

*Project:
GPT Language Translation Application*



*Powered by GPT 3.5 Turbo
Software Development Project
UFCFFF-30-3*

Link to my github repo here: <https://github.com/RyanL2004/ai-react-app>

Table of contents

- Chapter One.....3
- Introduction.....3
- Problem Statement.....3
- Scope and Limitations.....4
- Clear Application Area.....5
- Unique Value Proposition.....6

- Chapter Two.....10
- Literature Review.....10
- Introduction of literature Review.....10
- Critical Analysis of Existing Literature.....11
- Relevant Research Findings.....11
- Connection to the Project.....11
- Gaps In Existing Literature.....12
- Connection To the Project Plan.....12
- Conclusion of The Literature Review.....12

- Chapter Three.....12
- Documentation Of Implementation.....12

- Chapter Four.....35
- Conclusion and Reflection.....35
- Summary of Key Points.....35
- Reflection On the Report.....35
- Recommendations for the App Expansion.....36
- Closing Remark.....36
- References and appendices.....37
- Link to the project’s github repository.....37

Chapter One

1-Introduction:

In response to the escalating demand for precise and adaptable language translation solutions, this pioneering project aims to spearhead an innovative application powered by the state-of-the-art GPT-3.5 Turbo, a cutting-edge functionality offered by OpenAI. At the heart of this endeavour is a dedicated focus on elevating prompts to achieve unparalleled translation results, coupled with a distinctive capability to employ advanced calling and formatting functions for the training of the GPT AI. The application seamlessly integrates OpenAI's API, tapping into the expansive capabilities of advanced language models. Key components of this initiative encompass the implementation of user authentication and Identity and Access Management (IAM) through the robust framework of Okta. Additionally, the project leverages the robust capabilities of Azure Microsoft Cognitive Services for speech-related functionalities such as STT and TTS, ensuring a comprehensive and multifaceted language processing approach. Furthermore, the strategic integration of Firebase adds efficiency to the authentication processes, contributing to a streamlined and secure user experience. By addressing practical scenarios, this project places significant emphasis on safeguarding user privacy, navigating ethical considerations inherent in language processing technologies, and encouraging a culture of continuous improvement driven by the invaluable feedback of end-users. As I embark on this technological journey, the endeavour is not merely about building a translation application but also about creating a sophisticated, user-centric ecosystem. The commitment to ethical considerations, privacy safeguards, and the relentless pursuit of excellence through user feedback distinguishes this project as a hallmark in the realm of language translation applications. In the subsequent sections, I delve into the intricate layers of the methodology adopted, exploring the potential of GPT-3.5 Turbo, elucidating the nuances of user authentication frameworks, and elucidating the significance of language models in the ever-evolving landscape of Natural Language Processing (NLP). Through this comprehensive exploration, the aim is to contribute substantively to the field, setting new benchmarks for precision, adaptability, and user-centric innovation.

1.1-Problem Statement:

Breaking through language barriers is a common challenge that many industries grapple with, making it tricky to communicate and collaborate seamlessly. In this ever-expanding

globalized world, areas like E-commerce, Travel, Tourism, Healthcare, and Education are seeing more situations where people and organizations encounter issues due to linguistic differences. Trying to understand and convey information across languages isn't always smooth sailing, presenting some significant hurdles:

E-commerce:

In the online business world, companies face a common obstacle: language barriers. Trying to describe products, answer customer questions, and provide support becomes a bit of a struggle. These language challenges make it tough for businesses to connect with people globally.

Travel:

Think about traveling to a new place. It's exciting, but here's the catch: language can be a barrier. Tourists often find it hard to understand important information, whether it's spoken or written. This language puzzle adds a layer of difficulty to exploring foreign lands.

Healthcare:

Taking a step into healthcare, imagine the challenge when doctors and patients can't communicate easily due to language differences. It's not just about words; it can affect how medical information gets across, leading to possible misunderstandings and impacting the patient's care.

Education:

In the world of education, language learning and cultural programs encounter a setback. Students eager to understand different perspectives struggle when translations aren't quite right. It makes it harder for them to fully grasp the richness of other languages.

This challenge goes beyond just converting words from one language to another. It's about needing solutions that are like a reliable friend: precise, flexible, and all about you. We're talking about tools that can navigate the complexities of different fields and situations, like understanding the vibe of a room. The hitch is the current solutions don't quite cut it. They struggle to give us those finely tuned translations, leaving room for mix-ups, inefficiencies, and an obstacle for cross-cultural exchanges.

1.2-Scope and Limitations: (Cost effectiveness- boundaries, Languages, money)

Imagine this Language Translation App as a reliable language buddy, starting off by cracking the code between English and French. It's like having a helpful friend for situations in E-commerce, Travel, Healthcare, and Education making communication smoother in these areas.

But let's keep it real and be aware of the challenges because languages can be as tricky as navigating a maze. It's like dealing with the subtle differences in how people speak in different regions and being ready for any language someone might throw its way.

Now, here's the secret to make this app a language maestro it needs a guiding hand from someone who speaks French like a local, someone who gets the nuances, expressions, and

even the intricacies of second-degree sentences. Think of it as a coach, teaching the app the art of language, and only a French native speaker “Me” with a keen understanding of these details can make that happen.

While the application aims to provide a versatile solution, it recognizes that achieving comprehensive language coverage and addressing every linguistic nuance is an ongoing challenge and cost effective. Moreover, potential biases in AI models and ethical considerations in language processing technologies pose additional challenges that require careful navigation.

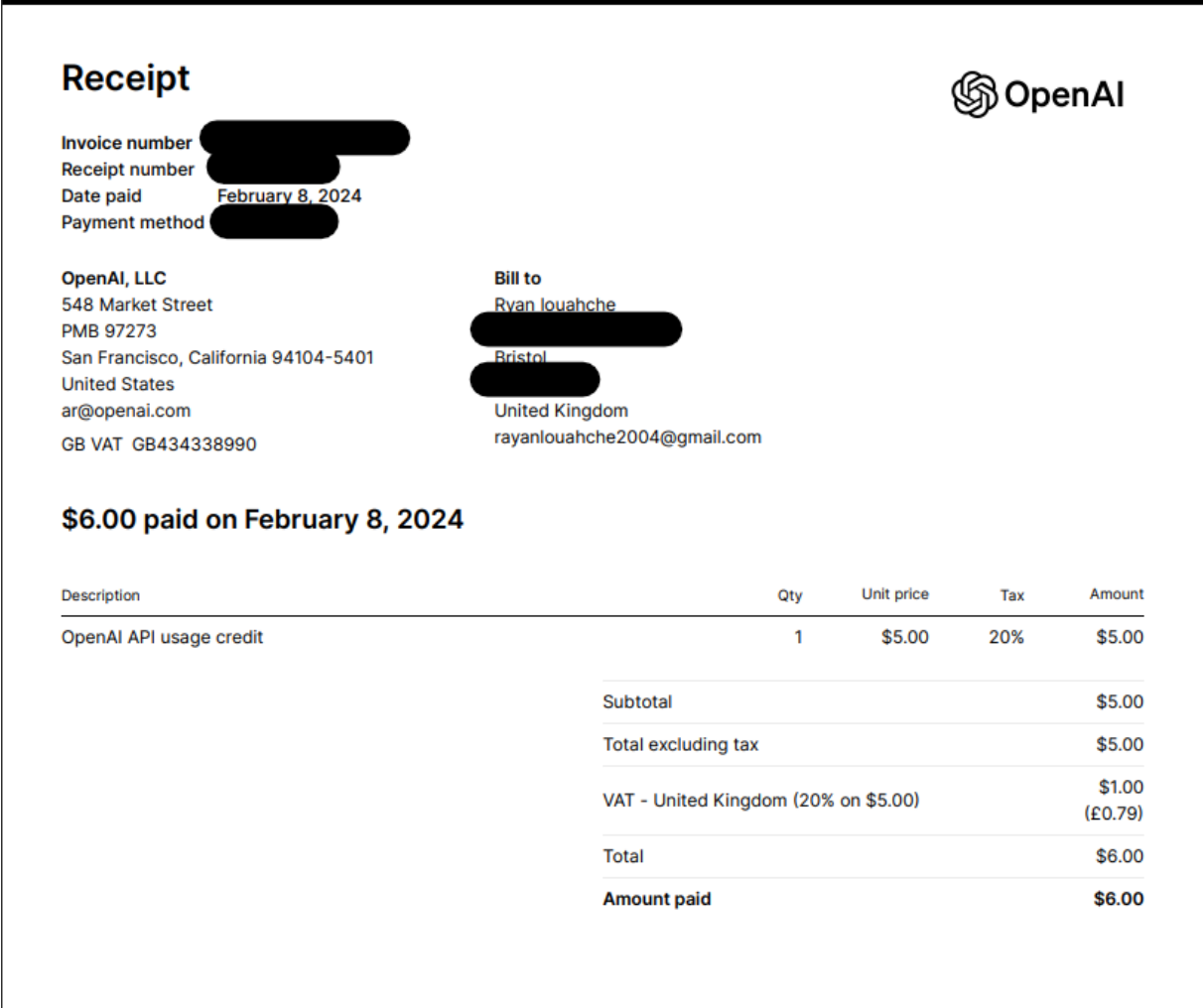


Figure. Cost Effectiveness spent on the AI model for dev and testing.

1.3-Clear Application Area:

While developing the language translation application powered by GPT3.5 Turbo, understanding the specific contexts and industries where language barriers pose challenges is essential to make a significant world impact. Tailoring an approach to meet the diverse

needs of end users, the application targets specific key sectors, where each present unique communication challenges that only the GPT3.5 LTA aims to address effectively.

Below are the valuable scenarios this Application aims to address:

E-commerce:

Language barriers can hinder the global reach of businesses [Citation], impeding communication between merchants and customers. Product listings, customer inquiries and support interactions often require accurate, precise, and timely translations to facilitate seamless transactions, the application seeks to bridge these language gaps by enabling automatic translation of product descriptions, customer queries, and support interactions to enhance cross-border commerce.

Travel and Tourism:

For Tourists, Travelers exploring foreign countries, Language differences can be a hurdle in navigating unfamiliar environments, reading directions panels, and communicating with locals. Whether seeking for directions, ordering food, or even socialising with people, encountering these issues hinder their experiences.

The application tends to provide real time translations of spoken, written, or JPG text translation content, empowering travellers to engage with local cultures.

Healthcare:

Effective communication between healthcare providers and patients is more than essential and paramount for ensuring high quality patient management, care, and safety. Language barriers can importune the exchange of critical medical information. Since my mom is a gynaecologist, she could share with me the application areas where the translation application could bring a plus to the body of medicine in the world.

She underlines the fact that many scientific words change from one language to other, words such as flu, Angina, bronchitis, rhinopharyngitis, not to mention the difficulty of having to understand a foreigner to explain the reason of discomfort.

The application endeavours to facilitate communication in healthcare settings, giving precise and context aware translations for scientific words, facilitating healthcare professionals to interact with patients more efficiently and comprehensively making their work less strenuous and saving them time.

