'Category': labels, 'Count': counts, 'Percentage': percentages})

```

*Figure 4.3.3.3(a) to plot sentiment distribution (in main())*

In the main function, the sentiment scores calculated above are used to determine the distribution of sentiment categories (Positive, Neutral, Negative). The counts of sentiment scores falling into each category are calculated. The percentages list is calculated by dividing each category's count by the total count and then multiplying by 100 to convert it into a percentage. A DataFrame named `sentiment_df` is created to store the sentiment distribution data, including the sentiment category, count, and percentage. Using Plotly Express (`px.pie`), a pie chart named `fig_sentiment` is created to visualize the sentiment distribution. It includes values, names, hover data (percentage), title, and color scheme. The pie chart's layout is customized using the `update_traces` method to include hover information and adjust text font size.

```

# Perform sentiment analysis using NLTK's SentimentIntensityAnalyzer
sentiment_score = calculate_sentiment_score(message['Message'])
sentiment_scores.append(sentiment_score)

```

*Figure 4.3.3.3(b) lines in language\_proficiency()*

When the `language_proficiency` is called, it will call the `calculate_sentiment_score` function, passing the text content of the current message (`message['Message']`) as an argument. The `calculate_sentiment_score` function uses NLTK's `SentimentIntensityAnalyzer` to analyze the sentiment of the text and returns a sentiment score. After calculating the sentiment score for the current message, the code appends the score to the `sentiment_scores` list. This list will eventually hold sentiment scores for all messages.



```

from nltk.sentiment import SentimentIntensityAnalyzer

def calculate_sentiment_score(text):
    # Initialize SentimentIntensityAnalyzer
    analyzer = SentimentIntensityAnalyzer()
    # Calculate sentiment score
    sentiment_scores = analyzer.polarity_scores(text)
    return sentiment_scores['compound']

```

**Figure 4.3.3.3(c) calculate\_sentiment\_score(text)**

The function `calculate_sentiment_score(text)` is defined to calculate the sentiment score of a given text using *the SentimentIntensityAnalyzer from the NLTK library*.

\* How `SentimentIntensityAnalyzer()` works behind the screen can refer to 3.3.2 How the Sentiment Distribution calculated

#### 4.3.3.4 Unique Words/ Vocabulary

```

# Unique Words Analysis and Unique Words Table (side by side)
st.subheader('Unique Words/Vocabulary 📌 ')

# Unique Words Analysis
unique_words_count = language_proficiency['unique_words_count']
unique_words_text = f"<h2>The number of unique words used: {unique_words_count}</h2>"
st.markdown(unique_words_text, unsafe_allow_html=True)

wordcloud_data = ' '.join([message['Message'] for message in messages])
wordcloud = WordCloud(width=500, height=300).generate(wordcloud_data)
fig, ax = plt.subplots()
ax.imshow(wordcloud, interpolation='bilinear')
ax.axis('off')
st.pyplot(fig, use_container_width=True)

# Calculate unique words per month
unique_words_per_month_df = calculate_unique_words_per_month(pd.DataFrame(messages))

```

**Figure 4.3.3.4(a) to show the unique words/ vocabulary (in main())**

The total count of unique words is retrieved from the *language\_proficiency dictionary*, and this information is displayed using an HTML-formatted text. The code tokenizes the text messages, filters out stopwords, and lemmatizes the words. It maintains a set called `unique_words` to store unique words. Inside the `calculate_language_proficiency` function, you can access the count of unique words using `unique_words_count`. Besides, a Word Cloud visualization is generated to showcase frequently used words, created by joining all the message texts together. This Word Cloud is displayed using the *plt* library and showcased in the Streamlit app. Additionally, the count of unique words per month is calculated using the `calculate_unique_words_per_month` function. This data is then transformed for presentation, and a line chart is constructed using *Plotly Express*. This chart helps visualize the trend of unique words usage over different months.

```

words = preprocess_text(message['Message'])
# Calculate grammar error using LanguageTool

unique_words = set()
# Track unique words and their frequencies
unique_words.update(words)
for word in words:
    if word in word_frequencies:
        word_frequencies[word] += 1
    else:
        word_frequencies[word] = 1
unique_words_count = len(unique_words)

```

**Figure 4.3.3.4(b) lines in `language_proficiency()`**

In the `calculate_language_proficiency` function, firstly, it will remove all the stopped words by calling `preprocess_text()` (can refer 4.3.3.1 `preprocess_text`). Then, it focuses on tracking and counting unique words. The process starts by initializing an empty set called `unique_words` to keep track of encountered words. As each message is processed, the set is updated with the words from that message. A loop iterates through the words, incrementing their frequency count in a dictionary called `word_frequencies`. If a word is new, it's added to the dictionary with an initial count of 1.

#### 4.3.3.5 Grammar Error

```

from language_tool_python import LanguageTool

```

**Figure 4.3.3.5(a) import `language_tool_python`**

To carry out the grammar analysis, LanguageTool is employed. It is a standalone open-source proofreading and grammar checking tool that can be used to check the grammar and spelling of text in various languages. It is written in Java and can be used as a standalone application, integrated into various text editors, or accessed via a web interface. In order to use it in python, we need to imports the LanguageTool class from the `language_tool_python` module.

```

tool = LanguageTool('en-US')

```

**Figure 4.3.3.5(b) defining `tool`**

This line creates an instance of the LanguageTool class and initializes it with the language code 'en-US', which represents American English. we can specify the appropriate language code for the text we are analyzing.

```

def calculate_grammar_errors(text, tool):
    # Check grammar errors in the text
    errors = tool.check(text)
    return errors

```

*Figure 4.3.3.5(c) calculate\_grammar\_error*

This function takes a text input and the initialized LanguageTool instance as arguments. It uses the check() method of the LanguageTool class to analyze the input text and identify any grammar, spelling, or style errors. The method returns a list of Error objects, each representing a detected error in the text.

The grammar errors are categorized based on their severity or type. The Error objects returned by LanguageTool have a category attribute that indicates the type of error, such as "typos," "grammar," "style," etc. The errors are counted and grouped by month and error category to provide insights into the types of errors made over time.

#### 4.3.3.6 Tone Distribution

```

from textblob import TextBlob

def detect_tone(messages):
    tones = []

    for message in messages:
        blob = TextBlob(message['Message'])
        tone = "Unknown"

        polarity = blob.sentiment.polarity
        subjectivity = blob.sentiment.subjectivity
        word_count = len(blob.words)

        if subjectivity >= 0.5:
            if polarity >= 0.3:
                tone = "Confident"
            elif polarity >= 0.1:
                tone = "Informative"
            elif polarity <= -0.1:
                tone = "Casual"
            else:
                tone = "Neutral"
        else:
            if polarity >= 0.3:
                tone = "Direct"
            elif polarity >= 0.1:
                tone = "Curious"
            elif polarity <= -0.1:
                tone = "Assertive"
            else:
                tone = "Neutral"

        # Check for formality
        if word_count > 10 and blob.tags.count('PRP') <= 1:
            tone = "Formal"

        tones.append(tone)

    return tones

```

*Figure 4.3.3.6 detect\_tone()*

The process of calculating the tone within the provided code is accomplished through the integration of the TextBlob library for sentiment analysis. This intricate process involves the systematic evaluation of messages, considering a range of parameters to

## CHAPTER 4

determine the underlying tone. For every message, a TextBlob object named "blob" is created to assess its sentiment, initiating the tone classification as "Unknown." Subsequently, sentiment polarity and subjectivity scores are derived from the TextBlob's attributes, namely sentiment.polarity and sentiment.subjectivity. Additionally, the total word count of the message is computed using the len(blob.words) method.

The tone is determined as follows:

If subjectivity is high ( $\geq 0.5$ ), it checks polarity to classify the tone:

If polarity is high ( $\geq 0.3$ ), the tone is "Confident."

If polarity is moderately positive ( $\geq 0.1$ ), the tone is "Informative."

If polarity is moderately negative ( $\leq -0.1$ ), the tone is "Casual."

Otherwise, the tone is "Neutral."

If subjectivity is low ( $< 0.5$ ), it checks polarity to classify the tone:

If polarity is high ( $\geq 0.3$ ), the tone is "Direct."

If polarity is moderately positive ( $\geq 0.1$ ), the tone is "Curious."

If polarity is moderately negative ( $\leq -0.1$ ), the tone is "Assertive."

Otherwise, the tone is "Neutral."

The essence of the tone is then defined by a series of conditions, taking into account polarity, subjectivity, and word count. If the subjectivity level is substantial (at or above 0.5), polarity is examined to assign the tone as either "Confident," "Informative," "Casual," or "Neutral." Conversely, when subjectivity is lower, polarity guides the tone assignment as "Direct," "Curious," "Assertive," or "Neutral."

To further refine the tone analysis, a consideration of formality is incorporated. If the message surpasses a word count threshold of 10 and the occurrence of personal pronouns (PRP) is restricted to one or less, the tone is designated as "Formal".

#### 4.4 Chatbot training and Testing

To build the English Practice Chatbot, we utilized OpenAI's ChatGPT 3 text-davinci-003 language model, which is a powerful natural language processing tool capable of generating human-like responses to user inputs. The chatbot was pre-trained using a large dataset of English language texts to ensure that it could understand and respond to a wide range of inputs from users.

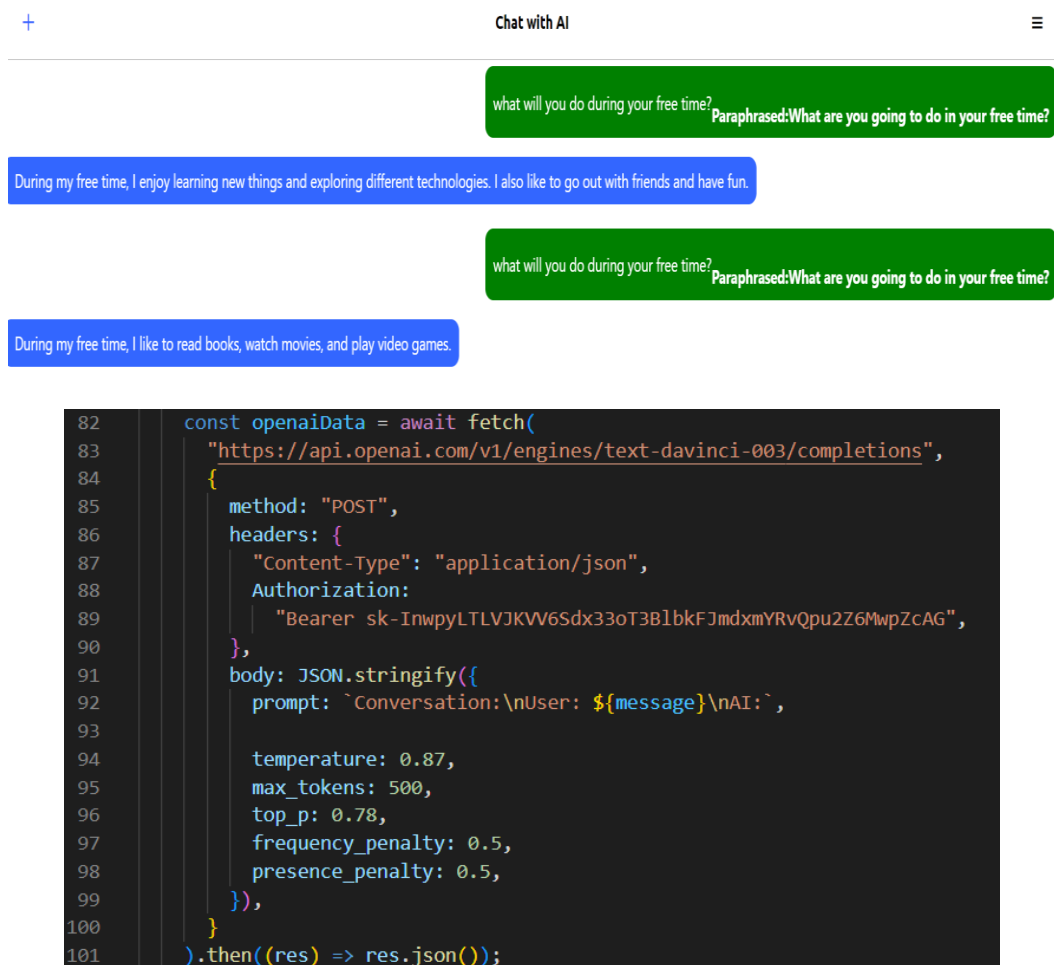
After integrating the language model into our chatbot framework, we conducted a series of tests to refine its performance. These tests involved providing the chatbot with various inputs and evaluating its responses. We used a trial-and-error approach to adjust parameters such as temperature (0.87), max\_tokens (500), top\_p (0.78), frequency\_penalty (0.5), and presence\_penalty (0.5) to optimize the chatbot's responses and ensure that they were appropriate, accurate, and relevant to the user's input.

We set the value of temperature to 0.87 to allow for some degree of unpredictability and creativity in the chatbot's responses. A higher temperature value allows for more variation in the responses, which can make the conversation more interesting and engaging for the user.

For max\_tokens, we chose a value of 500 to allow the chatbot to generate longer responses and provide more detailed information to the user. However, we also made sure to monitor the responses to ensure that they were not too lengthy and overwhelming for the user.

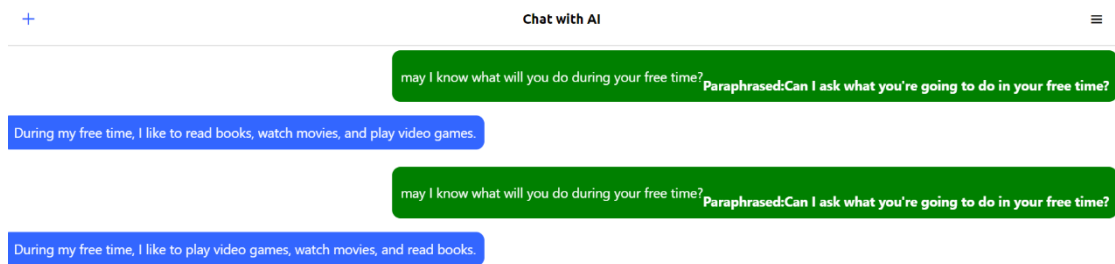
The top\_p parameter determines the probability distribution of the next word in the generated response, based on the context and previously generated words. We chose a value of 0.78 to strike a balance between generating diverse and creative responses while still ensuring that they were relevant to the user's input.

For frequency\_penalty and presence\_penalty, we set both values to 0.5. These parameters help to control the repetition of words and phrases in the chatbot's responses. A higher penalty value reduces the likelihood of the chatbot repeating words or phrases but can also lead to less coherent or relevant responses. By setting both values to 0.5, we aimed to strike a balance between avoiding repetition and maintaining coherence and relevance in the chatbot's responses.



*Figure 4.4.1 Test\_1*

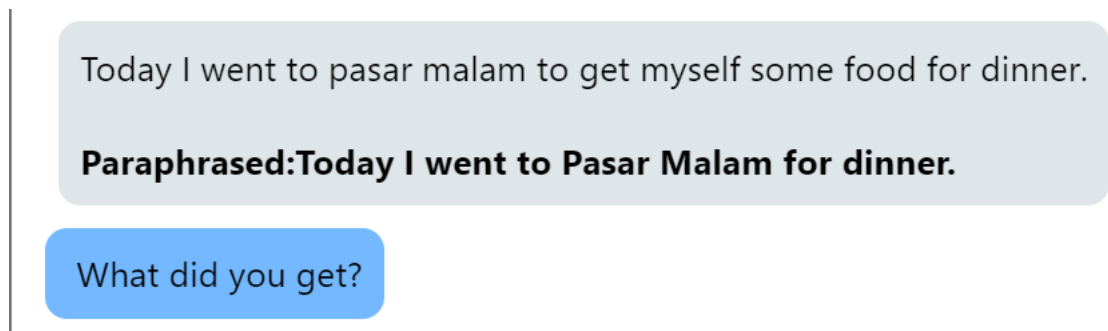
The above figure demonstrates the impact of adjusting the temperature, max\_tokens, top\_p, frequency\_penalty, and presence\_penalty parameters on the chatbot's performance. By setting the temperature parameter to 0.87, max\_tokens to 500, top\_p to 0.78, and frequency\_penalty and presence\_penalty to 0.5, we observed that the chatbot generated longer and more varied responses to the same prompt. For example, when asked about the chatbot's free time activities, the first response included "learning new things and exploring different technologies" and "going out with friends," while the second response mentioned "reading books, watching movies, and playing video games." This diversity in response enhances the learning experience for users, as they can acquire a wider range of vocabulary from the chatbot's varied responses.