Education:

Language learning and cultural exchange programs rely heavily on effective communication to facilitate meaningful interactions and learning experiences. However, language barriers can hinder the exchange of ideas and hinder students' ability to fully engage with course materials and cultural nuances. The application aims to support language learning and cultural exchange initiatives by providing accurate translations tailored to educational contexts, fostering cross-cultural understanding and enriching learning experiences.

User centric approach:

the GPT3.5 Turbo LTA is designed with a user centric approach, placing the needs and client's preferences of users at the forefront of the development process. Conducting Extensive User Surveys and Research, made me gain valuable insights into the specific area of application and overall end User requirements. Understanding user preferences, language proficiency levels and communication preferences. I aim to Tailor the application meets Uniqueness in the context of AI translation ensuring personalized and intuitive experience among new users.

Scalability Plan:

The initial focus in supporting translations between English and French, underlines the importance of scalability to accommodate additional Users with additional Languages and industries, a key consideration for future growth and app expansion. Reason why the initial Application architecture is flexible and scalable allowing seamless and additional integrations and features as user demand evolves. Ensuring the application scales efficiently is fundamental for well expansion of this AI translation APP.

1.4-Unique Value Proposition:

The development of such an AI language translation application that transcends conventional solutions introduces the concept of prompt engineering as a cornerstone of the unique value proposition. Prompt engineering is a novel approach to fine tuning the output of language models such as GPT 3.5 Turbo in this application's case. By crafting tailored prompts that guide the model's generation process towards more contextually relevant accurate translation, the user will have to precise the situation he is encountering such as:

-Spoken Phrase Translation with Speech-to-Text (English to French):

Prompt: "Translate the following spoken English phrase to French: '[User input: Spoken English phrase]'"

-Written Phrase Translation (French to English) with Text-to-Speech:

Prompt: "Translate the following written French phrase to English: '[User input: Written French phrase]'"

-Restaurant Menu Item Translation (French to English) with Text-to-Speech:

Prompt: "Translate the following French menu item to English: '[User input: Menu item]'"

-Train Station Announcement Translation with Speech-to-Text (English to French):

Prompt: "Translate the following train station announcement from English to French: '[User input: Train station announcement]'"

-Hotel Check-In Phrase Translation with Speech-to-Text (English to French):
Prompt: "Translate the following English check-in phrase to French: '[User input: Check-in phrase]'"

-Emergency Assistance Translation (French to English) with Text-to-Speech:
Prompt: "Translate the following emergency assistance phrase from French to English: '[User input: Emergency assistance phrase]'"

-Marketplace Bargaining Translation with Speech-to-Text (English to French):
Prompt: "Translate the following bargaining phrase from English to French: '[User input: Bargaining phrase]'"

Understanding Prompt Engineering:

Prompt Engineering involves careful design of formulation input the user may be using to generate specific and precise outputs. It provides the model with structured input that includes context, constraints, and guidelines steering the output towards desired outcomes preventing ethical biases. The whole prompt engineering process requires deep understanding of the intricacies of the language model and the linguistic nuances of the target language. By crafting Prompts that tailor to capture these nuances, the GPT3.5 Turbo LTA empowers to deliver translations that resonates authentically with users such as a real human translator.

Importance of native expertise in the Prompt Engineering process:

As part of the commitment to precision and authenticity, French has been selected as the inaugural language for the GPT-3.5 LTA. This decision was influenced by the expertise of my Supervisor, Bashir Sanusi, who emphasized the importance of native speakers in guiding language prompts. Leveraging native linguistic proficiency is essential in prompt engineering, as it allows for a deeper understanding of the intricacies of the language, including colloquialisms, idiomatic expressions, and cultural nuances. Since I am a native French speaker, this decision was made promptly and confidently, ensuring the integrity and accuracy of the language translation application.

Why French Matters: (reference)

French serves as an ideal starting point for the GPT3.5 LTA due to its rich linguistic heritage and global significance. As one of the most widely spoken languages in the world, French presents a diverse array of linguistic challenges and opportunities, making it an ideal testbed for the prompt engineering approach. By mastering the complexities of French translation, I lay a solid foundation for expanding the application to other languages, confident in the ability to navigate the intricacies of diverse linguistic landscapes.

Education: (Legal Documents Translation)

Consider the translation of legal documents, where precision and accuracy are paramount. By crafting prompts that include legal terminology, formatting conventions, and contextual cues, I can guide the language model to produce translations that adhere to legal standards and conventions. For example, prompts may specify the type of legal document (e.g., contract, affidavit) and include relevant legal terms and clauses to ensure the accuracy and validity of the translated text.

Tourism, Travel and Socialisation: (Casual Conversation Translation)

In contrast, translating casual conversations requires a different approach that captures the nuances of informal language and colloquial expressions. By designing prompts that mimic the conversational context, including slang terms, idiomatic expressions, and cultural references, I can tailor the language model's output to reflect the natural flow and tone of informal dialogue. Prompts may include sampling conversational snippets or scenarios to provide context for the translation process.

Business: (Technical Documentation Translation)

Technical documentation often contains specialized terminology and jargon that require precise translation. Prompt engineering allows us to guide the language model's output towards accurate technical translations by incorporating industry-specific terminology, abbreviations, and formatting conventions into the prompts. For example, prompts may include glossaries of technical terms, sample code snippets, or formatting instructions to ensure the fidelity and clarity of the translated documentation.

This sophisticated approach to achieving context-aware translations unlike traditional translation tools such as Google Translate that relies solely on input specifications, prompt engineering delves deeper into understanding the intricacies of language models and the specific nuances of target languages.

It recognizes that language is not only a sequence of words but a complex interplay of cultural context, idiomatic expressions, subtle nuances but also slang expressions. By crafting prompts that tailor to capture these nuances, the GPT3.5 Turbo LTA empowers to deliver translations that resonate authentically with users such as a real human translator.

By structuring prompts to encapsulate contextual cues and domain-specific knowledge the app finds itself equipped enough for usage and translation adaptation.

For example, in a scenario where the user seeks a translation for a medical document, the prompt engineering methodology would incorporate specialized medical terminology and contextual cues to ensure accurate and precise translations tailored to the healthcare domain. Similarly, in the realm of legal documents or technical manuals, prompts would be crafted to capture the specificity and precision required in these domains, thereby enhancing the relevance and accuracy of translations.

Examples of Prompt Engineering in Action:

Medical Translations:

*Prompt: "Translate the following medical term 'myocardial infarction' to French."
By structuring prompts to include domain-specific terms and context, the application can deliver accurate translations crucial for medical professionals and patients alike.*

Legal Documents:

*Prompt: "Translate the following legal clause 'force majeure' to Spanish."
Through prompt engineering, the application ensures translations maintain the legal precision and terminology essential for contractual agreements and legal proceedings.*

Casual Conversations:

*Prompt: "Translate the following colloquial phrase 'break a leg' to French."
Even in informal contexts, the application maintains cultural sensitivity and linguistic nuances, ensuring translations resonate naturally with users.*

Employing prompt engineering, elevates the quality and authenticity of translations, enabling the application to bridge linguistic barriers effectively and facilitate seamless communication across diverse contexts.

Chapter two:

2-Literature Review:

2.1-Introduction

In recent years, significant advancements have been made in natural language processing (NLP) and machine translation technologies, fuelled by breakthroughs in deep learning and neural network architectures (Brown et al., 2020). This literature review aims to critically analyse existing research findings and explore the connection between these advancements and the development of a language translation application leveraging GPT-3.5 Turbo or AI, with integrated vocal recognition for translation input.

2.2-Critical Analysis of Existing Literature:

The introduction of GPT-3 by Brown et al. (2020) marked a milestone in NLP, showcasing the remarkable capabilities of large-scale language models as few-shot learners. GPT-3's ability to perform various language tasks with minimal training data has garnered significant attention in the research community, with many studies focusing on its applications in machine translation.

Cho et al. (2019) provide a comprehensive overview of recent advances in neural machine translation, emphasizing the role of transformer models in improving translation accuracy and efficiency. Transformer architectures, as introduced by Vaswani et al. (2017), have revolutionized NLP tasks, serving as the foundation for models like GPT-3.

Wang et al. (2021) delve into the advancements in end-to-end speech translation using transformer models, highlighting the potential of integrating speech recognition and translation tasks. Graves et al. (2013) explore the use of deep recurrent neural networks for speech recognition, which aligns with the vocal recognition aspect of the translation application.

2.3-Relevant Research Findings:

The research findings indicate that transformer-based models like GPT-3 offer promising capabilities for language translation tasks, including fast and precise translations with minimal training data (Brown et al., 2020). These models leverage attention mechanisms to capture contextual information effectively, leading to improved translation quality. Additionally, advancements in speech recognition technology have paved the way for seamless integration of vocal input into translation applications. By combining transformer-based translation models with state-of-the-art speech recognition systems, the application aims to provide a user-friendly and efficient translation experience (Wang et al., 2021).

2.4-Connection to the Project:

The project seeks to build upon these research findings by developing a language translation application that harnesses the power of GPT-3.5 Turbo or AI for fast and precise translations. By integrating vocal recognition capabilities, users can input translations effortlessly using their voice, enhancing accessibility and usability (Brown et al., 2020).

Gaps in Existing Literature:

While existing research has demonstrated the effectiveness of transformer-based models and speech recognition technology in language translation applications, there remains a gap in understanding the specific challenges and limitations of integrating these technologies into a unified platform (Cho et al., 2019). Reviewing similar projects and their limitations can help identify areas for improvement and innovation in the translation application.

2.5-Connection to The Project Plan:

The insights gained from the literature review will inform the project plan, guiding the development of the language translation application. By addressing the gaps identified in existing literature and building upon recent research findings, I aim to create a robust and user-centric application that meets the evolving needs of language translation in today's digital age (Wang et al., 2021).

2.6-Conclusion of the Literature Review:

In conclusion, the literature review highlights the significant advancements in NLP, machine translation, and speech recognition technologies. By leveraging insights from existing research and addressing identified gaps, the project endeavours to contribute to the ongoing evolution of language translation applications, offering users a simple, fast, and precise translation experience (Brown et al., 2020).

Chapter Three:

3-Documentation of Implementation:

3.1-Research Resources:

In the development of the language translation application, leveraging a robust array of research resources is paramount to ensure informed decision-making, technical accuracy, and alignment with industry best practices. This section outlines the key research materials and tools that will guide the project implementation:

1- Development Environment: (Example and case Studies)

-Frontend Framework: **React**

React is chosen as the frontend framework for its numerous advantages in building interactive user interfaces. Here's a deeper dive into its utility:



Vite + React

```
1 Getting Started with Create React App
2
3 This project was bootstrapped with [Create React App](https://github.com/facebook/create-react-app).
4
5 ## Available Scripts
6
7 In the project directory, you can run:
8
9 ### `npm start`
10
11 Runs the app in the development mode.
12 Open [http://localhost:3000](http://localhost:3000) to view it in your browser.
13
14 The page will reload when you make changes.
15 You may also see any lint errors in the console.
16
17 ### `npm test`
18
19 Launches the test runner in the interactive watch mode.
20 See the section about [running tests](https://facebook.github.io/create-react-app/docs/running-tests) for more information.
21
22 ### `npm run build`
23
24 Builds the app for production to the `build` folder.
25 It correctly bundles React in production mode and optimizes the build for the best performance.
26
27 The build is minified and the filenames include the hashes.
28 Your app is ready to be deployed!
29
30 See the section about [deployment](https://facebook.github.io/create-react-app/docs/deployment) for more information.
31
32 ### `npm run eject`
33
34 **Note: this is a one-way operation. Once you `eject`, you can't go back!**
35
36 If you aren't satisfied with the build tool and configuration choices, you can `eject` at any time. This command will remove the single build
37 dependency from your project.
```

Figure1. React Application Creation

Component-Based Architecture: React's component-based architecture allows to break down the user interface into reusable components, promoting code reusability, maintainability, and scalability.