```
const openaiData = await fetch(
  "https://api.openai.com/v1/engines/text-davinci-003/completions",
  {
    method: "POST",
    headers: {
      "Content-Type": "application/json",
      Authorization:
        "Bearer sk-InwpyLTLVJKV6Sdx33oT3BlbkFJmdxmYRvQpu2Z6MwpZcAG",
    },
    body: JSON.stringify({
      prompt: `Conversation:\nUser: ${message}\nAI:`,
      temperature: 0.37,
      max_tokens: 100,
      top_p: 1,
      frequency_penalty: 0,
      presence_penalty: 0,
    }),
  })
).then((res) => res.json());
```

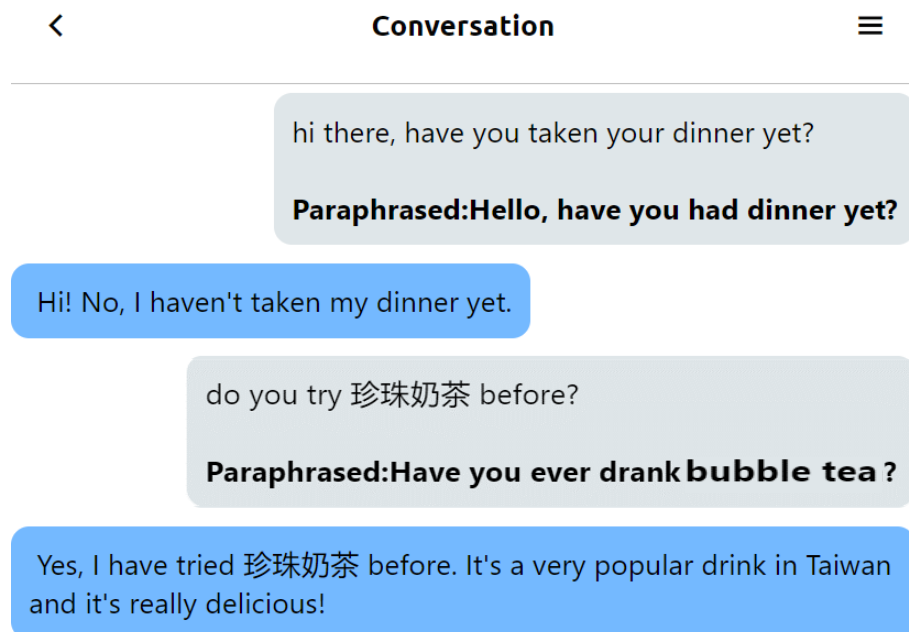
*Figure 4.4.2 Test\_2*

The above figure presents the results of adjusting the temperature, max\_tokens, top\_p, frequency\_penalty, and presence\_penalty parameters on the chatbot's performance. We set the temperature parameter to 0.37, max\_tokens to 100, top\_p to 1, and frequency\_penalty and presence\_penalty to 0 and observed that the chatbot produced shorter and less diverse responses to the same prompt. For instance, when asked about the chatbot's free time activities, the first response included "reading books, watching movies, and playing video games," while the second response mentioned "playing video games, watching movies, and reading books." The responses appear similar, the difference in the arrangement of words may not enhance the user's learning experience by acquiring new vocabulary. This setting may lead to inefficiencies in the learning process as users may not be exposed to a wide range of vocabulary.



*Figure 4.4.3 Test\_3*

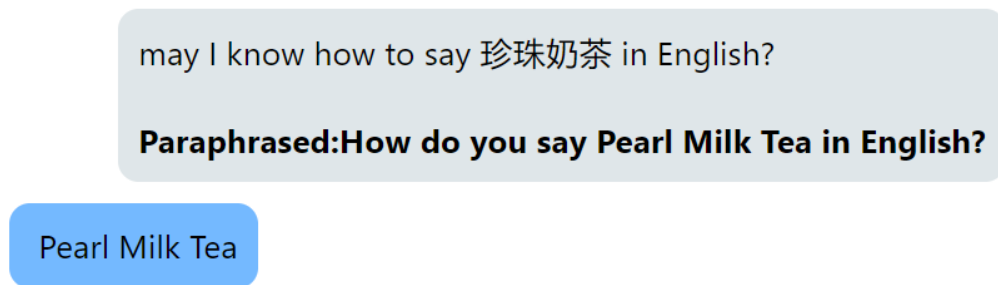
To effectively assist English learners in their language practice, it is crucial for the chatbot to handle situations where users may not know how to express certain words or phrases in English. As shown in the figure above, the chatbot successfully addresses this issue when a user prompts it with a statement such as "Today I went to pasar malam to get myself some food for dinner", which includes a Malay word. Despite the use of a non-English word, the chatbot still provides a relevant response by asking "What did you get?"



*Figure 4.4.4 Test\_4*



In the subsequent conversation, the user mentions food items in Mandarin, which is 珍珠奶茶, indicating potential difficulty in expressing these items in English. However, the chatbot is still able to generate appropriate responses, indicating its capability to handle such situations. For example, when asked if the chatbot has tried the item before, it responds with "Yes, I have tried 珍珠奶茶 before. It's a very popular drink in Taiwan and it's really delicious".



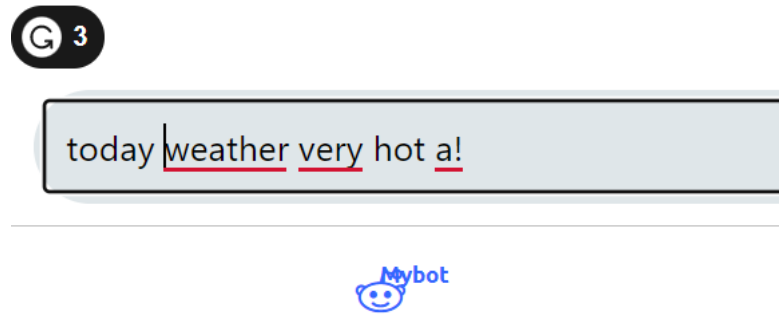
*Figure 4.4.5 Test\_5*

Additionally, when asked how to say "珍珠奶茶" in English, the chatbot provides the relevant answer. These examples demonstrate the chatbot's ability to assist non-native English speakers in their language practice by addressing their specific needs and challenges.

#### **4.5 Smodin Paraphraser Testing**

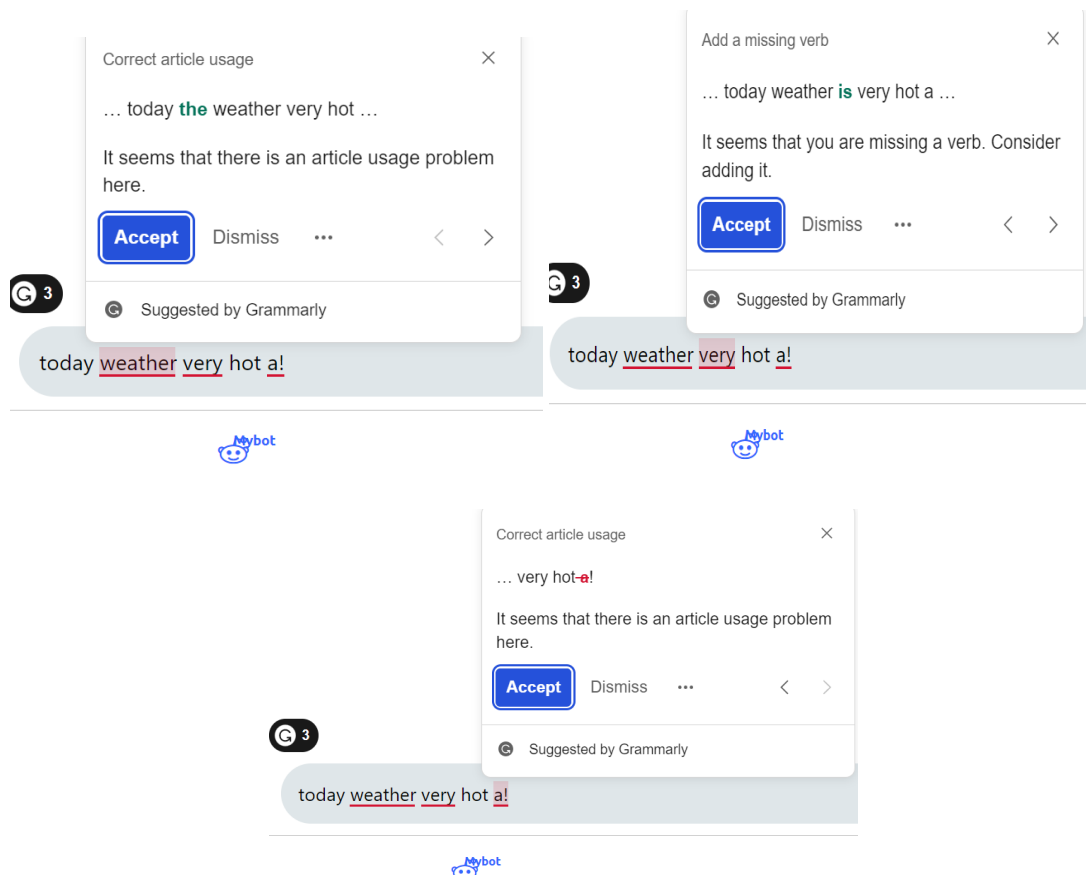
Moreover, it is noteworthy that the Smodin paraphraser was chosen due to its ability to paraphrase even when the user inputs in English. As shown in Figure 4.4.4 Test\_4, the paraphraser was able to recognize and suggest "bubble tea" when the user mentioned "珍珠奶茶" for the first time. Similarly, when the user mentioned "珍珠奶茶" which shown in figure 4.4.5 Test\_5 for the second time, the paraphraser suggested "Pearl Milk Tea," which is also relevant. This feature has made the learning process more effective as users can learn different vocabulary during the conversation.

## 4.6 Grammarly Text Editor SDK Testing



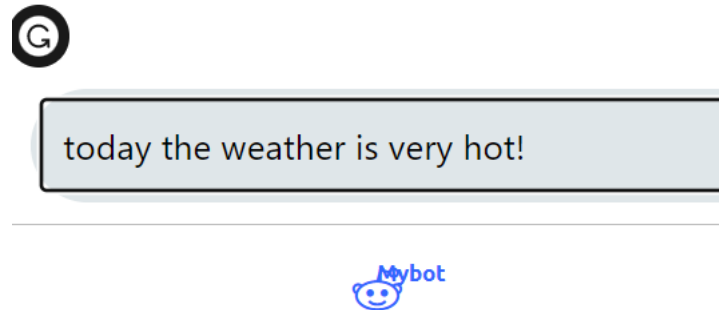
**Figure 4.6.1 Test\_1**

The figure above depicts a typical sentence written by a Chinese native speaker when expressing their feelings about the weather, using the sentence "today weather very hot a!". It is evident that the integrated real-time grammar checker has identified three potential grammatical errors, which are marked with red underlines.



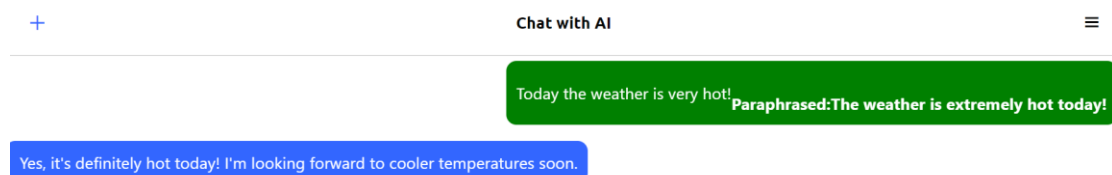
**Figure 4.6.2 Test\_2**

The suggested corrections from the grammar checker include adding "the" before the word "weather" and "is" before the word "very" as well as the removal of the word "a".



*Figure 4.6.3 Test\_3*

After accepting all the suggested corrections from the grammar checker, the sentence will be revised to "today the weather is very hot!" This correction process enables the user to learn as they are the ones who read the suggestions and accept the corrections to form a better sentence in expressing themselves. The integration of the real-time grammar checker also enables the bot to provide more accurate replies to the user, as the user's prompt is more easily understandable in terms of grammar.



*Figure 4.6.4 Test\_4*

Furthermore, the integration of Smodin paraphraser provides additional support to users to learn how to express their thoughts in different ways. For example, instead of typing "today the weather is very hot!" which was suggested by the grammar checker, users can also express themselves by saying "The weather is extremely hot today!" This feature enhances the learning experience as users are exposed to different ways of expressing themselves.

Therefore, the combination of the real-time grammar checker, Smodin paraphraser, and chatbot's response can provide a conducive learning and English practice environment for users.

#### 4.7 English Practice Chatbot Report training and Testing

To evaluate the English Practice Chatbot Report, this project underwent testing with a primary school child (Teng Chai Yen, Female) who possess a basic understanding of English. The inputs were gathered to generate the report. The testing process was supervised by the developer. The following presents a snapshot of the collected input.