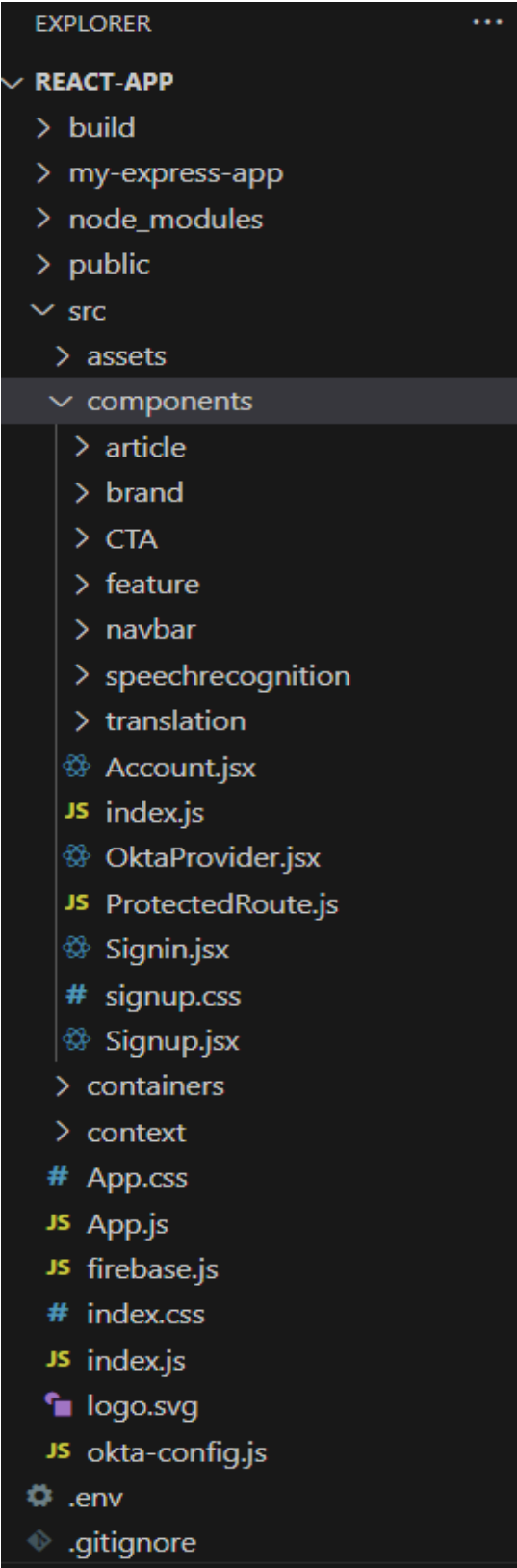


Figure 2. React Components Breakdown

Efficient Rendering: React's virtual DOM and reconciliation algorithm ensure efficient rendering of UI components, minimizing unnecessary DOM updates and improving performance.

Rich Ecosystem: React has a vast ecosystem of libraries, tools, and community support, providing access to a wide range of resources for building robust frontend applications.

Build Tool: Vite

Vite is selected as the build tool for its modern development experience and fast build times. Here's why it's beneficial:

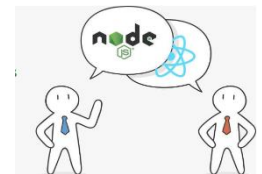
Fast Build Times: Vite leverages modern build tools like ESBuild to achieve blazingly fast build times, reducing development iteration cycles and improving developer productivity.

Modern Development Workflow: Vite's support for modern JavaScript features, hot module replacement (HMR), and instant server start enhances the development workflow, enabling rapid prototyping and iterative development.

Optimized Build Output: Vite generates optimized build output, including efficient JavaScript bundles and CSS files, resulting in smaller bundle sizes and improved application performance.

Backend Runtime: Node.js

Node.js powers the backend of the application, providing a JavaScript runtime environment for server-side logic implementation. Here's how it contributes to the project:









Nom	Modifié le	Type	Taille
 Install Additional Tools for Node.js	2/11/2024 9:46 PM	Raccourci	1 Ko
 Node.js command prompt	2/11/2024 9:46 PM	Raccourci	1 Ko
 Node.js documentation	2/11/2024 9:46 PM	Raccourci Internet	1 Ko
 Node.js website	2/11/2024 9:46 PM	Raccourci Internet	1 Ko
 Node.js	2/11/2024 9:46 PM	Raccourci	1 Ko
 Uninstall Node.js	2/11/2024 9:46 PM	Raccourci	1 Ko

Figure3. Node.js for backend development.

JavaScript Everywhere: With Node.js, we can use JavaScript for both frontend and backend development, facilitating code sharing, consistency, and developer productivity.

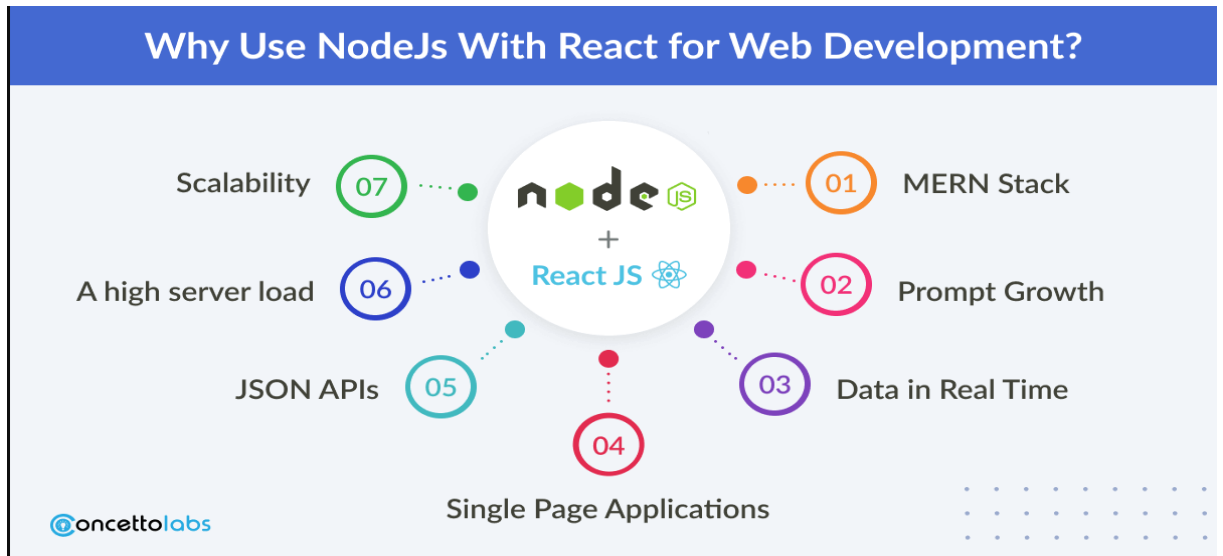


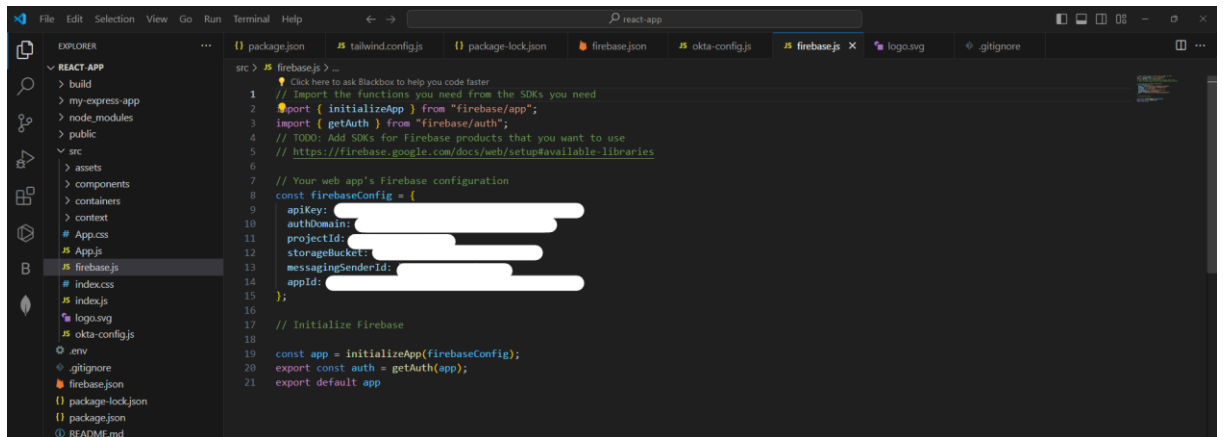
Figure4. Node.js+React JS (Figure from Concetto Labs's Forum)

Asynchronous Operations: Node.js's event-driven, non-blocking I/O model enables efficient handling of asynchronous operations, such as database queries and network requests, improving application responsiveness and scalability.

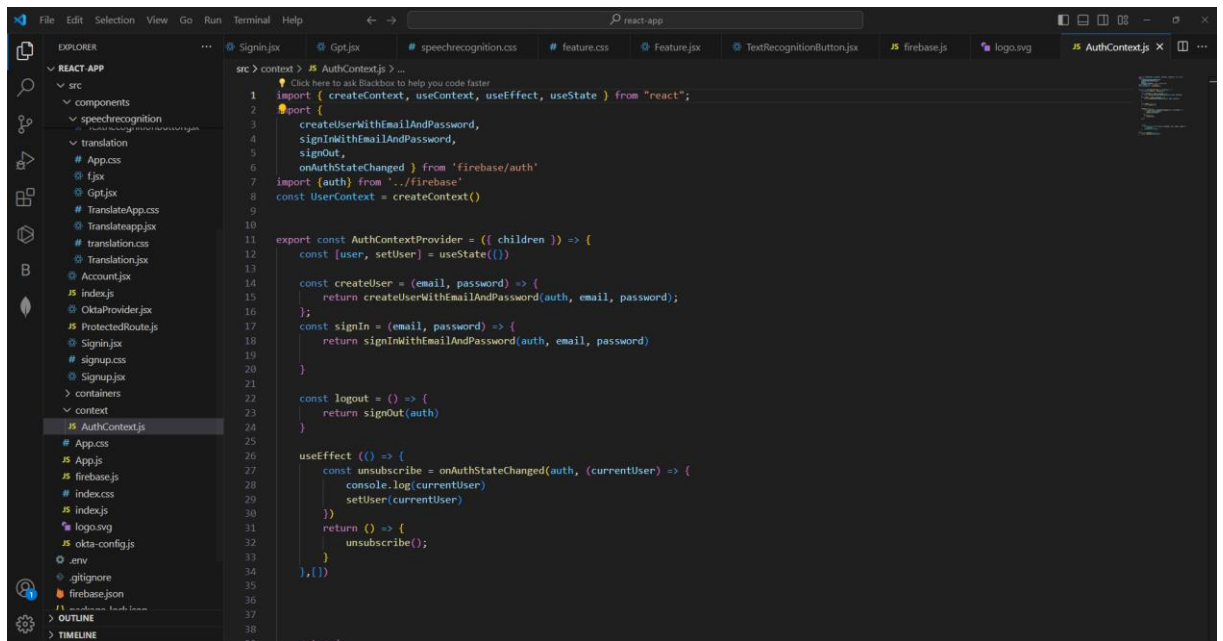
NPM Ecosystem: Node.js has a vast ecosystem of npm packages and modules, providing access to libraries and tools for building a wide range of backend functionalities, from web servers to data processing.

Authentication Service: Firebase Authentication

Firebase Authentication is integrated into the application for user authentication and authorization. Here's how it enhances the security and user management aspects of the project:



```
src > firebase.js > ...
1 // Click here to ask Blackbox to help you code faster
2 // Import the functions you need from the SDKs you need
3 import { initializeApp } from "firebase/app";
4 import { getAuth } from "firebase/auth";
5 // TODO: Add SDKs for Firebase products that you want to use
6 // https://firebase.google.com/docs/web/setup#available-libraries
7
8 // Your web app's Firebase configuration
9 const firebaseConfig = {
10   apiKey: [REDACTED],
11   authDomain: [REDACTED],
12   projectId: [REDACTED],
13   storageBucket: [REDACTED],
14   messagingSenderId: [REDACTED],
15   appId: [REDACTED]
16 };
17 // Initialize Firebase
18
19 const app = initializeApp(firebaseConfig);
20 export const auth = getAuth(app);
21 export default app
```



```
src > context > AuthContext.js > ...
1 import { createContext, useContext, useEffect, useState } from "react";
2 import {
3   createUserWithEmailAndPassword,
4   signInWithEmailAndPassword,
5   signOut,
6   onAuthStateChanged } from "firebase/auth"
7 import { auth } from "../firebase"
8 const UserContext = createContext()
9
10
11 export const AuthContextProvider = ({ children }) => {
12   const [user, setUser] = useState()
13
14   const createUser = (email, password) => {
15     return createUserWithEmailAndPassword(auth, email, password);
16   };
17   const signIn = (email, password) => {
18     return signInWithEmailAndPassword(auth, email, password);
19   };
20   const logout = () => {
21     return signOut(auth);
22   };
23
24   useEffect(() => {
25     const unsubscribe = onAuthStateChanged(auth, (currentUser) => {
26       console.log(currentUser);
27       setUser(currentUser);
28     });
29     return () => {
30       unsubscribe();
31     };
32   }, []);
33 }
34
35
36
37
38
```

Figure 5. FireBase Integration and Config

Secure User Management: Firebase Authentication provides robust authentication mechanisms, including email/password authentication, social login options (e.g., Google, Facebook), and multi-factor authentication (MFA), ensuring secure user management.

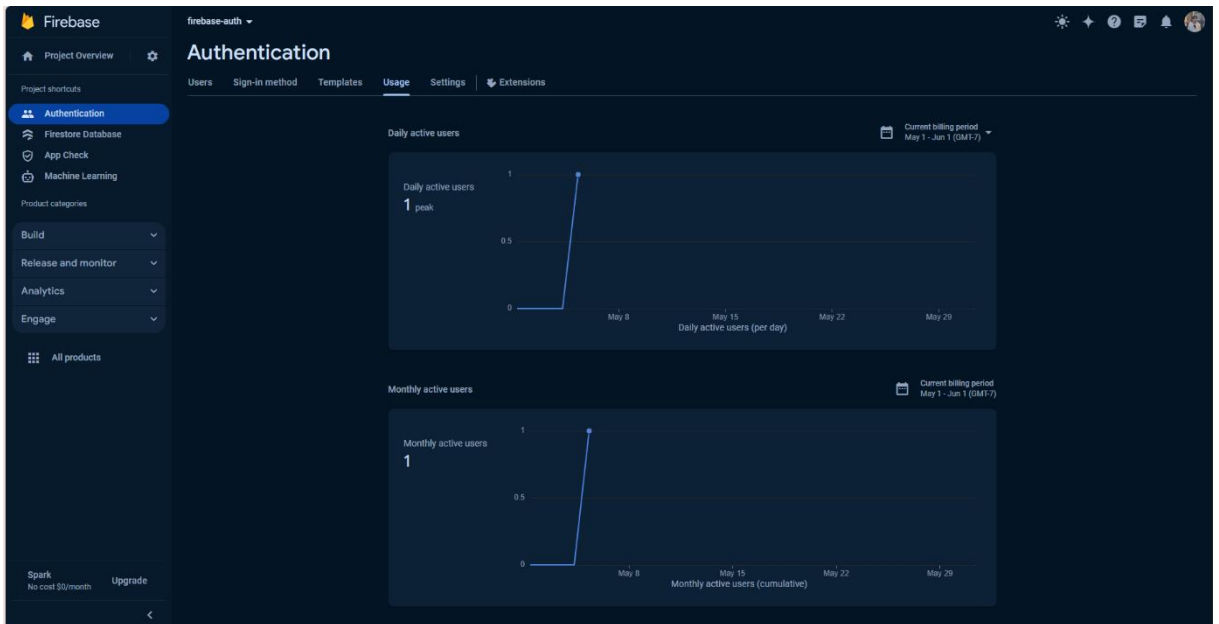


Figure5.1 FireBase User Login Management

Identifier	Providers	Created	Signed in	User UID
mielpops@gmail.com		5 May 2024	5 May 2024	wWT0xst5lswQJwb4kDcj3ovl...
test12345@gmail.com		13 Feb 2024	13 Feb 2024	Vo7yCCBwI8WHQI9EFSagZo...
testversion@gmail.com		29 Jan 2024	29 Jan 2024	nfawqoy0af7Z24YhT0g0OU...
kurva@gmail.com		12 Dec 2023	12 Dec 2023	YOkkgJFom8VBR0GHAslyY9r...
testestfmail@gmail.com		11 Dec 2023	11 Dec 2023	Ikx0dQ62JNoh21GhHlMwHhA...
rayanlouahche2004@g...		5 Dec 2023	9 Jan 2024	HjZFyjkCf9ZPKoy50yoswV4m...
finaltest2@test.com		2 Dec 2023	2 Dec 2023	5fPDkYQcPfkaxz00m8BRaEi...
finaltest@test.com		2 Dec 2023	2 Dec 2023	JEwk3iFD5f4embk6zCudfFg...
testme1@gmail.com		1 Dec 2023	2 Dec 2023	su0ENigG6q1WdWrlakSG2P...
testme@gmail.com		1 Dec 2023	1 Dec 2023	qW5jg8fACU9B2Y8ycrOIA9...
test@test.com		1 Dec 2023	1 Dec 2023	1bqD68RegaXfqrKdm952psEw...

Figure5.2 FireBase User Management

Easy Integration: Firebase Authentication offers seamless integration with other Firebase services, such as Firestore and Cloud Functions, simplifying the implementation of authentication-related features and workflows.