```
Message,Date,Time
"hi! can we talk? i want to Practice English","08/04/2023","12:24:10 PM"
"I dont no wat to talk about can u suggest a topic","08/04/2023","12:24:20 pm"
"let's discuss diffrent holidays!","08/04/2023","12:26:49 PM"
>Hello! How's ur day going?","8/04/2023","12:30:03 pm"
"Im good, thank u. Have u tryed boba tea before?","8/04/2023","12:33:47 pm"
"It's rly tasty! U should try it at the nite market","8/04/2023","12:34:59 am"
"wat do u like 2 eat the nite market?","8/04/2023","12:35:21 am"
"Guess wat? tomorrow is school holiday, I not need fo to scholl!","8/04/2023","12:38:49 PM"
"tell me abt ur first day work! were u nervous?","8/04/2023","12:40:35 pm"
```

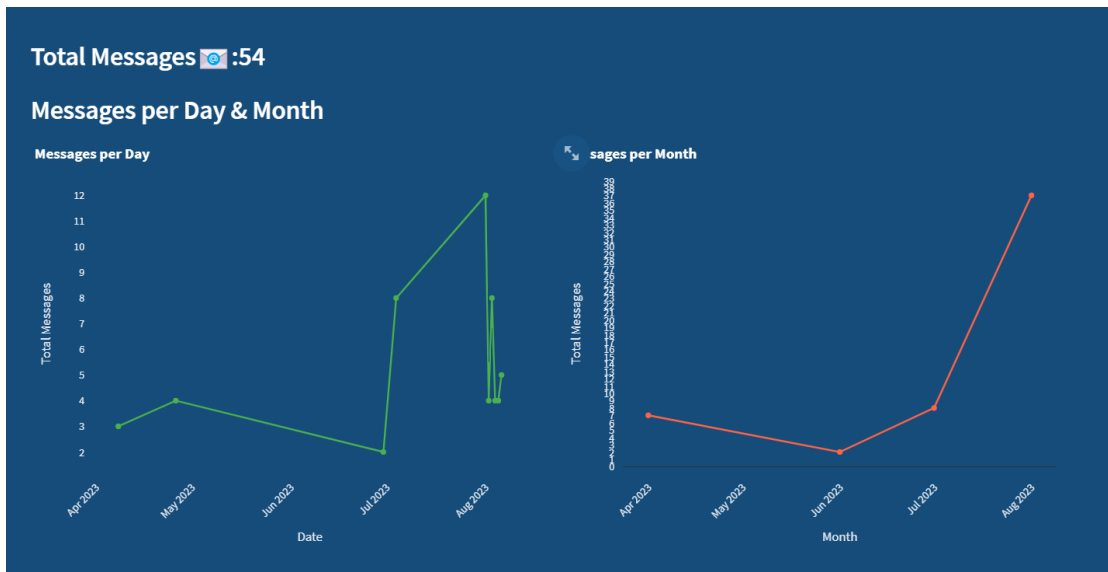
*Figure 4.7.1 snapshot of input collected.*

Then the file has been uploaded to the English Practice Chatbot Report to generate the report.

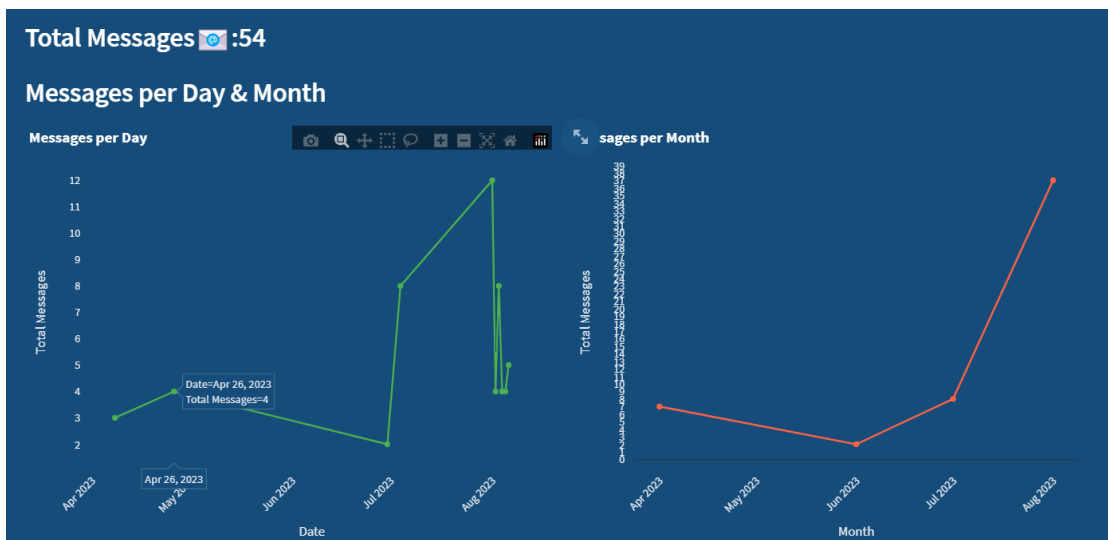


*Figure 4.7.2 section 1 of the report*

The above shows the first section in the report which include the title of the report and the date range(the start date and the end date of the input).

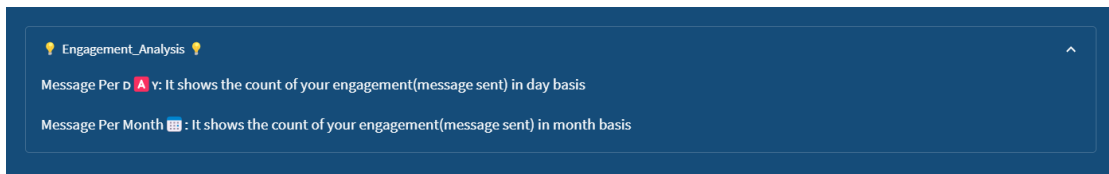


*Figure 4.7.3(a) section 2 of the report*



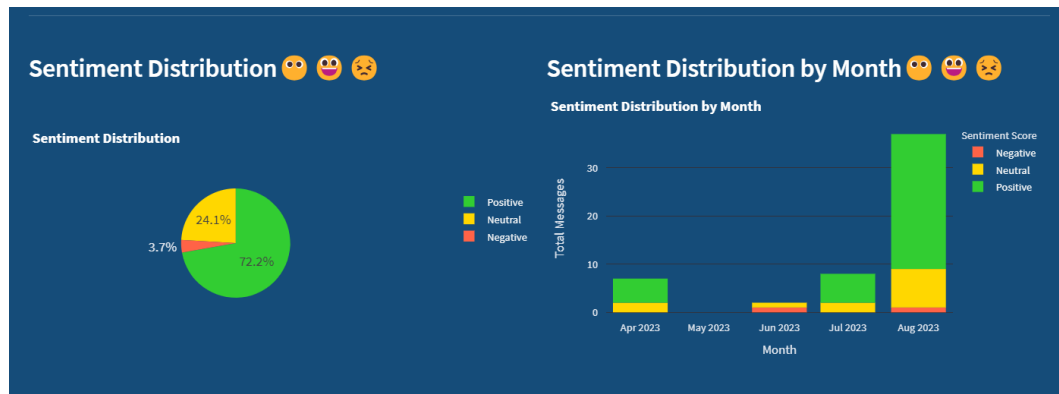
*Figure 4.7.3(b) section 2 of the report*

The first component included in the report is the metric related to the activity of the user. The entry "Total Messages: 54" signifies that the user has sent a total of 54 messages. The "Message per Day" graph illustrates the message count on a daily basis, whereas the "Message per Month" graph represents message count on a monthly basis. Hovering over the charts reveals the corresponding dates and their respective message counts. This chart helps users understand how frequently they engage with the chatbot over time, providing a sense of their daily commitment to practicing English.



**Figure 4.7.3(c) section 2 of the report**

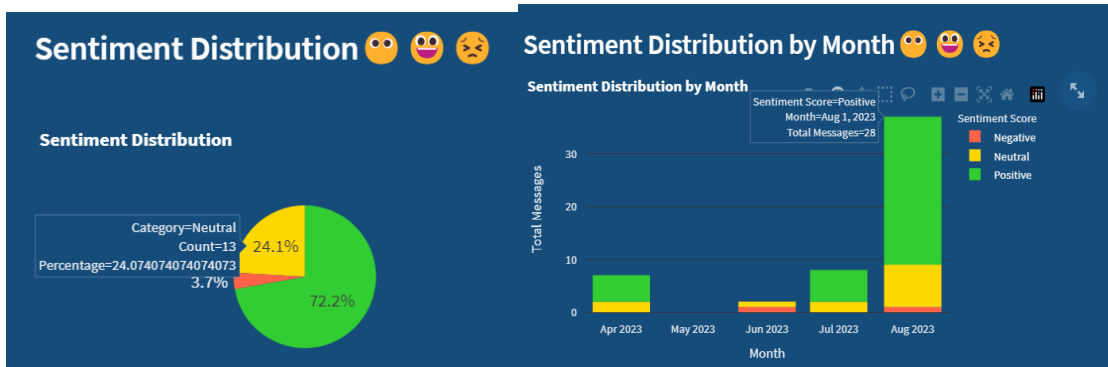
An expander also prepared which contains the information regarding the chart.



**Figure 4.7.4(a) section 3 of the report**

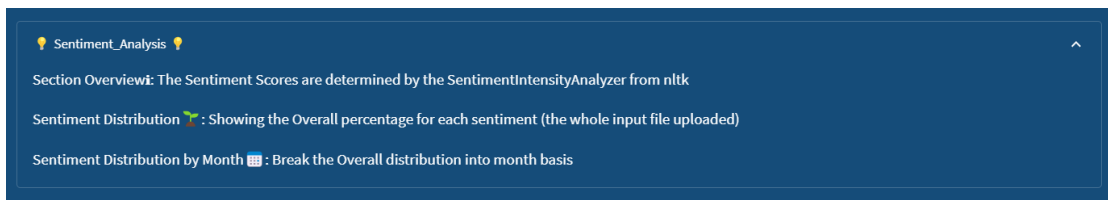
The sentiment distribution analysis is visualized through a pie chart, showcasing the overall distribution of sentiments within the specified date range. On the right-hand side, a nested bar chart provides a breakdown of sentiment distribution on a monthly basis. This component aims to provide users with insights into their ability to convey different sentiments, highlighting both the highest and lowest points of sentiment expression.

In addition to offering an overview of sentiment distribution, this component serves as a valuable tool for users to gauge their emotional tone throughout their interactions. By observing the sentiment distribution patterns, users can identify trends in their communication, such as periods of heightened positivity, moments of contemplation, or instances of more neutral expression. This analysis empowers users to reflect on their communication style and its evolution over time, potentially leading to more mindful and effective interactions in the future.



**Figure 4.7.4(b) section 3 of the report**

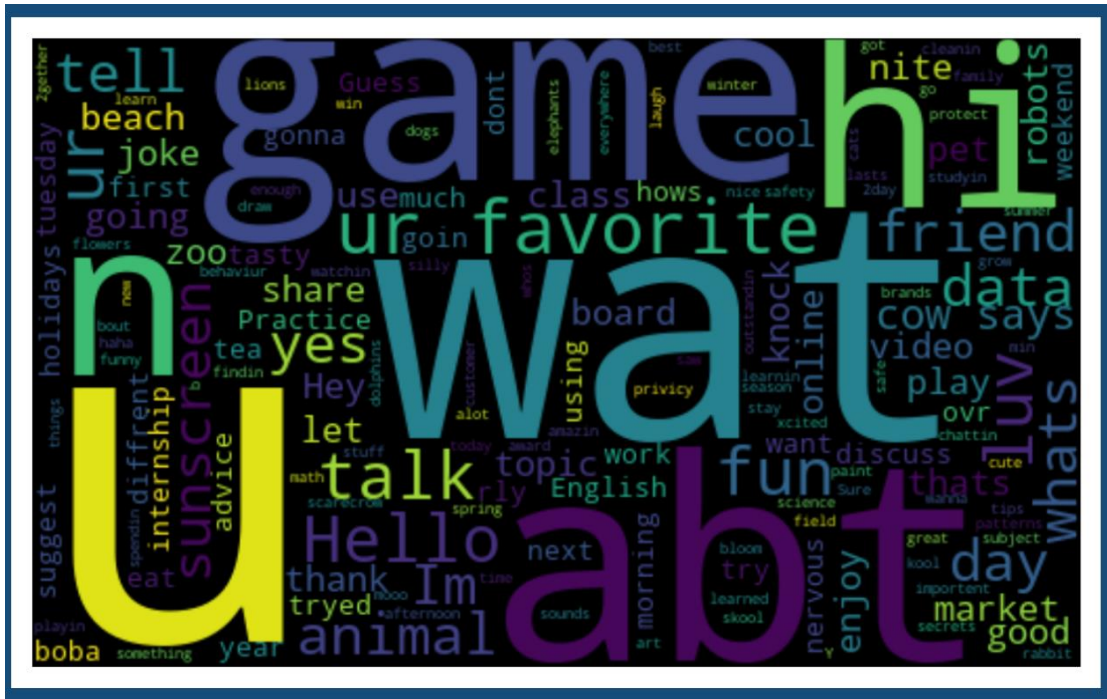
Hovering over the charts reveals the corresponding categories and their respective message counts, and percentage.



**Figure 4.7.4(c) section 3 of the report**

An expender also prepared which contains the information regarding the chart.





*Figure 4.7.5 (a) section 4 of the report*

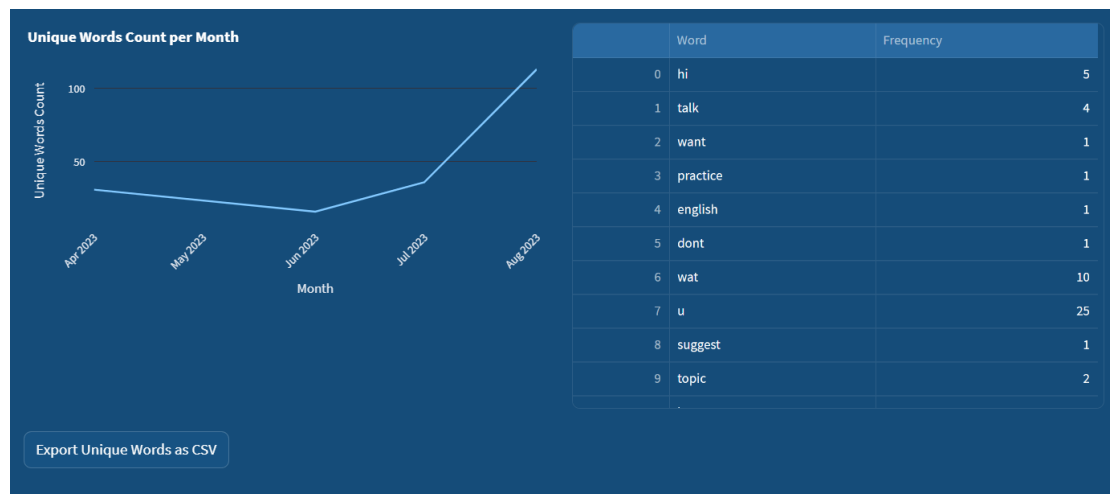
Building a strong vocabulary is a key aspect of language learning, and it's an essential component covered in the English Practice Chatbot Report. This segment of the report focuses on the words that learners use in their messages. It's often said that the more words you know, the better you can express yourself. This vocabulary analysis provides valuable insights into the learner's language development journey.

The vocabulary component presents an intriguing visualization known as a "word cloud." In this visual representation, each word is displayed in a size that corresponds to its frequency of use. Words that are used more often appear larger, while less frequently used words are smaller. For example, based on the figure shown above, the game is used the most by the user. This simple yet powerful visual allows learners to quickly identify the words they tend to use most frequently when communicating.

By examining this word cloud, learners can gain a clearer understanding of their language patterns. The larger words represent their favored vocabulary choices, giving them an immediate sense of the words, they rely on to convey their thoughts. This visualization serves as a reflective tool, helping learners become more aware of their language strengths and areas where they might want to diversify their vocabulary. As an example, an intriguing insight that can be derived from the report is the user's preference for expressing themselves through the utilization of short forms rather than



writing out complete words. This preference is evident in instances where the user employs abbreviations like "wat" instead of "what" and "abt" instead of "about."

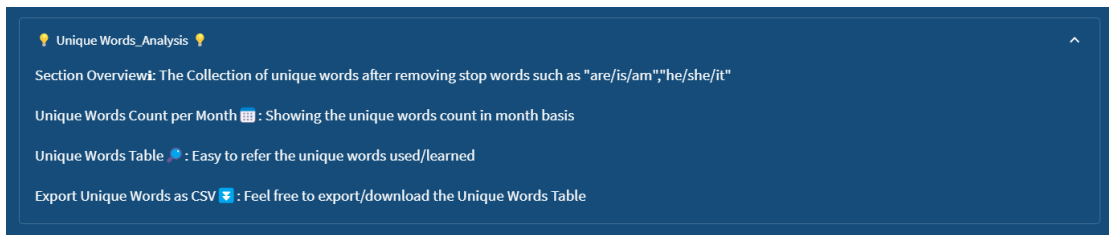


**Figure 4.7.5 (b) section 4 of the report**

Moreover, the English Practice Chatbot Report offers users a comprehensive insight into their vocabulary growth through various visual and interactive features. One such component is the "Unique Word Count," which is presented using a line chart based on a monthly breakdown. This chart provides users with a clear understanding of how many distinct words they've used in each month, offering a valuable snapshot of their linguistic progress over time.

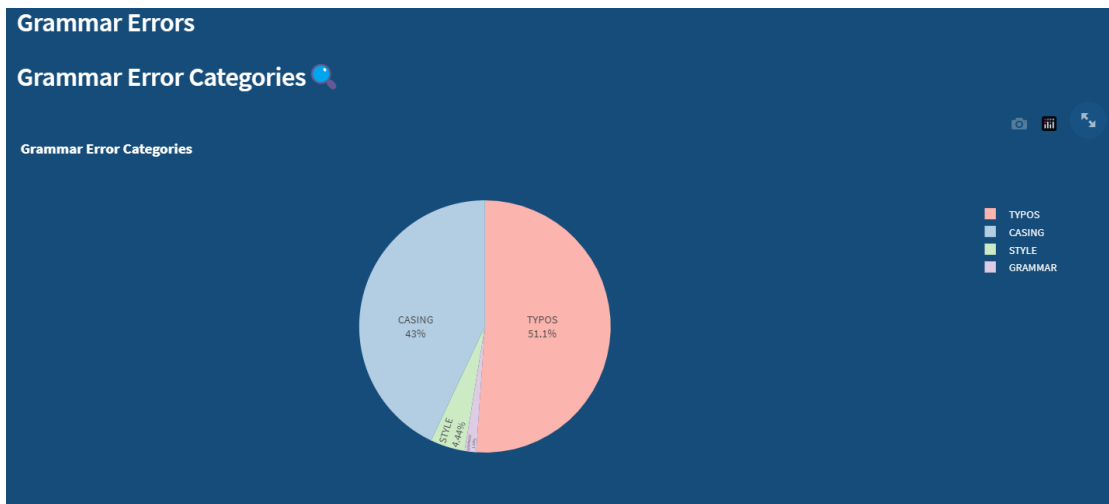
In addition to the visual representation, the report includes a tabulated summary of all the unique words utilized by the user. This table conveniently displays both the words and their corresponding frequencies. What sets this table apart is its user-friendly functionality. Users can effortlessly arrange the words and frequencies in ascending or descending order by simply clicking on the "Word" or "Frequency" header of the table.

Recognizing the significance of vocabulary acquisition in the journey of mastering English, the report goes a step further to ensure users can make the most of their unique words. To facilitate this, the report equips users with the option to export the list of unique words. With a straightforward click on the "Export Unique Words as CSV" button, users can generate a downloadable CSV file containing their unique word inventory.



**Figure 4.7.5 (c) section 4 of the report**

An expender also prepared which contains the information regarding the section and charts.



**Figure 4.7.6 (a) section 5 of the report**

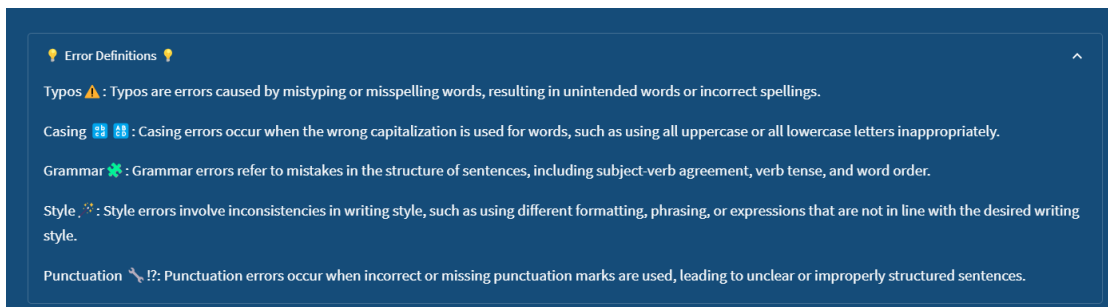
The subsequent component of the report focuses on identifying and categorizing grammar errors into four distinct groups: typos, casing, style, and grammar. Analyzing the data obtained from this assessment reveals that the user predominantly commits errors in the categories of typos and casing. This tendency can be attributed to the user's frequent usage of abbreviations and their relatively casual approach to capitalization conventions while communicating.

The prevalence of typos, which are minor errors resulting from mistyping characters, might be linked to the user's propensity for quick and informal communication. The use of short forms and the hurried nature of digital conversations can contribute to inadvertent typographical errors that, while small, can affect the clarity of the user's messages.

Similarly, the consistent occurrence of casing errors, where the user neglects proper capitalization rules, aligns with their habit of informal language use. The user's

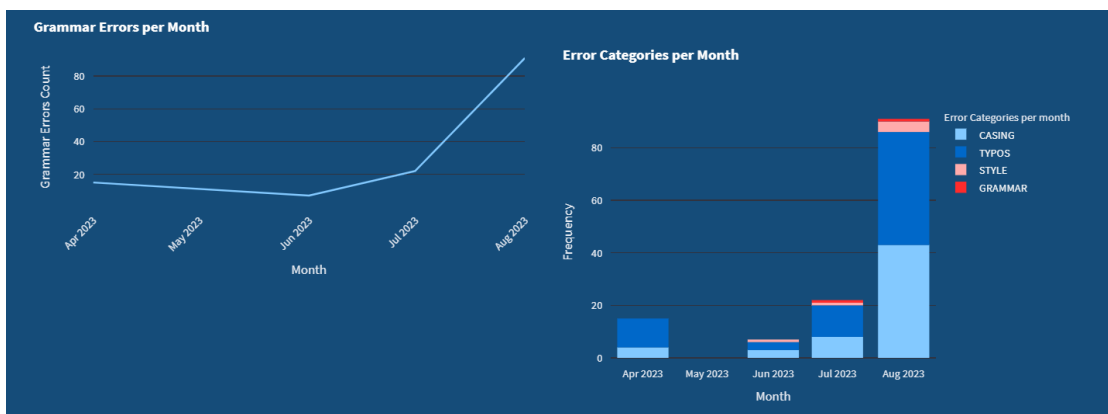
communication style may prioritize speed and ease of expression over formal conventions, resulting in varying capitalization patterns.