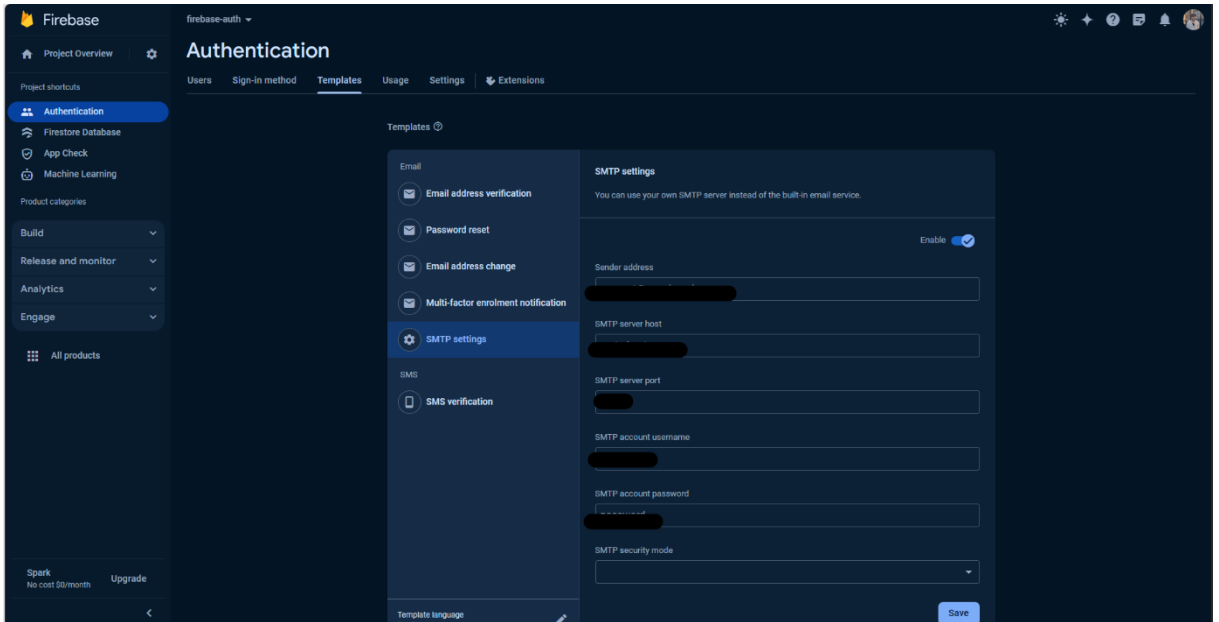


Figure 5.3 SMTP Server Creation

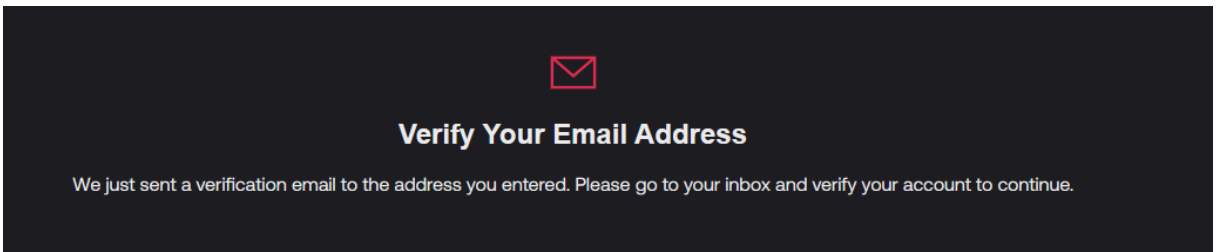
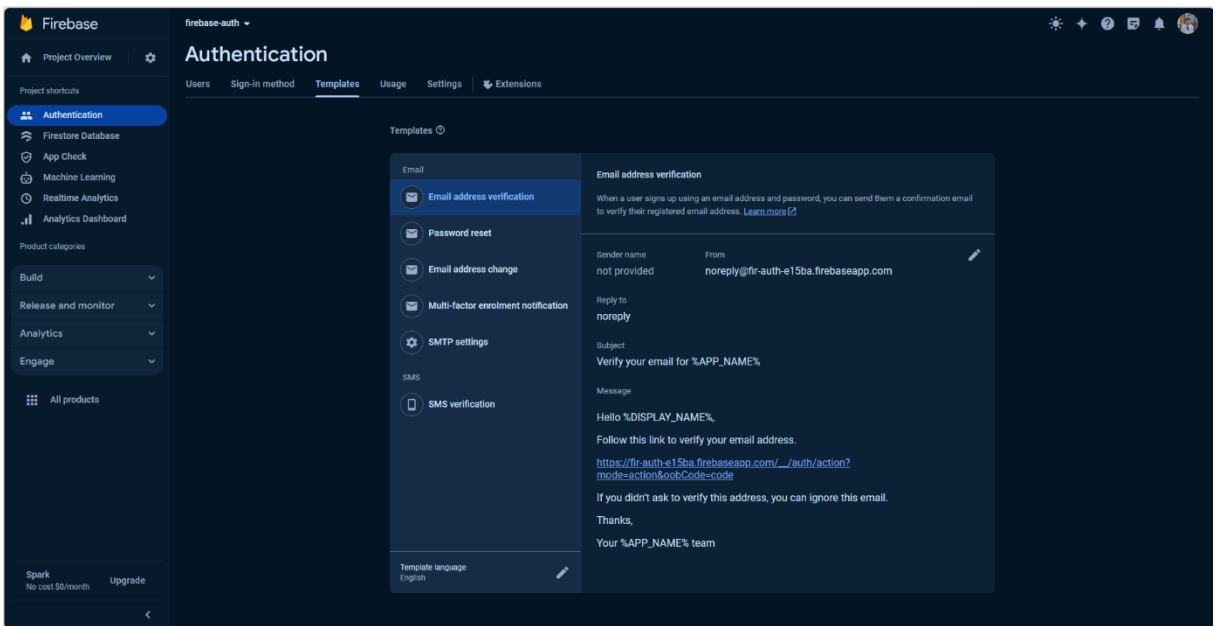


Figure 5.4 SMTP requests, Email Address Verification

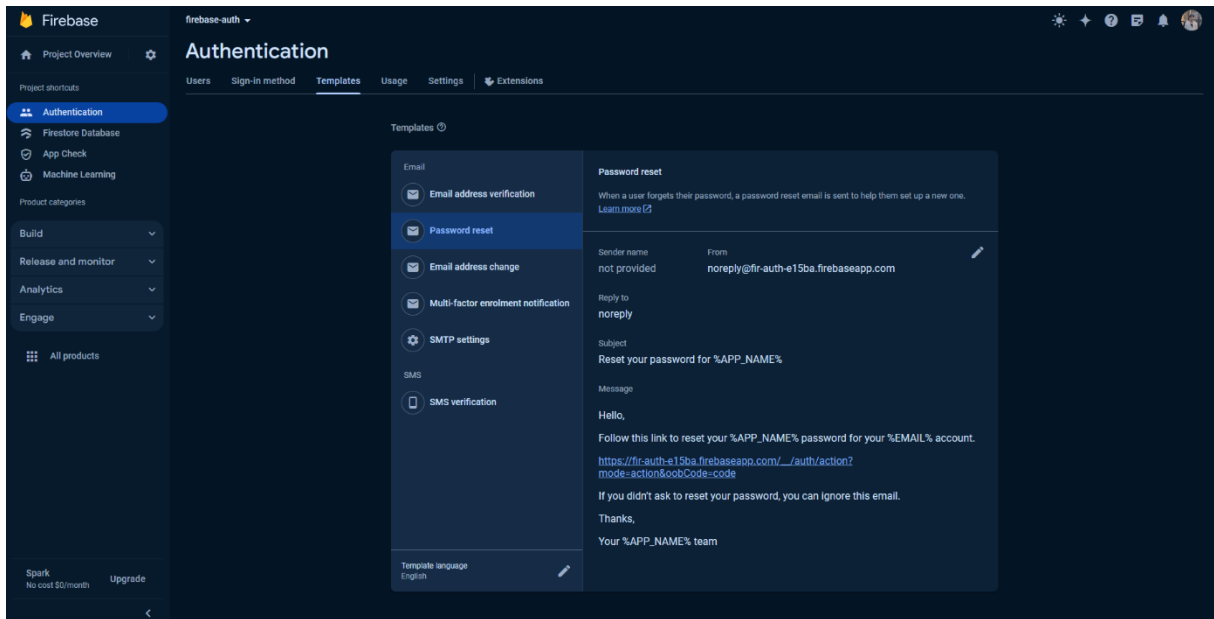


Figure 5.4 SMTP requests, Password Reset

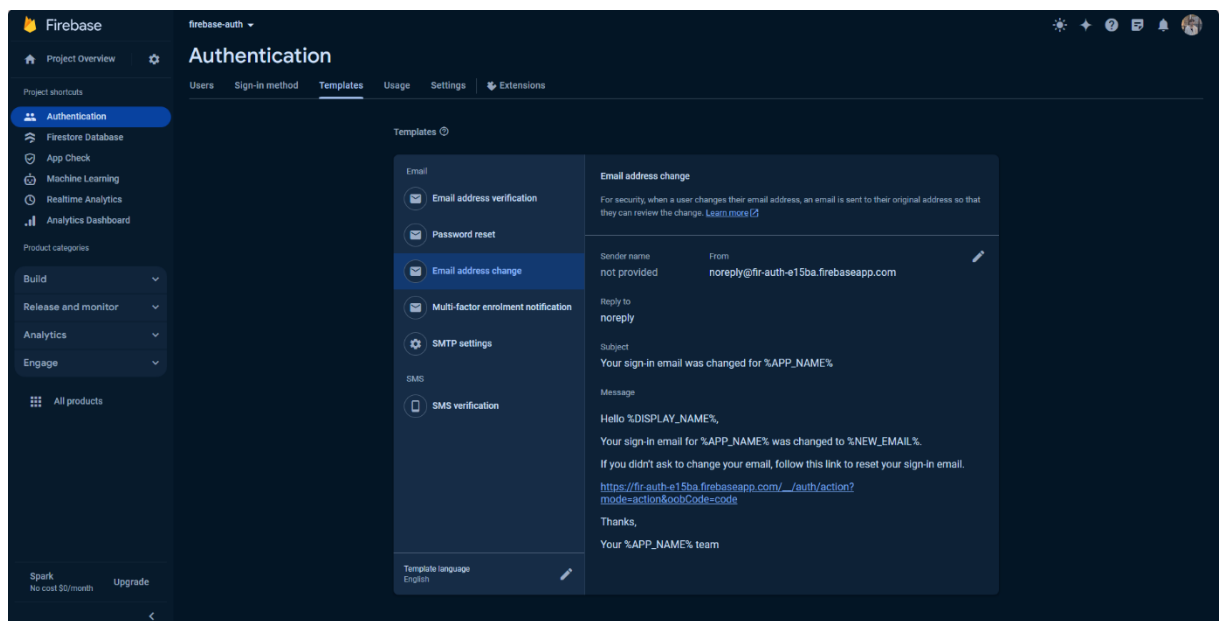


Figure 5.5 SMTP requests, Email Address Change

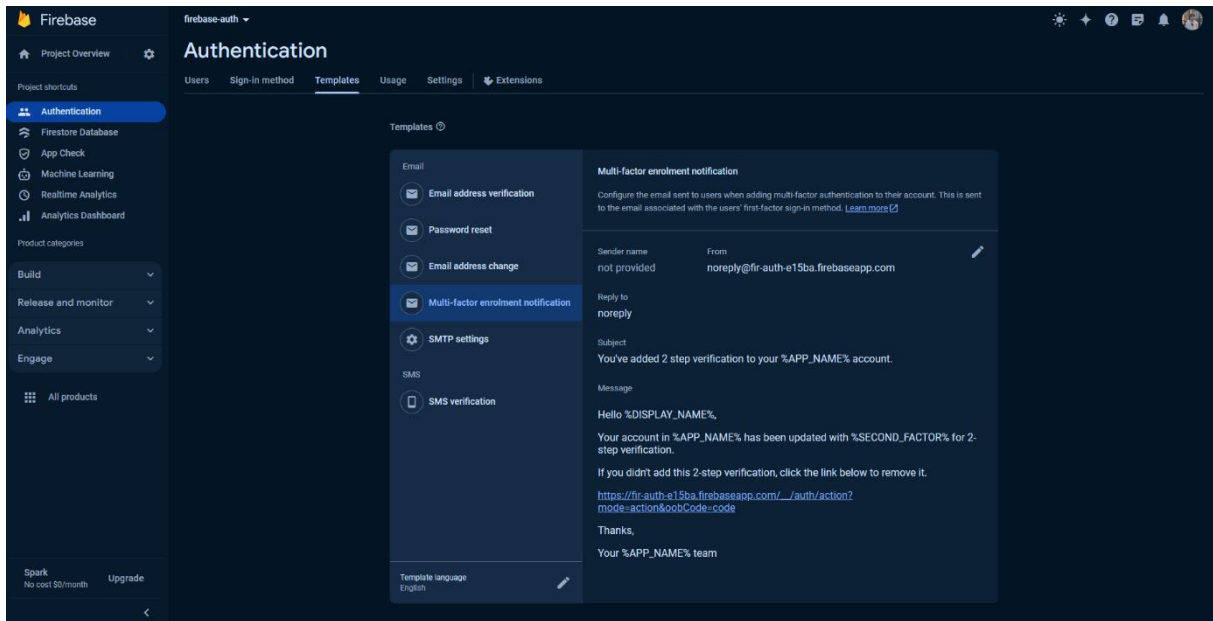


Figure 5.6 SMTP requests, Muti-factor Enrollment notification

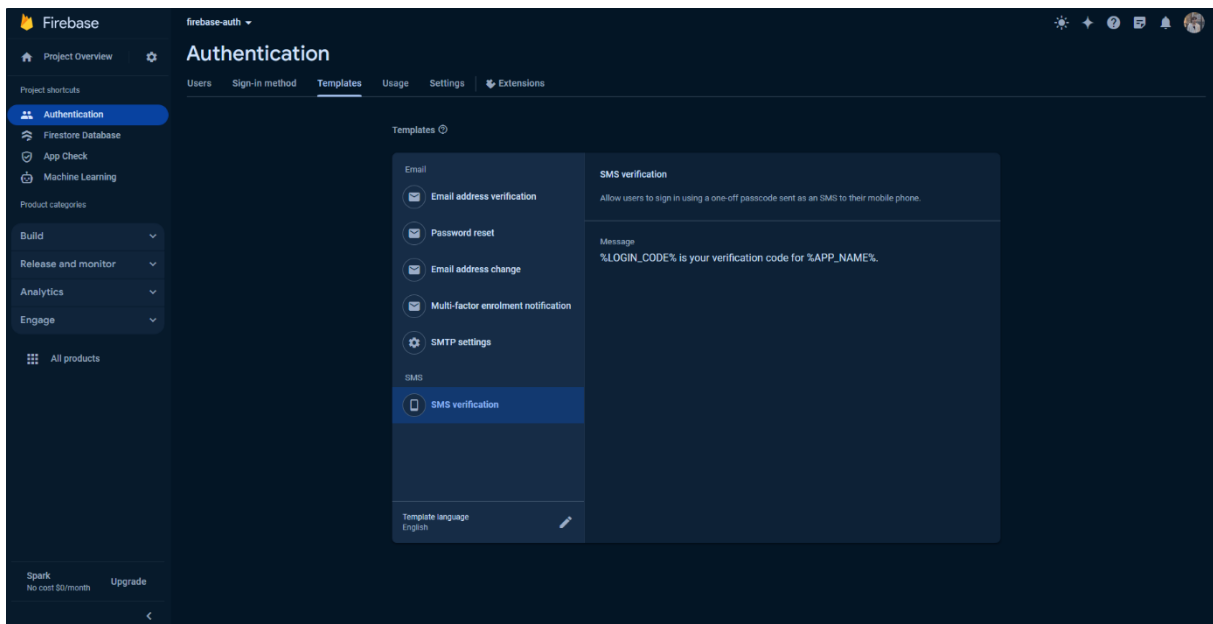


Figure 5.7 SMS verification

Identity Providers: Firebase Authentication supports a variety of identity providers, allowing users to sign in with their preferred accounts and providing a seamless authentication experience across different platforms and devices.