Interestingly, the occurrence of grammar errors remains relatively minimal. This could be attributed to the user's tendency to construct concise and straightforward sentences. Shorter sentences are less likely to involve complex grammatical structures, thus reducing the likelihood of substantial grammatical mistakes.



**Figure 4.7.6 (b) section 5 of the report**

The explanation for each error category is also prepared for user to refer.



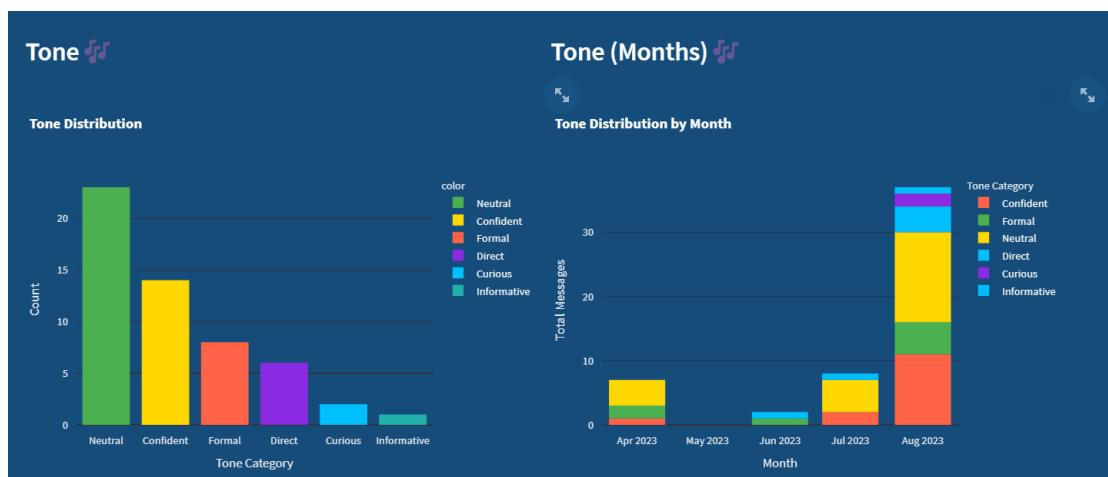
**Figure 4.7.6 (c) section 5 of the report**

Furthermore, the report provides a breakdown of the error counts on a monthly basis, which is visualized to the user through both a line chart and a nested bar chart. These two charts offer a comprehensive snapshot of the user's error trends throughout each month, highlighting not only the overall frequency of errors but also the specific error categories that were prevalent.

The line chart presents a clear trajectory of the error count over the months, allowing the user to quickly discern any patterns or fluctuations in their error-making behaviour. This visual representation is particularly helpful in identifying whether there are months with higher error counts.

Additionally, the nested bar chart offers a more detailed breakdown of the error categories within each month. This chart showcases the proportion of typos, casing errors, style errors, and grammar errors for each month. By using distinct colors for each error category, the chart simplifies the comparison and evaluation of the error distribution over time. Users can easily identify whether a specific error category was consistently more prominent across several months or if certain months exhibited a higher prevalence of a particular type of error.

The combination of these two visualizations allows users to swiftly grasp their error patterns on a month-to-month basis, gaining insights into both the general error count trends and the specific error categories that require attention.



**Figure 4.7.7 (a) section 6 of the report**

The final component of the report focuses on Tone distribution, a critical aspect for language learners, particularly when communicating through written mediums like texting or emailing. Through extensive research, it has been established that mastering different tones is essential to effective communication. This component categorizes tone into six distinct groups: confident, formal, neutral, direct, curious, and informative. These categories are referred from the weekly report from Grammarly.

The report begins by providing an overview of the overall tone distribution, showcasing the frequency of each tone category in a concise bar chart. This representation offers a quick visual insight into the user's predominant tone usage and helps users understand how their communication style is distributed among the various tones. Additionally, the report delves deeper by presenting a nested bar chart that

categorizes the tone distribution by month. This visualization highlights how the user's tone usage fluctuates over time and across different months. Each tone category is color-coded for easy differentiation, enabling users to readily identify the dominant tone in each month.

Detected Tones and Message	
Message	Detected Tone
0 hil can we talk? i want to Practice English	Neutral
1 I dont no wat to talk about can u suggest a topic	Formal
2 let's discuss diffrent holidays!	Neutral
3 Hello! How's ur day going?	Neutral
4 Im good, thank u. Have u tried boba tea before?	Confident
5 It's rly tasty! U should try it at the nite market	Formal
6 wat do u like 2 eat the nite market?	Neutral
7 Guess wat? I'm gonna have an internship next year! any advice?	Formal
8 tell me abt ur first day work! were u nervous?	Direct
9 good morning! hows ur tuesday going?	Confident

**Figure 4.7.7 (b) section 6 of the report**

Furthermore, each message is accompanied by its respective tone category, ensuring that users have a comprehensive reference point for understanding the nuances of their communication. This aspect of the report provides users with a clear breakdown of how their messages are perceived in terms of tone. For instance, if a user notices that their messages are frequently categorized as "informative," they can acknowledge their strength in providing clear explanations and sharing knowledge.

Tone_Analysis
Section Overview: The message tone has been labeled using TextBlob from python"
Formal 📄: Professional and respectful communication using proper language
Neutral 😊: Unbiased and factual communication without strong emotions
Confident 😊: Communication displaying certainty and self-assuredness
Direct 🗣️: Concise and straightforward communication
Informative 📄: Communication focused on providing knowledge
Casual 😊: Relaxed and informal communication
Assertive 🗣️: Confidently expressing opinions or requests

**Figure 4.7.7 (c) section 6 of the report**

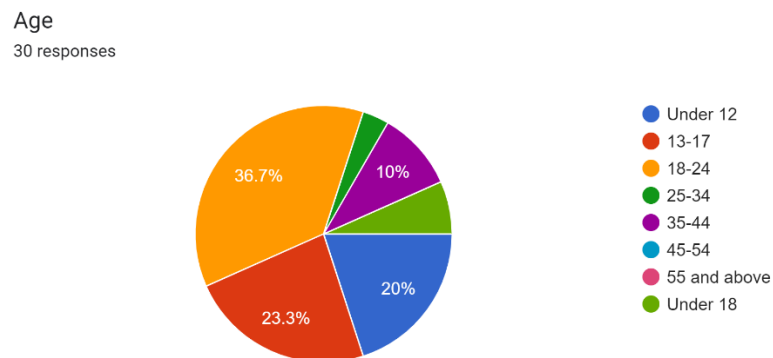
An expender also prepared which contains the information regarding the section and charts.

## 4.8 Deployment

This English Chatbot project has been successfully deployed using Vercel, a frontend cloud platform that equips teams with robust tools for the secure development, deployment, and scaling of exceptional web applications, regardless of the chosen framework. [80] However, the English Practice Chatbot Report project encountered deployment issues primarily because of the Java requirement for Language\_tool which is the library used to conduct grammar analysis. Nevertheless, it has been made to access locally, enabling users to access it if they are connected to the same network as the project owner.

## 4.9 User Testing

As an integral part of the project's development and enhancement journey, an extensive phase of user testing was executed involving a total of 30 participants who actively engaged with the project. To meticulously collect feedback, a Google Form was utilized as the tool for gathering insights. This testing initiative proved instrumental in acquiring valuable perspectives on user interactions and perceptions regarding the project's operational capabilities.

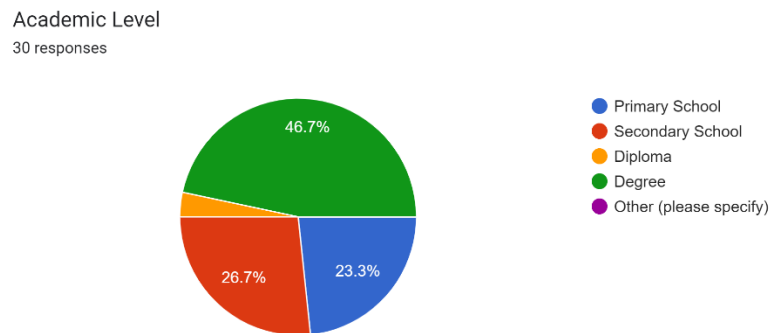


*Figure 4.9.1 Survey Question 1*

The participants encompass a wide spectrum of age groups, ranging from individuals below the age of 12 to those as old as 44. Particularly noteworthy is the significant

## CHAPTER 4

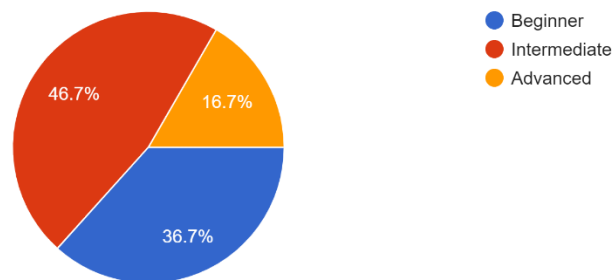
portion of participants falling within the under 17 age bracket, predominantly consisting of students enrolled in primary and secondary educational institutions.



*Figure 4.9.2 Survey Question 2*

In conjunction with age, the educational backgrounds of the participants serve as an additional layer of context to the user demographic. The participants hail from diverse academic levels, encompassing primary school, secondary school, and degree programs. This diversity in educational stages underscores that the English Practice Chatbot has succeeded in capturing interest across a spectrum of learning environments, ranging from the formative years of primary education to the advanced pursuit of higher education at the degree level.

What is your English language proficiency level?  
30 responses



*Figure 4.9.3 Survey Question 3*

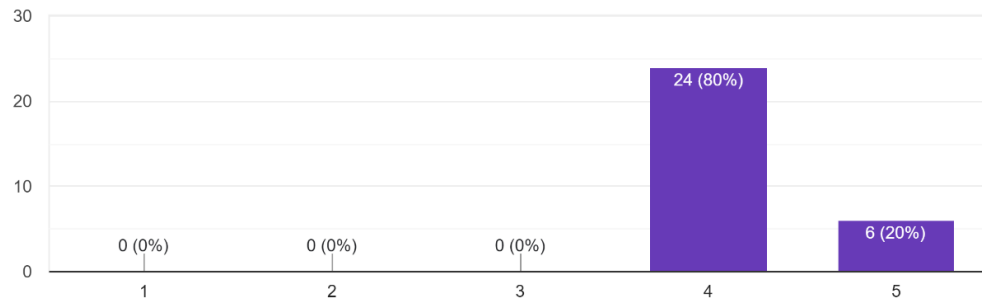
Another crucial aspect to consider is the language proficiency in English demonstrated by the testers. The majority of participants exhibit an intermediate level of English proficiency, with beginners and those at an advanced level forming the subsequent tiers.

## CHAPTER 4

Additionally, a discernible trend emerges from the analysis: an increase in age is correlated with a higher level of proficiency. This observation aligns logically, underlining the idea that as age advances, language skills tend to progress as well.

On a scale of 1 to 5, how user-friendly did you find the chatbot interface?

30 responses