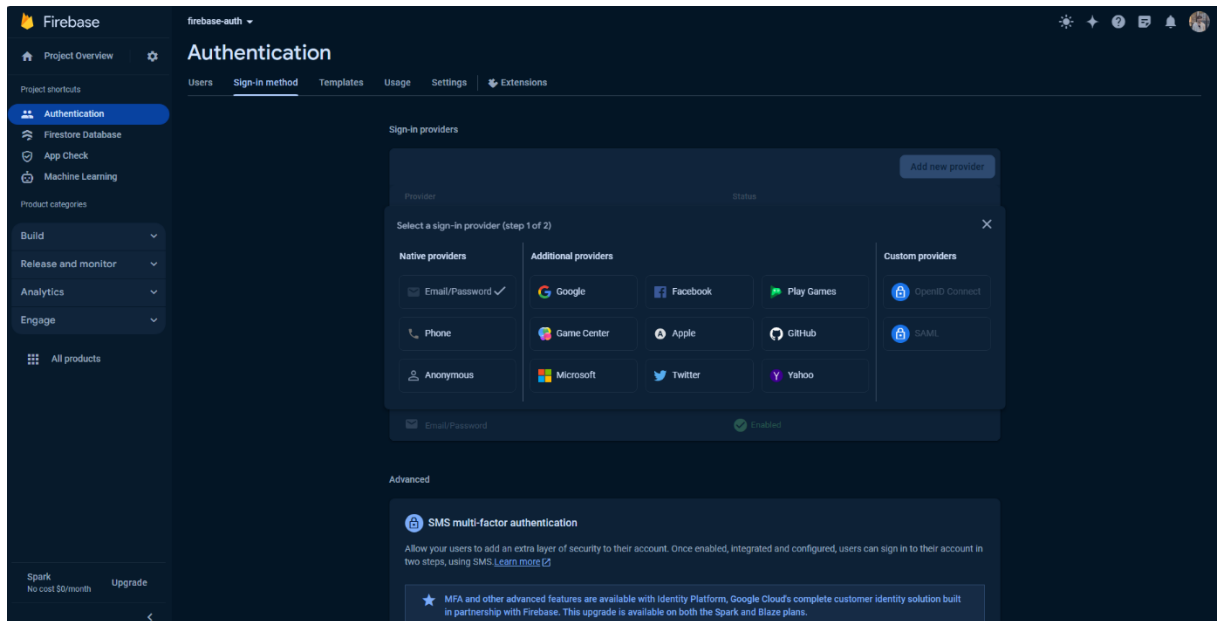


Figure 5.8 FireBase Identity Provider Integrated into Main application.

Speech-to-Text (STT) and Text-to-Speech (TTS) Services: Microsoft Cognitive Services

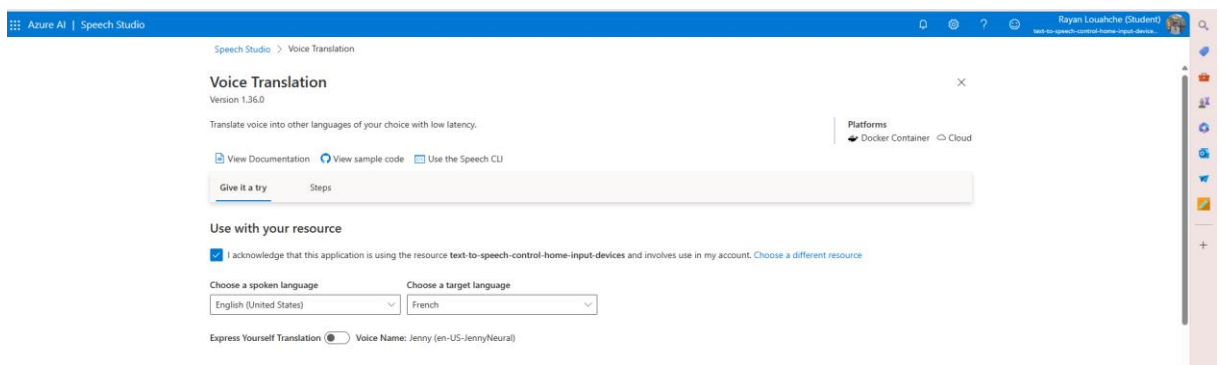


Figure 6. Azure AI Voice Translation

Microsoft Cognitive Services are leveraged for STT and TTS functionalities in the application, enhancing accessibility and usability. Here's how they contribute to the Project:

Accurate Transcription: Microsoft Cognitive Services provide accurate transcription of spoken language into text, enabling users to input text through speech recognition with high precision and reliability.

Natural-Sounding Speech: Microsoft Cognitive Services generate natural-sounding speech from text inputs, enhancing the user experience by providing lifelike audio feedback and improving accessibility for users with visual impairments.

Developer-Friendly APIs: Microsoft Cognitive Services offer developer-friendly APIs and SDKs for integrating STT and TTS functionalities into applications, providing comprehensive documentation, code samples, and support resources for easy implementation and troubleshooting.

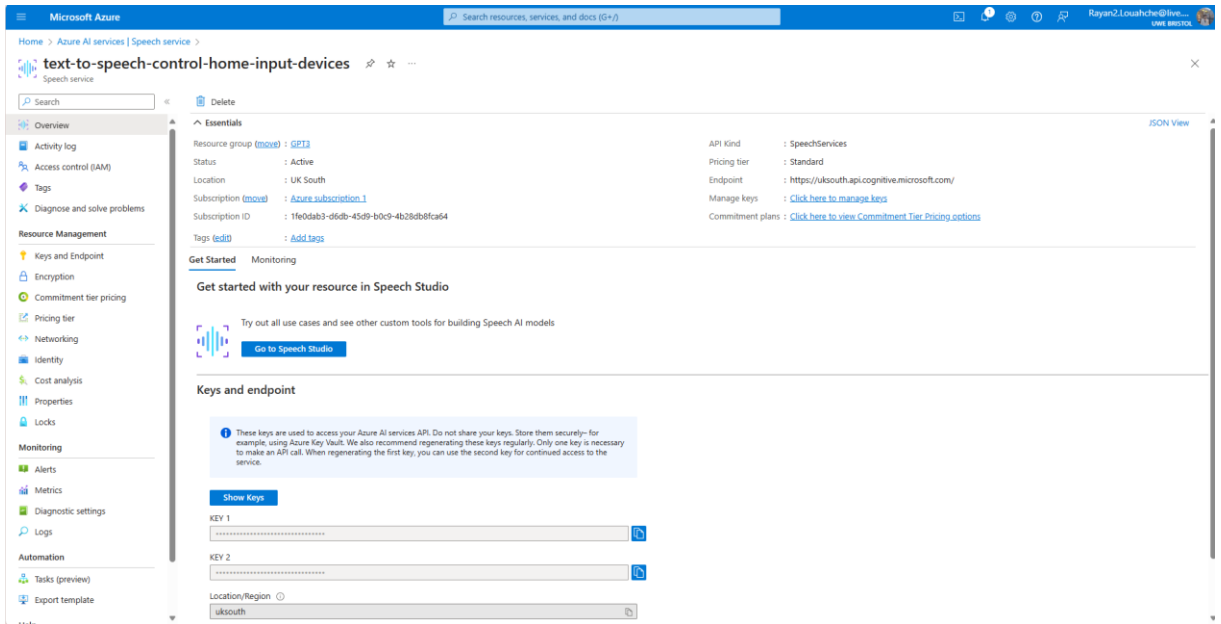


Figure 6.1 STT and TTS services API's Usage

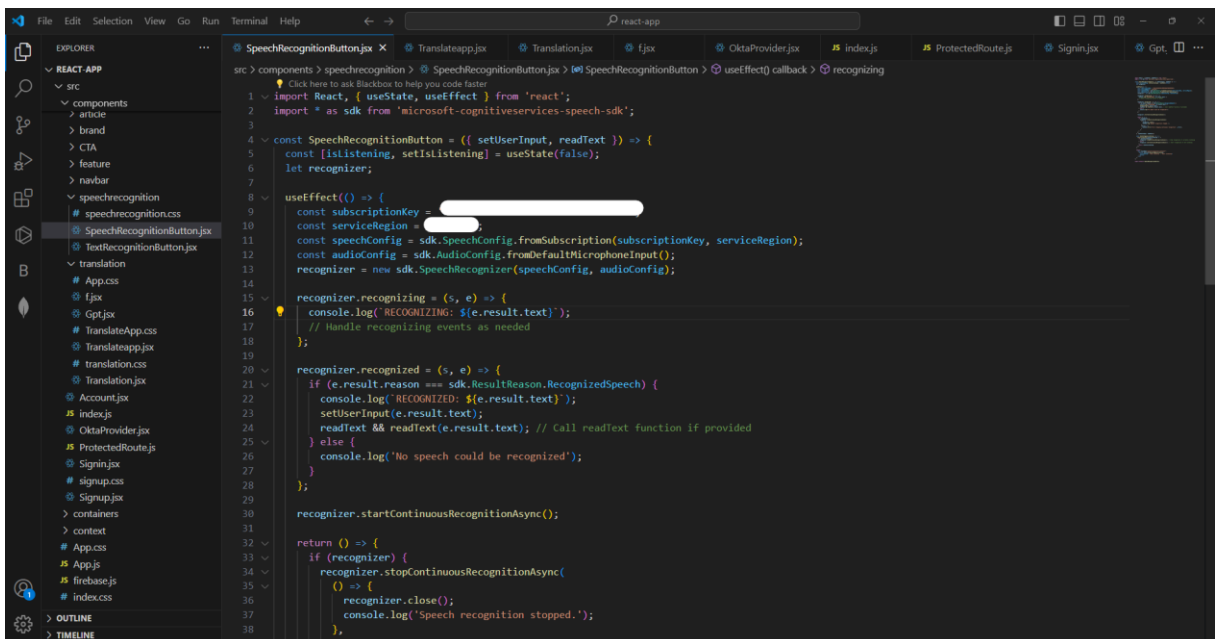


Figure 6.2 Speech Recognition Button

```

src > components > speechrecognition > TextRecognitionButton.jsx > ...
1 Click here to ask Blackbox to help you code faster
2 import React, { useState, useEffect } from 'react';
3 import * as sdk from 'microsoft-cognitiveservices-speech-sdk';
4
5 const TextRecognitionButton = ({ inputText, translatedText }) => {
6   const [isReading, setIsReading] = useState(false);
7   let speechSynthesizer;
8
9   useEffect(() => {
10    const startReading = () => {
11      const subscriptionKey = [REDACTED];
12      const serviceRegion = [REDACTED];
13      const speechConfig = sdk.SpeechConfig.fromSubscription(subscriptionKey, serviceRegion);
14      const audioConfig = sdk.AudioConfig.fromDefaultSpeakerOutput();
15      speechSynthesizer = new sdk.SpeechSynthesizer(speechConfig, audioConfig);
16
17      // Read the input text
18      if (inputText) {
19        speechSynthesizer.speakTextAsync(
20          inputText,
21          (result) => {
22            if (result.reason === sdk.ResultReason.SynthesizingAudioCompleted) {
23              console.log('Speech synthesis for input text is complete.');