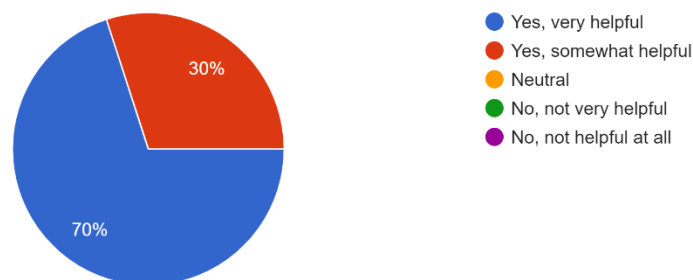


*Figure 4.9.4 Survey Question 4*

Moving forward, our attention transitions to the queries associated with the online platform, initiating our investigation with an examination of the interface's user-friendliness. Within the cohort of 30 testers, a substantial 24 participants bestowed an admirable rating of 4 out of 5, signaling their optimistic endorsement of the interface's navigability and ease of use. Furthermore, a distinct subgroup of 6 testers extended the highest possible rating of 5 out of 5, magnifying the design's inherent intuitiveness and user-centric design. Derived from this outcome, it can be inferred that the chatbot-related website's straightforward and uncomplicated interface resonates well with users, affirming its acceptance among the tester community.

Did you find the real-time grammar checking helpful for improving your language skills?

30 responses

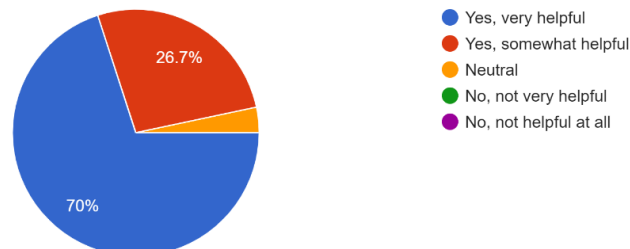




*Figure 4.9.5 Survey Question 5*

Moving forward, the subsequent question delves into the utility of the real-time grammar checking feature. A unanimous consensus emerges among all 30 testers, affirming that the real-time grammar checking significantly contributes to refining their language capabilities. Within this unanimous agreement, a notable 70% of participants, primarily situated within the beginner level of English language proficiency, express strong enthusiasm for the feature, perceiving it as highly advantageous. Meanwhile, the remaining 30%, predominantly distributed across the intermediate and advanced levels of English language proficiency, regard the feature as moderately beneficial. From this amalgamation of perspectives, a compelling deduction can be drawn: the real-time grammar checker holds inherent value in providing guidance to users, particularly as they hone their English skills through interactive conversations with the chatbot. This analysis underscores the feature's pivotal role in accommodating learners across various stages of language proficiency.

Did you find the paraphrasing feature helpful for improving your language skills?  
30 responses

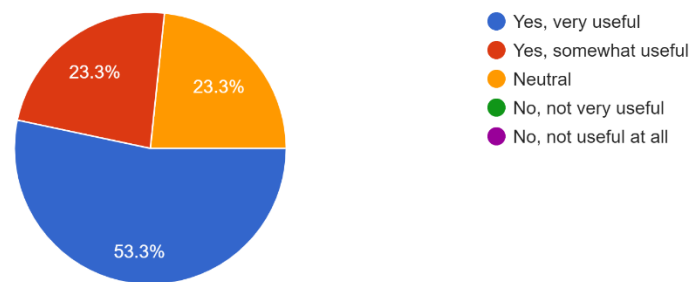
*Figure 4.9.6 Survey Question 6*

Moving on to the subsequent topic, we delve into the evaluation of the paraphrasing feature. Impressively, a substantial 70% of respondents, primarily stemming from the beginner and intermediate English proficiency levels, laud the feature as remarkably beneficial. Further adding to this tapestry of insights, approximately 26.7% of testers, primarily positioned within the advanced English proficiency level, view the feature as moderately advantageous. In a minority representation, 3.3% of testers from the advanced proficiency level remain neutral in their assessment of the paraphrasing

## CHAPTER 4

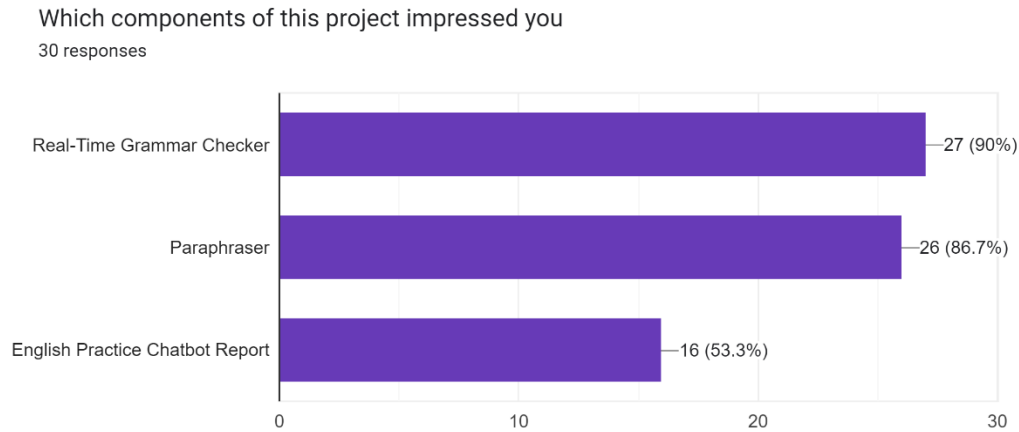
feature. This nuanced panorama offers a comprehensive snapshot of user perceptions, indicating that the paraphrasing tool is predominantly embraced by those at the early and intermediate stages of language learning.

Did you find the "English Practice Chatbot Report" useful in tracking your learning progress?  
30 responses



*Figure 4.9.7 Survey Question 7*

Furthermore, we delve into the assessment of the "English Practice Chatbot Report" and its utility. A notable 53.3% of participants regard the report as highly advantageous, underlining its value in their language learning journey. Following closely, 23.3% of testers perceive the report as moderately beneficial, acknowledging its role in enhancing their learning experience. Concurrently, another 23.3% of respondents assume a neutral stance regarding the report's usefulness. This intricate tapestry of feedback provides a multi-dimensional perspective, illustrating that over half of the testers find substantial value in the report, while others acknowledge its positive influence to varying degrees or remain in a neutral position.



*Figure 4.9.8 Survey Question 8*

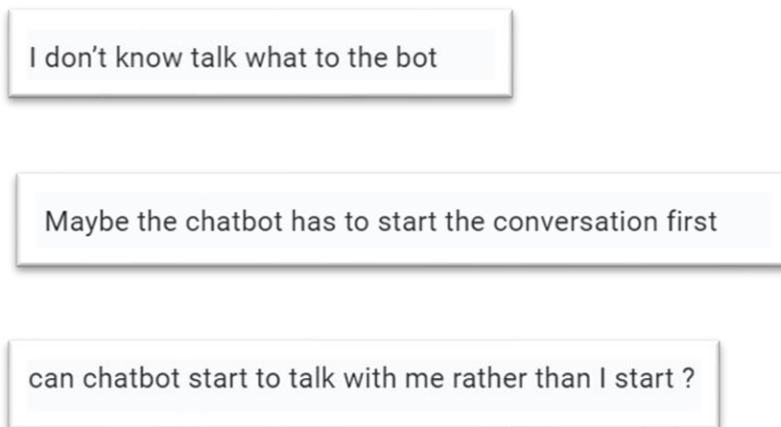
Inquiring about the elements of this project that left an impression on the testers, it's evident that the real-time grammar checker garnered the highest level of approval, with the paraphraser following closely, and the English Practice Chatbot Report securing the third position. Specifically, 27 testers were notably impressed by the real-time grammar checker's capabilities, underscoring its significance. The paraphraser, being a distinctive feature, captured the attention of 26 testers. Lastly, the English Practice Chatbot Report, which serves as a comprehensive progress tracker, made a favorable impact on 16 testers. This hierarchy of appreciation among these project components offers valuable insights into the aspects that resonated most effectively with the testers.

The final question is an open-ended inquiry, inviting testers to share any additional comments, feedback, or suggestions they might have—a response that is not obligatory. Extracted below are discernible insights from the comments received.

lose when starting a conversation with the bot

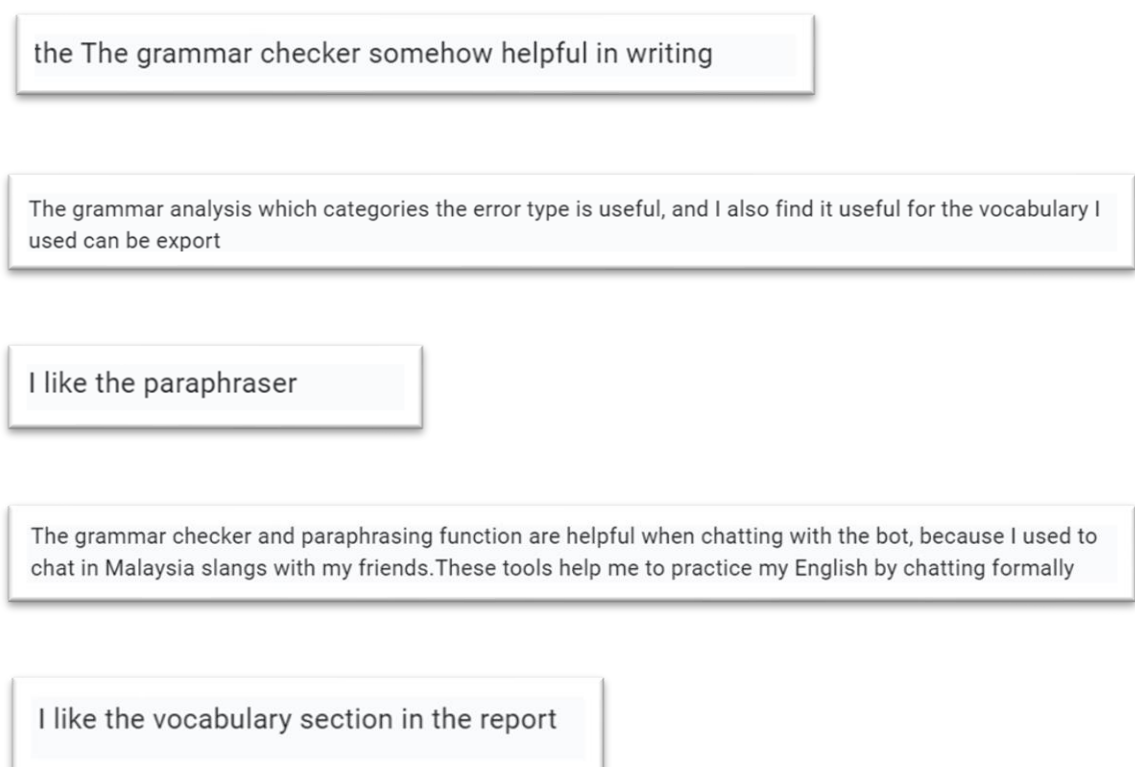
good to have the export vocabulary function, but will lose when deciding what topic to chat with the bot in the beginning

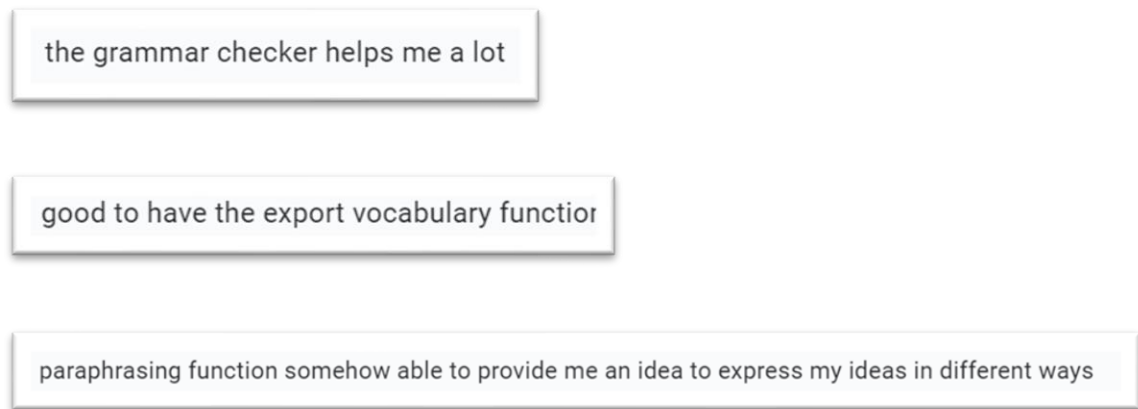
## CHAPTER 4



*Figure 4.9.9 Survey Question 9 (a)*

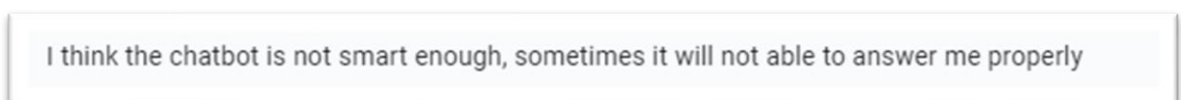
Based on the outcomes, it becomes evident that 5 testers share the sentiment that they face challenges in initiating conversations with the bot due to uncertainty about the topic or how to commence the interaction. Notably, these testers primarily align with the beginner level of English proficiency. This insight underscores the importance of providing guidance or prompts to help users, especially those at the beginner level, overcome the initial hurdle of starting a conversation, thereby enhancing their engagement and experience with the chatbot.





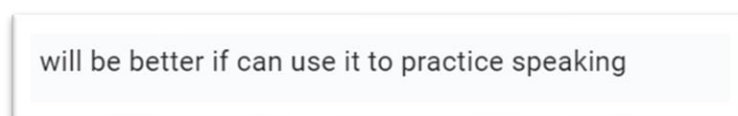
***Figure 4.9.10 Survey Question 9(b)***

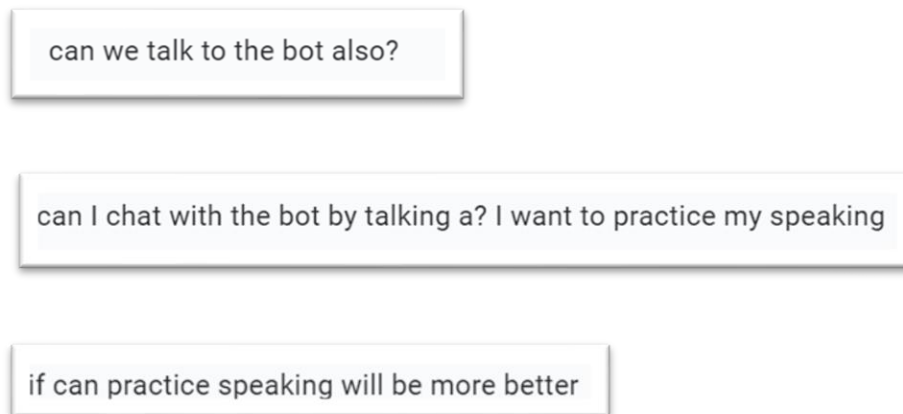
Moving forward, it's noteworthy that 8 testers have taken the initiative to provide specific comments regarding distinct components of the project, namely the Grammar Checker, Paraphraser, and English Practice Chatbot Report. Among these comments, a discernible trend emerges wherein the testers acknowledge the practicality of both the grammar checker and the paraphraser features. However, it's particularly intriguing that in the context of the report, the vocabulary and grammar sections have garnered notable attention from the testers, who evidently find them to be of significant value. This feedback provides a nuanced perspective on the components that resonate most with the testers and showcases the significance of certain aspects of the English Practice Chatbot in aiding language learning and improvement.



***Figure 4.9.11 Survey Question 9(c)***

Furthermore, one of the testers also given a comment about the respond of the chatbot, saying that the reply from the chatbot might not that relevant. This might be due to the version of Open AI used





*Figure 4.9.12 Survey Question 9(d)*

Lastly, an area that garnered attention to suggested enhancements. Notably, 4 testers express the viewpoint that the project's efficacy could be further amplified if it encompassed a feature allowing them to not only practice by engaging in text-based conversations with the bot, but also afforded them the opportunity to hone their speaking skills. This feedback underscores the potential value of integrating a speaking practice element into the project, thereby catering to a broader range of language learning objectives and creating a more comprehensive and versatile language learning experience for the users.

### 4.10 Enhancement

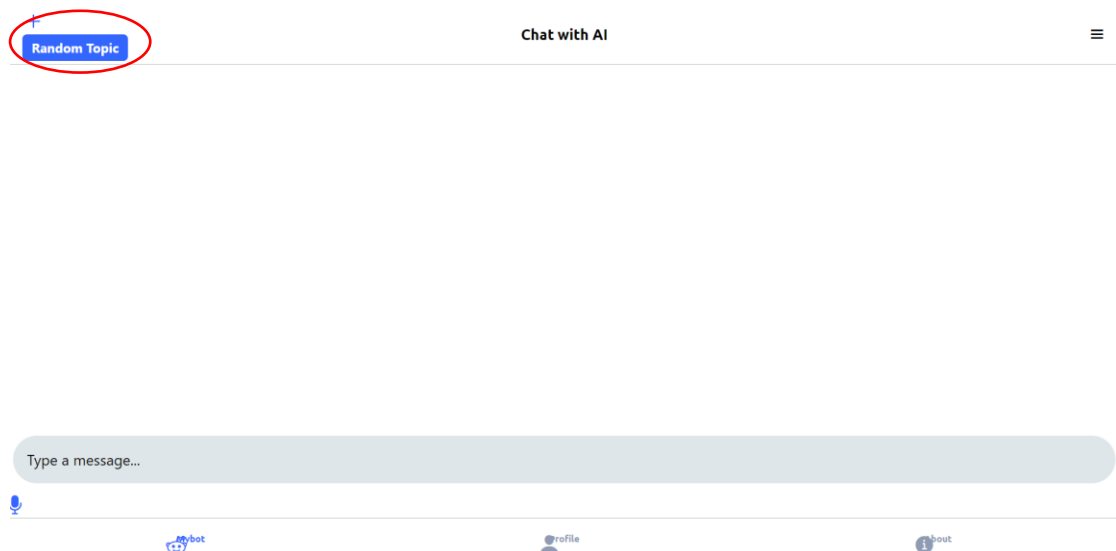
To solve the problem of initiating conversations with the chatbot a “Random Topic” button has been added.