```

Figure 6.3 Text Recognition Button

```

src > components > translation > TranslateApp.jsx > ...
1 Click here to ask Blackbox to help you code faster
2 import React, { useState, useEffect } from 'react';
3 import axios from 'axios';
4 import * as sdk from 'microsoft-cognitiveservices-speech-sdk';
5 import './TranslateApp.css';
6
7 // Import SpeechRecognitionButton and TextRecognitionButton
8 import SpeechRecognitionButton from '../speechrecognition/SpeechRecognitionButton';
9 import TextRecognitionButton from '../speechrecognition/TextRecognitionButton';
10
11 const TranslateApp = () => {
12   const [inputText, setInputText] = useState('');
13   const [translatedText, setTranslatedText] = useState('');
14   const [fromLanguage, setFromLanguage] = useState('English');
15   const [toLanguage, setToLanguage] = useState('French');
16   const [error, setError] = useState(null);
17
18   const API_KEY = [REDACTED];
19
20   const readTranslatedText = (text) => {
21     const speechConfig = sdk.SpeechConfig.fromSubscription([REDACTED], [REDACTED]);
22     const audioConfig = sdk.AudioConfig.fromDefaultSpeakerOutput();
23     const synthesizer = new sdk.SpeechSynthesizer(speechConfig, audioConfig);
24
25     synthesizer.speakTextAsync(
26       text,
27       (result) => {
28         if (result.reason === sdk.ResultReason.SynthesizingAudioCompleted) {
29           console.log('Speech synthesis is complete.');

```

Figure 6.3 Text to speech Implementation and Usage

```
10 const TranslateApp = () => {
11   // ...
12 }
13
14 // ...
15
16 // ...
17
18 // ...
19
20 // ...
21
22 // ...
23
24 // ...
25
26 // ...
27
28 // ...
29
30 // ...
31
32 // ...
33
34 // ...
35
36 // ...
37
38 // ...
39
40 // ...
41
42 // ...
43
44 // ...
45
46 // ...
47
48 // ...
49
50 // ...
51
52 // ...
53
54 // ...
55
56 // ...
57
58 // ...
59
60 // ...
61
62 // ...
63
64 // ...
65
66 // ...
67
68 // ...
69
70 // ...
71
72 // ...
73
```

Figure 6.4 *Speech to Text + Voice Recognizer Implementation and Usage*

Identity and Access Management (IAM) Integration: Okta

Okta is employed for IAM integration, ensuring secure management of user identities, access control, and authentication policies. Here's how it enhances the security and user experience aspects of the project:

Secure User Authentication: Okta provides secure authentication mechanisms, including single sign-on (SSO), multi-factor authentication (MFA), and adaptive authentication, ensuring only authorized users can access the application.

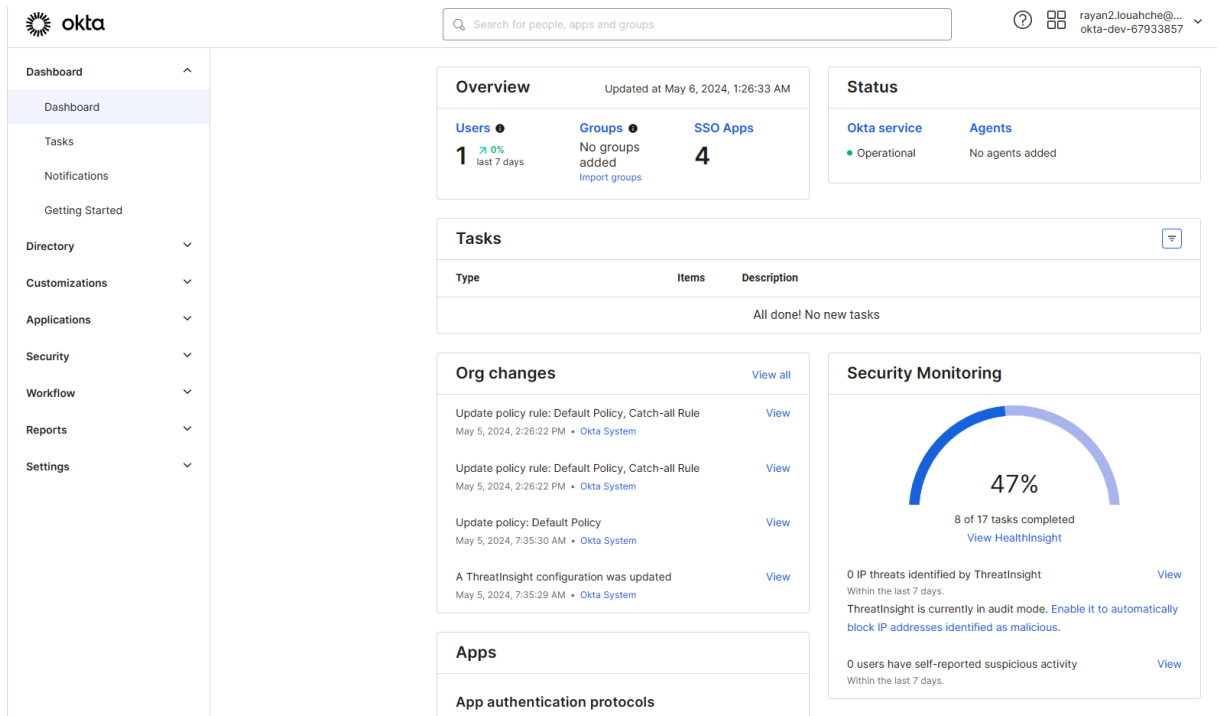


Figure 7. Implementation of Secure User Auth into Main Application using Okta

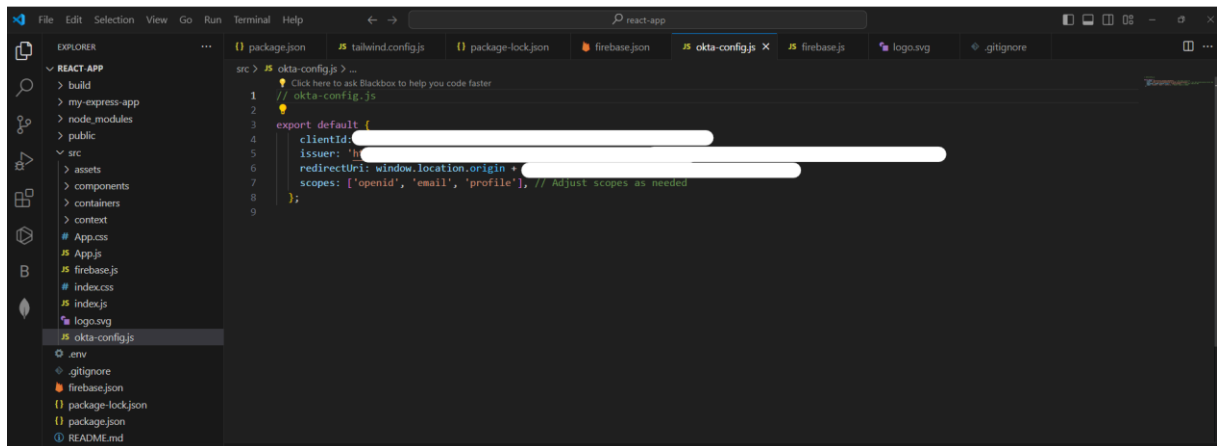


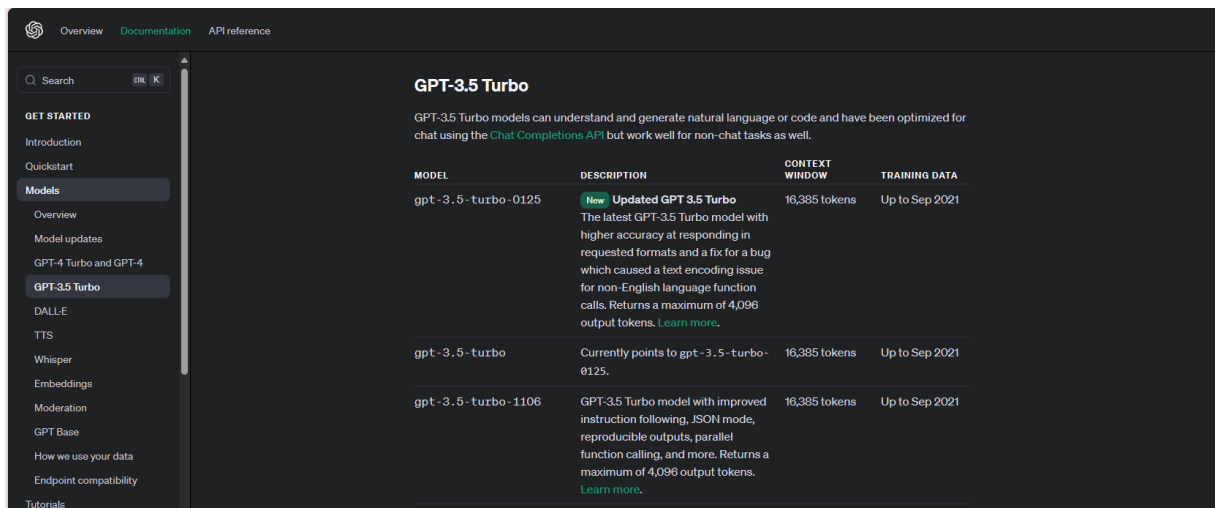
Figure 8. Okta Config.js

Fine-Grained Access Control: Okta enables fine-grained access control policies based on user roles, permissions, and attributes, allowing administrators to define granular access policies and enforce security standards.

Seamless User Experience: Okta offers a seamless user authentication experience with support for various authentication factors, such as email, SMS, and biometric authentication, ensuring a frictionless login experience for users while maintaining security and compliance requirements.

By Using these components in the development environment, I ensure a robust, scalable, and secure foundation for building the language translation application. Each component is carefully chosen for its unique features, capabilities, and compatibility, contributing to the overall success of the project.