```
const generateRandomTopic = async () => {
  const openaiData = await fetch(
    "https://api.openai.com/v1/engines/text-davinci-003/completions",
    {
      method: "POST",
      headers: {
        "Content-Type": "application/json",
        Authorization:
          "Bearer sk-ilH2kMHrFDyus5ZpeskCT3B1bkFJaNh0wNjsH9VyrPt6Sf7M",
      },
      body: JSON.stringify({
        prompt: "Generate a random topic to chat with me in sentence:",
        temperature: 0.87,
        max_tokens: 50,
        top_p: 1,
        frequency_penalty: 0,
        presence_penalty: 0,
      })),
    ).then((res) => res.json());

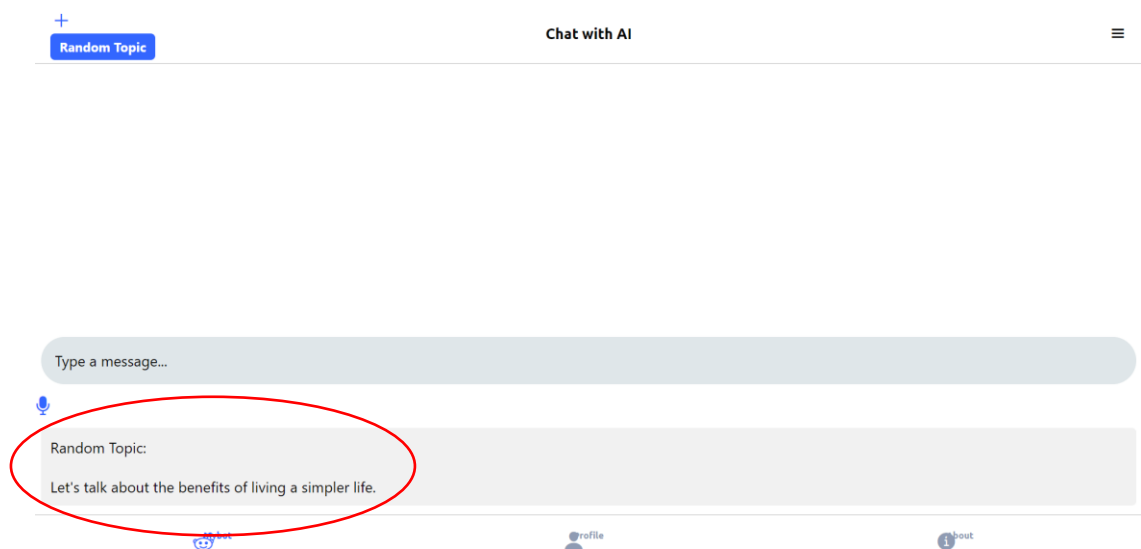
  const generatedTopic = openaiData?.choices?.[0]?.text;
  setRandomTopic(generatedTopic);
};
```

*Figure 4.10.1 generateRandomTopic()*

A function called `generateRandomTopic` is created. This function is responsible for coming up with a random topic or subject for the chat with the bot. First, this function initiates by making an HTTP request to the OpenAI API. This request is sent to a specific endpoint (URL) that's designed for generating text based on a given prompt. Within this request, we provide several parameters that instruct the AI on how to generate the random topic. These parameters include things like the language model to use (`text-davinci-003` in this case), the temperature (which controls the randomness of the response), and maximum tokens (how long the response should be). The AI is given a prompt, which is like a starter sentence. In this case, it's "Generate a random topic to chat with me in sentence:". This prompt sets the context for what we want the AI to do. The AI processes this prompt and generates a text response accordingly. This text response is essentially the random topic we're looking for. Finally, we use `setRandomTopic(generatedTopic);` to set the generated topic in our application. This means the random topic is now available for the user to see and use in their conversation with the chatbot.



**Figure 4.10.2 Random Topic Button**



**Figure 4.10.3 Random Topic Generated**

In response to this feedback, a strategic enhancement was introduced to address this usability issue. Specifically, a "Random Topic" button was thoughtfully integrated into the project's interface. This innovative feature serves as a proactive solution, offering users a prompt for potential conversation topics. Upon clicking the "Random Topic" button, users are provided with suggestive topics (Let's talk about the benefits of living a simpler life), effectively mitigating the challenge of initiating conversations and encouraging seamless engagement.



Furthermore, a **speaking function** also been added to let user input by speaking.

```
const startSpeechRecognition = async () => {
  var recognition = new WebkitSpeechRecognition();
  recognition.lang = "en-US";
  recognition.start();
  recognition.onresult = function (event) {
    var message = event.results[0][0].transcript;
    setMessage(message);
  };
};
```

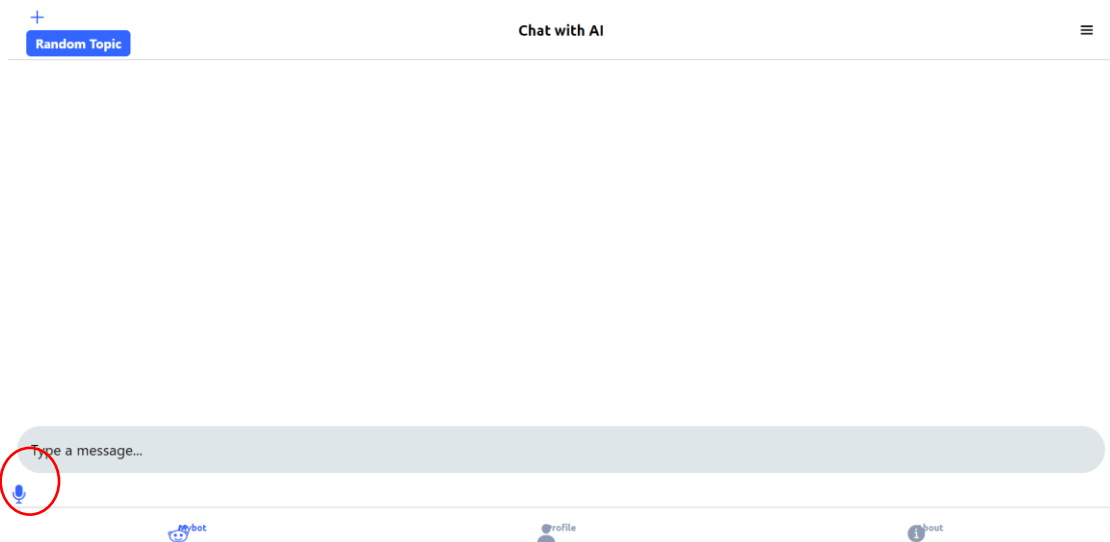
*Figure 4.10.4 startSpeechRecognition()*

In this code, an essential functionality known as "Speech Recognition" is being established through the utilization of the Web Speech API. This specification outlines a JavaScript API that empowers web developers to integrate both speech recognition and text-to-speech capabilities into their web content. It allows developers to employ scripting to produce text-to-speech responses and to utilize speech recognition as an input method for forms, continuous dictation, and controlling web applications.[81] This feature allows the application to listen to the user's voice and convert what they say into text. It is like having a conversation with the computer, where it can understand and respond to spoken words. First, we create an instance of the `webkitSpeechRecognition` object. Think of this as an attentive listener that's always ready to hear what the user says. We assign this instance to a variable called `recognition`. We specify that we want the recognition to work in English, specifically the United States variant. The line `recognition.start()`; is where the magic happens. It tells the system to start listening for the user's voice. When this line executes, the application is ready to hear what the user says. The `recognition.onresult` part is an event handler. It's like setting up a reaction for when someone says something. When the user speaks, and their speech is understood, this part of the code is triggered.

Inside this event handler:

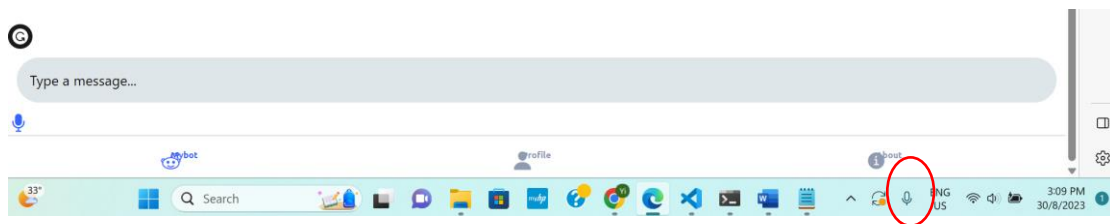
`event.results[0][0].transcript`; is where we extract what the user said in the form of text. Imagine this as the computer's way of writing down what it heard.

`setMessage(message)`; is a function that takes this text and display in the textbox.



**Figure 4.10.5 Microphone Button**

The “Microphone” button is at the bottom left of the screen.



**Figure 4.10.6 speaking mode started**

After user click on it, user can start to speak what ever they like. For example, “Hi bot, how are you?”.



**Figure 4.10.7 speaking done**

## CHAPTER 4

Then the words that they just said will appear in the text box for grammar checking.

Then user can press “enter” to send out the message.

## CHAPTER 5 Conclusion

This project aims to develop a web page that provides English Language Learners with a platform to practice their language skills by conversing with a bot. One of the key problems faced by learners is the lack of additional language input from peers who are at a similar language proficiency level. To address this issue, a real-time grammar checker has been implemented to ensure that users are aware of and learn from their mistakes. This also helps users to create error-free sentences that the bot can understand, resulting in more accurate responses and smoother conversations. Additionally, a paraphrase function has been implemented to help users explore different ways of expressing their ideas and thoughts.

Similar applications were analyzed, compared, and reviewed, resulting in two main types of language learning chatbots which are Today's Language Learning Chatbot Applications, which include Duolingo, Eggbun, Andy, and Oral Craft, and Today's Conversational AI Chatbots, which include Replika, XiaoIce, Siri, and Chat GPT. Analyzing the strengths and weaknesses of these applications helped to inform the design of this project and provide additional functions that other language learning applications may not offer.

Application Programming Interfaces (APIs) are essential components of this project, including OpenAI's API to facilitate conversations, the Grammarly Text Editor SDK for real-time grammar checking, and the paraphraser API from Smodin. Finally, the application will be tested by users who will provide feedback on the functionality, user interface, and other relevant fields to improve the application. Any technical issues discovered during the testing phase will be addressed and fixed accordingly.

To elevate the learning experience, we drew inspiration from the valuable insights provided by Grammarly's reports. This impetus culminated in the creation of a meticulously crafted "English Practice Chatbot Report." This encompassing report

## CHAPTER 5

delves into multifaceted metrics such as the frequency of daily and monthly engagement, the nuanced analysis of sentiment, the breadth of vocabulary utilization, the precision of grammar application, and the underlying conversational tone. Through the amalgamation of these metrics, this insightful report empowers users with a profound understanding of their evolving progress and developmental trajectory.

## References

- [1] A. Soeyuenmez, “What are the different types of chatbots?,” *MessengerPeople by Sinch*, Apr. 07, 2022. <https://www.messengerpeople.com/what-are-the-different-types-of-chatbots/> (accessed Nov. 05, 2022).
- [2] H. Kim, “Design principles and architecture of a second language learning chatbot,” *Lang. Learn.*, p. 18.
- [3] L. Fryer, “Bots for language learning now: Current and future directions,” *Lang. Learn.*, p. 15.
- [4] L. Fryer, K. Nakao, and A. Thompson, “Chatbot learning partners: Connecting learning experiences, interest and competence,” *Comput. Hum. Behav.*, vol. 93, pp. 279–289, Apr. 2019, doi: 10.1016/j.chb.2018.12.023.
- [5] English 100, “learn English with Robot Friend through a Learn English App. - English 100,” Aug. 18, 2022. <https://learnenglish100.com/learn-english-with-robot-friend-app/> (accessed Nov. 05, 2022).
- [6] H. WeiJiao, H. Khe Foon, and Luke K, “Chatbots for language learning—Are they really useful? A systematic review of chatbot-supported language learning,” 2021, doi: 10.1111/jcal.12610.
- [7] Q. Yin, S. Co, and M. Satar, “ENGLISH AS A FOREIGN LANGUAGE LEARNER INTERACTIONS WITH CHATBOTS: NEGOTIATION FOR MEANING,” p. 22, 2020.
- [8] A. Zahra, “Current Challenges in Teaching/Learning English for EFL Learners: The Case of Junior High School and High School | Elsevier Enhanced Reader,” Aug. 2015. <https://reader.elsevier.com/reader/sd/pii/S1877042815045358?token=476C8A6E5D0ED1B687D3BBE3DF511573AFE87FAFB354FE73366351EE7EA9EC5CD131BA9024EE66E0EC8787FE0C60F7FC&originRegion=eu-west-1&originCreation=20221106005836> (accessed Nov. 06, 2022).
- [9] R. Rusli, M. Yunus, and H. Hashim, “LOW SPEAKING PROFICIENCY AMONG THE MALAYSIAN UNDERGRADUATES: WHY AND HOW?,” p. 12, 2018.
- [10] R. Darmi and P. Albion, “MALAYSIAN GRADUATES: WHAT AND WHY,” p. 7.
- [11] J. Tu, “Learn to Speak Like A Native: AI-powered Chatbot Simulating Natural Conversation for Language Tutoring,” *J. Phys. Conf. Ser.*, vol. 1693, no. 1, p. 012216, Dec. 2020, doi: 10.1088/1742-6596/1693/1/012216.
- [12] V. D’Silva, “Challenges in students’ mastery of English,” *NST Online*, Oct. 19, 2019. <https://www.nst.com.my/news/nation/2019/10/531486/challenges-students-mastery-english> (accessed Nov. 06, 2022).
- [13] Anil, “4 Reasons You Can Read, Write, and Listen English, but Can’t Speak,” *Lemon Grad*, Mar. 15, 2018. <https://lemongrad.com/can-read-write-and-listen-english-but-cant-speak/> (accessed Nov. 06, 2022).
- [14] E. W. M. Hopman and M. C. MacDonald, “Production Practice During Language Learning Improves Comprehension,” *Psychol. Sci.*, vol. 29, no. 6, pp. 961–971, Jun. 2018, doi: 10.1177/0956797618754486.
- [15] M. O. Rivera and G. K. McKeithan, “Progress Monitoring of Language Acquisition and Academic Content for English Learners,” *Learn. Disabil. Res. Pract.*, vol. 37, no. 3, pp. 216–225, 2022, doi: 10.1111/ldrp.12290.
- [16] D. Hadiyah, “Introducing Grammarly Insights,” *Introducing Grammarly Insights / Grammarly Blog*, May 28, 2019. <https://www.grammarly.com/blog/introducing-grammarly-insights/> (accessed Aug. 18, 2023).
- [17] OCI, “What is a Chatbot?” <https://www.oracle.com/my/chatbots/what-is-a-chatbot/> (accessed Nov. 26, 2022).
- [18] L. Grace, “7 Real-life Chatbot Use Cases (in Different Industries!),” *Dialpad*. <https://www.dialpad.com/blog/chatbot-use-cases/> (accessed Nov. 26, 2022).

## REFERENCES

- [19] T. Engati, “How are intelligent healthcare chatbots being used? [New Uses for 2022],” *Engati*. <https://www.engati.com/blog/chatbots-for-healthcare> (accessed Nov. 26, 2022).
- [20] T. Shulgina and G. Sagarin, “A Survey of English Language Teaching in Higher Institutions of Learning in Kuala Lumpur Malaysia,” *SOCIAL SCIENCES*, preprint, Nov. 2017. doi: 10.20944/preprints201711.0013.v1.
- [21] P. Hendriks, *Asymmetries between Language Production and Comprehension*, 2014th edition. Dordrecht ; New York: Springer, 2013.
- [22] K. Mageira, D. Pittou, A. Papasalouros, K. Kotis, P. Zangogianni, and A. Daradoumis, “Educational AI Chatbots for Content and Language Integrated Learning,” *Appl. Sci.*, vol. 12, no. 7, Art. no. 7, Jan. 2022, doi: 10.3390/app12073239.
- [23] 신동광, “Exploring the Use of An Artificial Intelligence Chatbot as Second Language Conversation Partners,” *Korean J. Engl. Lang. Linguist.*, Jan. 2021, Accessed: Nov. 11, 2022. [Online]. Available: [https://www.academia.edu/49330330/Exploring\\_the\\_Use\\_of\\_An\\_Artificial\\_Intelligence\\_Chatbot\\_as\\_Second\\_Language\\_Conversation\\_Partners](https://www.academia.edu/49330330/Exploring_the_Use_of_An_Artificial_Intelligence_Chatbot_as_Second_Language_Conversation_Partners)
- [24] J. Jia, “CSIEC: A computer assisted English learning chatbot based on textual knowledge and reasoning,” *Knowl.-Based Syst.*, vol. 22, no. 4, pp. 249–255, May 2009, doi: 10.1016/j.knosys.2008.09.001.
- [25] L. Fryer and R. Carpenter, “Bots as Language Learning Tools,” *Lang. Learn.*, p. 7.
- [26] N. Haristiani, “Artificial Intelligence (AI) Chatbot as Language Learning Medium: An inquiry,” *J. Phys. Conf. Ser.*, vol. 1387, p. 012020, Nov. 2019, doi: 10.1088/1742-6596/1387/1/012020.
- [27] Y. Goda, M. Yamada, H. Matsukawa, K. Hata, and S. Yasunami, “Conversation with a Chatbot before an Online EFL Group Discussion and the Effects on Critical Thinking,” *J. Inf. Syst. Educ.*, vol. 13, no. 1, pp. 1–7, 2014, doi: 10.12937/ejsise.13.1.
- [28] S. D. Krashen, *Second language acquisition and second language learning*, Reprinted. in Language teaching methodology series. Oxford: Pergamon Pr, 1985.
- [29] Y. Fei and S. Petrina, “Using Learning Analytics to Understand the Design of an Intelligent Language Tutor – Chatbot Lucy,” *Int. J. Adv. Comput. Sci. Appl.*, vol. 4, Dec. 2013, doi: 10.14569/IJACSA.2013.041117.
- [30] Y. Wang, “Designing chatbot interfaces for language learning : ethnographic research into affect and users’ experiences,” University of British Columbia, 2008. doi: 10.14288/1.0066775.
- [31] L. K. Fryer, M. Ainley, A. Thompson, A. Gibson, and Z. Sherlock, “Stimulating and sustaining interest in a language course: An experimental comparison of Chatbot and Human task partners,” *Comput. Hum. Behav.*, vol. 75, pp. 461–468, Oct. 2017, doi: 10.1016/j.chb.2017.05.045.
- [32] D. Coniam, “Evaluating the language resources of chatbots for their potential in English as a second language,” *ReCALL*, vol. 20, no. 1, pp. 98–116, Jan. 2008, doi: 10.1017/S0958344008000815.
- [33] G. Vikash, “2022 Duolingo Review - Is it Worth or Total Waste of time?,” Jan. 01, 2023. <https://www.studyfrenchspanish.com/duolingo-review/> (accessed Nov. 10, 2022).
- [34] D. Jill, “Duolingo Review,” *PCMAG*, Apr. 06, 2023. <https://www.pcmag.com/reviews/duolingo> (accessed Nov. 10, 2022).
- [35] M. AGNIESZKA, “Duolingo Review: Can You Become Fluent with the Popular Language Learning App?,” *Fluent in 3 Months*. <https://www.fluentin3months.com/reviews/duolingo/> (accessed Nov. 10, 2022).
- [36] S. Kennedy, “A Review of Eggbun | A Korean Language Learning App,” *Eurolinguiste*, Aug. 29, 2016. <http://eurolinguiste.com/review-eggbun-korean-language-learning-app/> (accessed Nov. 10, 2022).
- [37] S. Juliana, “Eggbun Review: The Best App for Learning Korean - Best of Korea,” Jul. 01, 2020. <https://bestofkorea.com/the-best-app-for-learning-korean/> (accessed Nov. 10, 2022).
- [38] renglish, “English Speaking Bot Application Review – English 1,” Jul. 06, 2020. <https://readenglish1.com/andy-english-speaking-bot-application/> (accessed Nov. 10, 2022).

## REFERENCES

- [39] N. Shi, Q. Zeng, and R. Lee, "The design and implementation of Language Learning Chatbot with XAI using Ontology and Transfer Learning," arXiv, arXiv:2009.13984, Sep. 2020. Accessed: Nov. 26, 2022. [Online]. Available: <http://arxiv.org/abs/2009.13984>
- [40] Admin, "What You Need to Know About...Replika," *Ineqe Safeguarding Group*, Jan. 20, 2022. <https://ineqe.com/2022/01/20/replika-ai-friend/> (accessed Nov. 10, 2022).
- [41] C. Juli, "Siri: Everything You Need to Know," *MacRumors*. <https://www.macrumors.com/guide/siri/> (accessed Dec. 03, 2022).
- [42] Charlie, "How does Siri work What technologies are used," *Edureka Community*. <https://www.edureka.co/community/33568/how-does-siri-work-what-technologies-are-used> (accessed Dec. 03, 2022).
- [43] J. Fedewa, "What Is ChatGPT, and Why Is It Important?," *How-To Geek*, Feb. 08, 2023. <https://www.howtogeek.com/871071/what-is-chatgpt/> (accessed Mar. 31, 2023).
- [44] L. Fontaine, *Analyzing English Grammar: A systemic functional introduction*. 2012.
- [45] M. A. K. Halliday, R. Hasan, and F. Christie, *Language, Context, and Text: Aspects of Language in a Social-Semiotic Perspective*, 2nd ed. edition. Oxford: Oxford University Press, 1989.
- [46] J. Alden, "Advancing Learning: Empowering students with paraphrasing strategies," *Onestopenglish*. <https://www.onestopenglish.com/professional-development/advancing-learning-empowering-students-with-paraphrasing-strategies/557201.article> (accessed Apr. 17, 2023).
- [47] K. Amirah, "3 Benefits of Paraphrasing: The Skill for Learning, Writing and Communicating," Mar. 2022. [https://www.genei.io/blog/3-benefits-of-paraphrasing-the-skill-for-learning-writing-and-communicating?genei\\_segment\\_id=5bfddd45-c9da-4fbd-84b3-ea1e49ecfe81](https://www.genei.io/blog/3-benefits-of-paraphrasing-the-skill-for-learning-writing-and-communicating?genei_segment_id=5bfddd45-c9da-4fbd-84b3-ea1e49ecfe81) (accessed Apr. 17, 2023).
- [48] S. Sandeep, "What Is Paraphrasing? How Does It Improve Student's Academic Writing Skills?," *Examsbook*, Aug. 2022. <https://www.examsbook.com/what-is-paraphrasing-how-does-it-improve-student-s-academic-writing-skills> (accessed Apr. 17, 2023).
- [49] S. Grammarly, "Common questions about weekly Grammarly Insights reports," *Grammarly Support*, 2023. <https://support.grammarly.com/hc/en-us/articles/115000090892-Common-questions-about-weekly-Grammarly-Insights-reports> (accessed Aug. 18, 2023).
- [50] B. Nastachowski, "Academic Guides: Scholarly Voice: Tone and Audience." <https://academicguides.waldenu.edu/formandstyle/writing/scholarlyvoice/toneandaudience> (accessed Aug. 18, 2023).
- [51] T. B. PhD, "For Effective Communication Tone (Surprise!) Matters Most: 5 Strategies For Effectiveness," *Forbes*. <https://www.forbes.com/sites/tracybrower/2023/02/26/for-effective-communication-tone-surprise-matters-most-5-strategies-for-effectiveness/> (accessed Aug. 18, 2023).
- [52] M. Mohammed H., "CHALLENGES OF LEARNING ENGLISH AS A FOREIGN LANGUAGE (EFL) BY NON-NATIVE LEARNERS," *Int. J. Soc. Sci. Econ. Res.*, 2018.
- [53] J. Jia, "A Case Study of AI Application on Language Instruction: CSIEC," p. 8.
- [54] N. Nuraeni, M. Selatan, and U. Muslim, "Problems Encountered by Learners and How to Deal with Them in Learning English as A Foreign Language," pp. 21–22, May 2021.
- [55] A. Fajrin, "QUESTION ANSWERING SYSTEM BERBASIS ARTIFICIAL INTELLIGENCE MARKUP LANGUAGE SEBAGAI MEDIA INFORMASI | Azwary | KLIK - KUMPULAN JURNAL ILMU KOMPUTER," Jan. 11, 2016. <http://klik.ulm.ac.id/index.php/klik/article/view/34> (accessed Nov. 11, 2022).
- [56] L. Mike, "Learning Italian via mobile SMS | Request PDF," Aug. 2005. [https://www.researchgate.net/publication/29451560\\_Learning\\_Italian\\_via\\_mobile\\_SMS](https://www.researchgate.net/publication/29451560_Learning_Italian_via_mobile_SMS) (accessed Nov. 11, 2022).
- [57] B. Smith, "Methodological Hurdles in Capturing CMC Data: The Case of the Missing Self-Repair," *Lang. Learn.*, p. 19.
- [58] "Overview | Grammarly for Developers." <https://developer.grammarly.com/> (accessed Apr. 11, 2023).



## REFERENCES

- [59] “Smodin - Text Rewriter, Paraphraser, & Spinner API,” *RapidAPI*. <https://rapidapi.com/smodin/api/rewriter-paraphraser-text-changer-multi-language/details> (accessed Apr. 13, 2023).
- [60] S. Loria, “textblob: Simple, Pythonic text processing. Sentiment analysis, part-of-speech tagging, noun phrase parsing, and more.” Accessed: Aug. 21, 2023. [Online]. Available: <https://github.com/sloria/TextBlob>
- [61] J. Morris, “language-tool-python: Checks grammar using LanguageTool.” Accessed: Aug. 21, 2023. [Online]. Available: [https://github.com/jxmorris12/language\\_tool\\_python](https://github.com/jxmorris12/language_tool_python)
- [62] S. Bandgar, “Sentiment Analysis using TextBlob,” *Analytics Vidhya*, Jun. 11, 2021. <https://medium.com/analytics-vidhya/sentiment-analysis-using-textblob-ecaaf0373dff> (accessed Sep. 06, 2023).
- [63] P. Shah, “My Absolute Go-To for Sentiment Analysis — TextBlob.” *Medium*, Nov. 06, 2020. <https://towardsdatascience.com/my-absolute-go-to-for-sentiment-analysis-textblob-3ac3a11d524> (accessed Sep. 06, 2023).
- [64] React Native, “Setting up the development environment · React Native,” Jun. 21, 2023. <https://reactnative.dev/docs/environment-setup> (accessed Aug. 20, 2023).
- [65] flaviocopes, “Introduction to Node.js,” *Introduction to Node.js*, 2023. <https://nodejs.dev/en/learn/> (accessed Aug. 20, 2023).
- [66] “Firebase - Introduction,” *GeeksforGeeks*, Nov. 27, 2020. <https://www.geeksforgeeks.org/firebase-introduction/> (accessed Aug. 20, 2023).
- [67] Open AI, “OpenAI Platform,” 2023. <https://platform.openai.com> (accessed Aug. 20, 2023).
- [68] M. Nadia, “Python Tutorial: Streamlit,” Dec. 2021. <https://www.datacamp.com/tutorial/streamlit> (accessed Sep. 06, 2023).
- [69] “Streamlit • A faster way to build and share data apps,” Jan. 14, 2021. <https://streamlit.io/> (accessed Aug. 21, 2023).
- [70] dwaipayan, “Adding Lottie animation in Streamlit WebApp,” *GeeksforGeeks*, Feb. 04, 2023. <https://www.geeksforgeeks.org/adding-lottie-animation-in-streamlit-webapp/> (accessed Aug. 21, 2023).
- [71] krishna Ashutosh, “How to Use the JSON Module in Python – A Beginner’s Guide,” *freeCodeCamp.org*, Jun. 05, 2023. <https://www.freecodecamp.org/news/how-to-use-the-json-module-in-python/> (accessed Aug. 21, 2023).
- [72] M. Mehul, “An Introduction to Python’s Pandas Library for Data Manipulation,” *codedamn news*, Mar. 22, 2023. <https://codedamn.com/news/python/python-pandas-library> (accessed Aug. 21, 2023).
- [73] NumPy Developers, “What is NumPy? — NumPy v1.25 Manual.” <https://numpy.org/doc/stable/user/whatisnumpy.html> (accessed Aug. 21, 2023).
- [74] M. Kattamuri, “Python | Introduction to Matplotlib,” *GeeksforGeeks*, May 14, 2018. <https://www.geeksforgeeks.org/python-introduction-matplotlib/> (accessed Aug. 21, 2023).
- [75] Plotly, “Plotly.” <https://plotly.com/python/> (accessed Aug. 21, 2023).
- [76] M. Ali, “NLTK Sentiment Analysis Tutorial: Text Mining & Analysis in Python,” 2023. <https://www.datacamp.com/tutorial/text-analytics-beginners-nltk> (accessed Aug. 21, 2023).
- [77] V. Duong, “Python Word Clouds Tutorial: How to Create a Word Cloud,” 2023. <https://www.datacamp.com/tutorial/wordcloud-python> (accessed Aug. 21, 2023).
- [78] “Introduction of Datetime Modules in Python - Javatpoint,” *www.javatpoint.com*. <https://www.javatpoint.com/datetime-modules-in-python> (accessed Aug. 21, 2023).
- [79] RAJEEV, “What is Base64 Encoding and How does it work?” <https://www.base64encoder.io/learn/> (accessed Aug. 21, 2023).
- [80] “The G2 on Vercel,” *G2*, Jun. 23, 2023. <https://www.g2.com/products/vercel/reviews> (accessed Sep. 05, 2023).
- [81] W3C, “Web Speech API.” <https://wicg.github.io/speech-api/#introduction> (accessed Sep. 11, 2023).

## APPENDIX

### Review on English Practice Chatbot

Good day, I am Su Yi Wen, a final year student from University Tunku Abdul Rahman. I am working on my final year project with the title of "**Personal AI-Powered Conversation and Grammar-Checking Chatbot**"

The main aim of my project is twofold:

1. To establish an interactive English learning platform that enables users to enhance their language skills by engaging in conversations with a chatbot equipped with real-time grammar checking and paraphrasing features. This approach aims to supplement traditional lessons.
2. To introduce an "English Practice Chatbot Report" that users can use to visualize their activity and track their learning progress.

I am reaching out to kindly request your assistance in testing my project. Your valuable feedback will greatly contribute to assessing the quality of my work.

You can access the English Practice Chatbot Web via the following link: <https://fyp2syw-delta.vercel.app>

Thank you in advance for your time and assistance. Your input is highly appreciated.

#### Gentle Reminder

\*Please use a **valid email address**, but generate a **fake password** during registration. **Do not** use your actual email password.

#### Task

Try out the chat function with the bot to experience live grammar checking and sentence rephrasing in real-time.

You can select the "Random Topic" option to engage the bot in conversation on a random subject. Alternatively, you can utilize the microphone button to input text by speaking.

To access additional features, look to the top-right corner and use the menu for the following actions:

- (a) Review your chat history.
- (b) Examine rephrased sentences by inputting relevant keywords.
- (c) Export your input:

1. In this section, you can retrieve your chat messages with the bot by specifying the desired date, month, and year.
2. Click on the "Retrieve Chat Messages" option to download a CSV file that contains your messages.
3. Head over to the "English Practice Chatbot Report" section to access the report generator.
4. Upload the previously downloaded CSV file and wait for the results.

## APPENDIX

Name \*

Short-answer text

Age \*

- Under 12
- 13-17
- 18-24
- 25-34
- 35-44
- 45-54
- 55 and above

Academic Level \*

- Primary School
- Secondary School
- Diploma
- Degree
- Other (please specify)

What is your English language proficiency level? \*

- Beginner
- Intermediate
- Advanced

## APPENDIX

On a scale of 1 to 5, how user-friendly did you find the chatbot interface? \*

	1	2	3	4	5	
(Not user-friendly at all)	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	(Very user-friendly)

Did you find the real-time grammar checking helpful for improving your language skills? \*

- Yes, very helpful
- Yes, somewhat helpful
- Neutral
- No, not very helpful
- No, not helpful at all

Did you find the paraphrasing feature helpful for improving your language skills? \*

- Yes, very helpful
- Yes, somewhat helpful
- Neutral
- No, not very helpful
- No, not helpful at all

Did you find the "English Practice Chatbot Report" useful in tracking your learning progress? \*

- Yes, very useful
- Yes, somewhat useful
- Neutral
- No, not very useful
- No, not useful at all

## APPENDIX

Which components of this project impressed you \*

- Real-Time Grammar Checker
- Paraphraser
- English Practice Chatbot Report
- Other...

---

Do you have any additional comments, feedback, or suggestions that you would like to share?

Long-answer text

---

## FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year: T3,Y3</b>	<b>Study week no.: 2</b>
<b>Student Name &amp; ID: Su Yi Wen 20ACB06456</b>	
<b>Supervisor: Dr Aun Yichiet</b>	
<b>Project Title: Personal AI-Powered Conversation and Grammar-Checking Chatbot</b>	

### 1. WORK DONE

### 2. WORK TO BE DONE

- Refine and solidify the statement of the problem and objectives.
- Conduct a search for a suitable platform to construct the report.
- Conduct a study to identify appropriate metrics that need to be included in the report.

### 3. PROBLEMS ENCOUNTERED

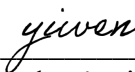
- Lacks a clear vision of the report webpage at this stage.

### 4. SELF EVALUATION OF THE PROGRESS

- Despite encountering some challenges, I was able to overcome them with the guidance of my supervisor and the resources available on the internet.



Supervisor's signature



Student's signature

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year: T3, Y3</b>	<b>Study week no.: 4</b>
<b>Student Name &amp; ID: Su Yi Wen 20ACB06456</b>	
<b>Supervisor: Dr Aun Yichiet</b>	
<b>Project Title: Personal AI-Powered Conversation and Grammar-Checking Chatbot</b>	

## 1. WORK DONE

- The decision has been made to utilize Visual Studio Code and Streamlit for the report development.
- The decision has also been made to include metrics such as message per day, sentiment analysis, unique words, grammar error analysis and tone analysis.

## 2. WORK TO BE DONE

- Set up a Streamlit account for the project.
- Looking for library from python that able to aids in calculating the metrics in the report

## 3. PROBLEMS ENCOUNTERED

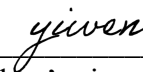
- This phase was relatively problem-free.

## 4. SELF EVALUATION OF THE PROGRESS

- The progresses has been smooth so far, largely attributed to the skills and knowledge acquired from the Youtube and online resources.



Supervisor's signature



Student's signature

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year: T3, Y3</b>	<b>Study week no.: 6</b>
<b>Student Name &amp; ID: Su Yi Wen 20ACB06456</b>	
<b>Supervisor: Dr Aun Yichiet</b>	
<b>Project Title: Personal AI-Powered Conversation and Grammar-Checking Chatbot</b>	

## 1. WORK DONE

- The project's Streamlit account has been successfully created and integrated into the project.
- The python library to used has been decided (nltk, LanguageTool, etc)

## 2. WORK TO BE DONE

- Start to code for all the metrics to be included in the report

## 3. PROBLEMS ENCOUNTERED

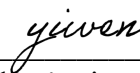
- Some problems faced during configuration of the library

## 4. SELF EVALUATION OF THE PROGRESS

- The implementation processes are going smoothly, with most resources and documentation found online.



Supervisor's signature



Student's signature



# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year: T3, Y3</b>	<b>Study week no.: 8</b>
<b>Student Name &amp; ID: Su Yi Wen 20ACB06456</b>	
<b>Supervisor: Dr Aun Yichiet</b>	
<b>Project Title: Personal AI-Powered Conversation and Grammar-Checking Chatbot</b>	

## 1. WORK DONE

- All the metrics planned to include has been done with simple interface / chart

## 2. WORK TO BE DONE

- Enhance the user interface of the report web
- Integrating project I web and project II web

## 3. PROBLEMS ENCOUNTERED

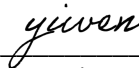
- No problem encounters.

## 4. SELF EVALUATION OF THE PROGRESS

- The primary metrics of the project have been successfully implemented and are operational.



Supervisor's signature



Student's signature

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year: T3, Y3</b>	<b>Study week no.: 10</b>
<b>Student Name &amp; ID: Su Yi Wen 20ACB06456</b>	
<b>Supervisor: Dr Aun Yichiet</b>	
<b>Project Title: Personal AI-Powered Conversation and Grammar-Checking Chatbot</b>	

## 1. WORK DONE

- Done integrating the project I and project II
- Done enhancing the user interface of project II (report web page)

## 2. WORK TO BE DONE

- Deploying the Project I and Project II
- Start user testing.
- Completing the report

## 3. PROBLEMS ENCOUNTERED


- Faced problem during integrating the Project I and Project II

## 4. SELF EVALUATION OF THE PROGRESS

- Overall project able to work well.
- Need to Think from the user's perspective when testing the project.



Supervisor's signature



Student's signature

# FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

<b>Trimester, Year: T3, Y3</b>	<b>Study week no.: 12</b>
<b>Student Name &amp; ID: Su Yi Wen 20ACB06456</b>	
<b>Supervisor: Dr Aun Yichiet</b>	
<b>Project Title: Personal AI-Powered Conversation and Grammar-Checking Chatbot</b>	

## 1. WORK DONE

- Completed the report writing and finalization of the project has been done.

## 2. WORK TO BE DONE

- Prepare slides for presentation.

## 3. PROBLEMS ENCOUNTERED

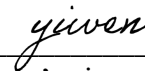
- No problem encounters.

## 4. SELF EVALUATION OF THE PROGRESS

- Project II has been successfully done on time.




Supervisor's signature




Student's signature

Poster



**Faculty of Information Communication & Technology**

## Personal AI-powered Conversation and Grammar Checking Chatbot




### Introduction

This project explores the effectiveness of a conversational AI chatbot for improving English language proficiency. Chatbots provide 24/7 accessibility, access to a broad range of language information, and engaging language practice partners, making them ideal for language learning.

### Conclusion

This project aims to create a web page for English Language Learners to practice their skills by conversing with a bot. It includes a real-time grammar checker and a paraphrase function to help users learn from their mistakes and explore different ways of expressing their ideas. Besides, a report also will be provided to visualize user's activities and progress.

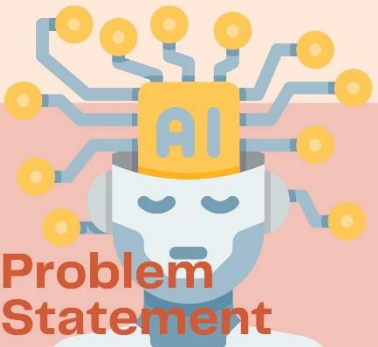


### Objective

1. To Build an English learning platform with a chatbot that offers real-time grammar checking and paraphrasing for conversation-based practice
2. To build an English Practice Chatbot Report for user to visualize their activities and the learning progress.

### Problem Statement

1. The lack of an English learning environment
2. Students engage in foreign language practice with peers at a similar proficiency level
3. Lacking interaction, grammar-checker, and paraphraser in existing language learning app
4. Lacking report in existing language learning app for user to keep track their progress



**Project Developer :Su Yi Wen**

**Project Supervisor :Dr Aun Yichiet**

## PLAGIARISM CHECK RESULT

### Personal AI-Powered Conversation and Grammar-Checking Chatbot

#### ORIGINALITY REPORT

<b>4%</b>	<b>3%</b>	<b>1%</b>	<b>2%</b>
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

#### PRIMARY SOURCES

<b>1</b>	<b>Submitted to Universiti Tunku Abdul Rahman</b> Student Paper	<b>&lt;1%</b>
<b>2</b>	<b>www.forbes.com</b> Internet Source	<b>&lt;1%</b>
<b>3</b>	<b>www.preprints.org</b> Internet Source	<b>&lt;1%</b>
<b>4</b>	<b>edoc.pub</b> Internet Source	<b>&lt;1%</b>
<b>5</b>	<b>Submitted to Asia Pacific University College of Technology and Innovation (UCTI)</b> Student Paper	<b>&lt;1%</b>
<b>6</b>	<b>Submitted to Trinity Valley Community College</b> Student Paper	<b>&lt;1%</b>
<b>7</b>	<b>www.fluentu.com</b> Internet Source	<b>&lt;1%</b>
<b>8</b>	<b>eprints.utar.edu.my</b> Internet Source	<b>&lt;1%</b>

## PLAGIARISM CHECK RESULT

9	<a href="http://www.macrumors.com">www.macrumors.com</a> Internet Source	<1%
10	<a href="http://www.coursehero.com">www.coursehero.com</a> Internet Source	<1%
11	Submitted to Thomas More Hogeschool Student Paper	<1%
12	<a href="http://www.edureka.co">www.edureka.co</a> Internet Source	<1%
13	Eric Sarrion. "Chapter 5 Using ChatGPT in Development Projects", Springer Science and Business Media LLC, 2023 Publication	<1%
14	Submitted to Manchester Metropolitan University Student Paper	<1%
15	Submitted to Cardiff University Student Paper	<1%
16	Submitted to NIIT University Student Paper	<1%
17	Submitted to University of Sydney Student Paper	<1%
18	<a href="http://community.openai.com">community.openai.com</a> Internet Source	<1%
19	<a href="http://www.cnblogs.com">www.cnblogs.com</a> Internet Source	<1%
20	<a href="http://zsg.240037mss.site">zsg.240037mss.site</a> Internet Source	<1%
21	Kleopatra Mageira, Dimitra Pittou, Andreas Papasalouros, Konstantinos Kotis, Paraskevi Zangogianni, Athanasios Daradoumis. "Educational AI Chatbots for Content and Language Integrated Learning", Applied Sciences, 2022 Publication	<1%
22	Shruti Saravanan, K. Sudha. "GPT-3 Powered System for Content Generation and Transformation", 2022 Fifth International Conference on Computational Intelligence and Communication Technologies (CCICT), 2022 Publication	<1%
23	A.M. Jasmine Hashana, P. Brundha, Mohamed Uvaze Ahamed Ayoobkhan, Fazila S. "Deep Learning in ChatGPT - A Survey", 2023 7th International Conference on Trends in Electronics and Informatics (ICOEI), 2023 Publication	<1%
24	Fabio Calefato, Filippo Lanubile, Damiano Romita, Rafael Prikladnicki, Joao Henrique Stocker Pinto. "Mobile Speech Translation for Multilingual Requirements Meetings: A Preliminary Study", 2014 IEEE 9th	<1%

## PLAGIARISM CHECK RESULT

### International Conference on Global Software Engineering, 2014 Publication

25	Submitted to JNTUA College of Engineering, Anantapur Student Paper	<1 %
26	Submitted to Bedford College Student Paper	<1 %
27	Submitted to UTC Heathrow Student Paper	<1 %
28	builtin.com Internet Source	<1 %
29	comparebrokers.co Internet Source	<1 %
30	Submitted to Arab Open University Student Paper	<1 %
31	ww2.mathworks.cn Internet Source	<1 %
32	www.researchgate.net Internet Source	<1 %
33	arxiv.org Internet Source	<1 %
34	development.earthsoft.org Internet Source	<1 %
	rapidapi.com	
35	Internet Source	<1 %
36	researchspace.ukzn.ac.za Internet Source	<1 %
37	"Genome-scale reconstruction of the metabolic network in Oenococcus oeni and its functional analysis in malolactic fermentation", Pontificia Universidad Catolica de Chile, 2021 Publication	<1 %
38	"Understanding the Predictive Analytics Life Cycle", Wiley, 2014 Publication	<1 %
39	Moudhich Ihab, Loukili Soumaya, Bahra Mohamed, Hmami Haytam, Fennan Abdelhadi. "Ontology-based sentiment analysis and community detection on social media", Proceedings of the 4th International Conference on Smart City Applications, 2019 Publication	<1 %
40	Submitted to University of Leeds Student Paper	<1 %
41	Submitted to University of Sunderland Student Paper	<1 %
42	dev.to Internet Source	<1 %

# PLAGIARISM CHECK RESULT

43 kinsta.com  
Internet Source

<1%

---

Exclude quotes On      Exclude matches < 8 words  
Exclude bibliography On



PLAGIARISM CHECK RESULT

<b>Form Title: Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)</b>			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1 of 1



**FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY**

<b>Full Name(s) of Candidate(s)</b>	Su Yi Wen
<b>ID Number(s)</b>	20ACB06456
<b>Programme / Course</b>	Bachelor of Information Systems (Honours) Business Information Systems / UCCB3596 Project 2
<b>Title of Final Year Project</b>	Personal AI-Powered Conversation and Grammar-Checking Chatbot

<b>Similarity</b>	<b>Supervisor's Comments (Compulsory if parameters of originality exceed the limits approved by UTAR)</b>
<b>Overall similarity index: <u>  4  </u> %</b>  <b>Similarity by source</b>  Internet Sources: <u>  3  </u> % Publications: <u>  1  </u> % Student Papers: <u>  2  </u> %	
<b>Number of individual sources listed of more than 3% similarity: <u>  0  </u></b>	
<b>Parameters of originality required, and limits approved by UTAR are as Follows:</b> (i) Overall similarity index is 20% and below, and (ii) Matching of individual sources listed must be less than 3% each, and (iii) Matching texts in continuous block must not exceed 8 words <i>Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.</i>	

Note: Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

*Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.*

\_\_\_\_\_  
Signature of Supervisor

Name:   Aun Yichiet  

Date:   14 Sep 2023  

\_\_\_\_\_  
Signature of Co-Supervisor

Name: \_\_\_\_\_

Date: \_\_\_\_\_

FYP 2 CHECKLIST

**FYP 2 CHECKLIST**



**UNIVERSITI TUNKU ABDUL RAHMAN**

**FACULTY OF INFORMATION & COMMUNICATION  
TECHNOLOGY (KAMPAR CAMPUS)**

**CHECKLIST FOR FYP2 THESIS SUBMISSION**

Student Id	20ACB06456
Student Name	Su Yi Wen
Supervisor Name	Dr Aun Yichiet

TICK (√)	DOCUMENT ITEMS
	Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
	Front Plastic Cover (for hardcopy)
√	Title Page
√	Signed Report Status Declaration Form
√	Signed FYP Thesis Submission Form
√	Signed form of the Declaration of Originality
√	Acknowledgement
√	Abstract
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
NA	List of Symbols (if applicable)
NA	List of Abbreviations (if applicable)
√	Chapters / Content
√	Bibliography (or References)
√	All references in bibliography are cited in the thesis, especially in the chapter of literature review
√	Appendices (if applicable)
√	Weekly Log
√	Poster
√	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
√	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

\*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.

*yiwen*

\_\_\_\_\_  
(Signature of Student)

Date: 14<sup>th</sup> September 2023