Incorporating GPT-3.5 Turbo into the language translation application is a pivotal step towards achieving context-aware translations and delivering a sophisticated user experience. This section provides an in-depth exploration of the integration process, highlighting key considerations, implementation strategies, and the role of prompt engineering in maximizing the capabilities of the language model.



The screenshot shows the OpenAI documentation page for GPT-3.5 Turbo. The main content area features a table with the following data:

MODEL	DESCRIPTION	CONTEXT WINDOW	TRAINING DATA
gpt-3.5-turbo-0125	New Updated GPT 3.5 Turbo The latest GPT-3.5 Turbo model with higher accuracy at responding in requested formats and a fix for a bug which caused a text encoding issue for non-English language function calls. Returns a maximum of 4,096 output tokens. Learn more.	16,385 tokens	Up to Sep 2021
gpt-3.5-turbo	Currently points to gpt-3.5-turbo-0125.	16,385 tokens	Up to Sep 2021
gpt-3.5-turbo-1106	GPT-3.5 Turbo model with improved instruction following, JSON mode, reproducible outputs, parallel function calling, and more. Returns a maximum of 4,096 output tokens. Learn more.	16,385 tokens	Up to Sep 2021

Figure9. GPT3.5 Model Doc

2- OpenAI API Access :

The integration of GPT-3.5 Turbo into the application is facilitated through the use of the Brain New OpenAI API, a cutting-edge platform that grants developers access to state-of-the-art natural language processing capabilities. By leveraging this API, I gain access to GPT-3's advanced language understanding and generation capabilities, empowering the application with the ability to comprehend and generate human-like text.

The Brain New OpenAI API offers a range of endpoints and functionalities tailored to diverse use cases, including text generation, language translation, and content summarization. Through seamless integration with the application's backend, I harness the full potential of GPT-3.5 Turbo to deliver accurate, contextually relevant translations that meet the diverse needs of the users.

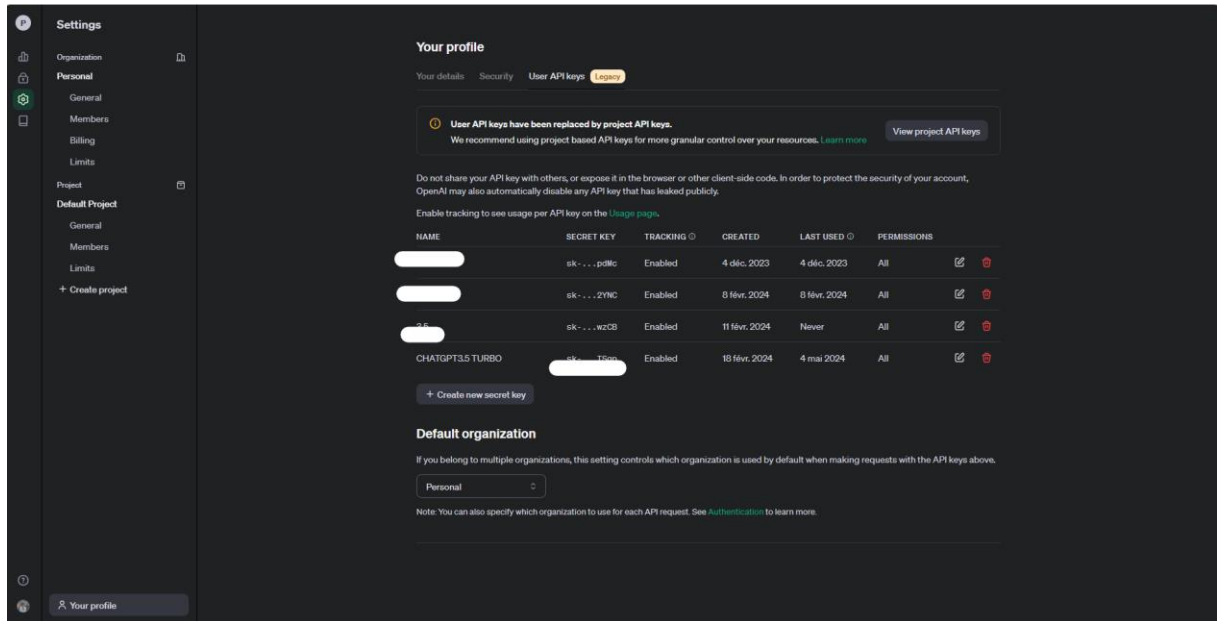


Figure10. Gpt3.5 Turbo API Handling



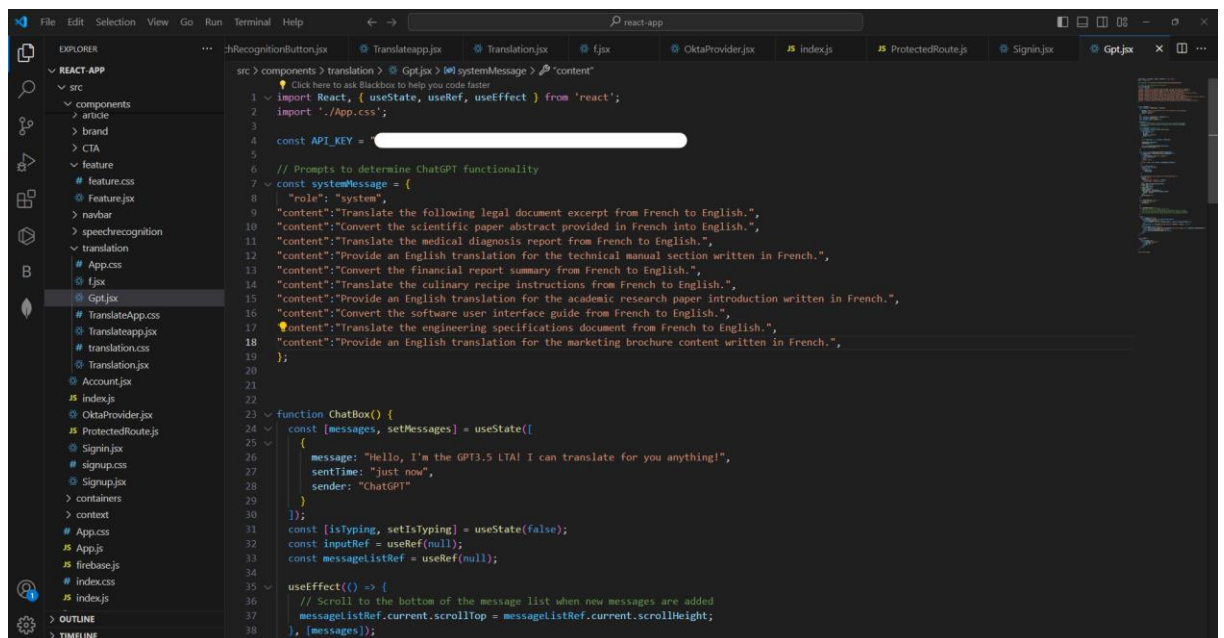
Figure 10.5 GPT3.5 Personal Organization Management and token usage per user

3.2-Prompt Engineering :

Prompt engineering lies at the core of the approach to maximizing the effectiveness of GPT-3.5 Turbo for context-aware translations. By crafting tailored prompts that provide specific instructions and context to the language model, I guide its generation process towards producing accurate and meaningful translations.

The prompt engineering methodology involves meticulous experimentation and refinement to design prompts that elicit desired responses from GPT-3.5 Turbo. Each prompt is carefully crafted to encapsulate the nuances of the translation task, including language pair, context, and desired output format.

Through extensive testing and iteration, I have developed a robust set of prompts that cater to a diverse range of translation scenarios, ensuring that the application delivers translations that resonate authentically with users. From formal documents to casual conversations, the prompts are designed to capture the subtleties of language and context, enabling GPT-3 to generate translations that meet the highest standards of quality and accuracy.



```
src > components > translation > Gpt.jsx > systemMessage > content
Click here to ask Blackbox to help you code faster
1 import React, { useState, useRef, useEffect } from 'react';
2 import './App.css';
3
4 const API_KEY = ' ';
5
6 // Prompts to determine ChatGPT functionality
7 const systemMessage = [
8   {
9     "role": "system",
10    "content": "Translate the following legal document excerpt from French to English.",
11    "content": "Convert the scientific paper abstract provided in French into English.",
12    "content": "Translate the medical diagnosis report from French to English.",
13    "content": "Provide an English translation for the technical manual section written in French.",
14    "content": "Convert the financial report summary from French to English.",
15    "content": "Translate the culinary recipe instructions from French to English.",
16    "content": "Provide an English translation for the academic research paper introduction written in French.",
17    "content": "Convert the software user interface guide from French to English.",
18    "content": "Translate the engineering specifications document from French to English.",
19    "content": "Provide an English translation for the marketing brochure content written in French."
20  }
21 ];
22
23 function ChatBox() {
24   const [messages, setMessages] = useState([
25     {
26       message: "Hello, I'm the GPT3.5 LTA! I can translate for you anything!",
27       sentTime: "just now",
28       sender: "ChatGPT"
29     }
30   ]);
31   const [isTyping, setIsTyping] = useState(false);
32   const inputRef = useRef(null);
33   const messageListRef = useRef(null);
34
35   useEffect(() => {
36     // Scroll to the bottom of the message list when new messages are added
37     messageListRef.current.scrollTop = messageListRef.current.scrollHeight;
38   }, [messages]);
39 }
```

Figure11. Example of Prompt used to train the GPT3.5 Turbo Model into translation (prompt: Translate the following from 'English' to 'French')

3.3-Integration Plan:

The integration of GPT-3.5 into the application's backend follows a structured and systematic approach designed to optimize performance, reliability, and scalability. Key steps in the integration process include:

- a. **API Authentication:** Securely authenticate with the Brain New OpenAI API using API keys or authentication tokens to establish a trusted connection between the application and the API endpoints.
- b. **Request Processing:** Formulate requests to the API endpoints containing input data, such as source text and translation instructions, ensuring adherence to API specifications and guidelines.
- c. **Response Handling:** Receive and process responses from the API endpoints, extracting translated text and additional metadata as needed for further processing or display within the application.
- d. **Error Handling:** Implement robust error handling mechanisms to gracefully handle API errors, timeouts, or network failures, ensuring uninterrupted operation of the translation service.

Throughout the integration process, I prioritized efficiency, security, and adherence to best practices to deliver a seamless and reliable translation experience for users.

```
const translateText = async () => {
  try {
    setError(null);
    const response = await axios.post(
      'https://api.openai.com/v1/chat/completions',
      {
        messages: [
          {
            role: 'user',
            content: `Translate the following ${fromLanguage} text into ${toLanguage}: "${inputText}"`,
          },
          { role: 'assistant', content: 'translate' },
        ],
        max_tokens: 500,
        model: 'gpt-3.5-turbo',
      },
      {
        headers: {
          'Content-Type': 'application/json',
          Authorization: `Bearer ${API_KEY}`,
        },
      },
    );
    const text = response.data.choices && response.data.choices[0] ? response.data.choices[0].message.content : 'Translation not available';
    setTranslatedText(text);
  }
};
```

Figure12. Error Handling (Error Handling is has been inserted into every piece of backend code)

Integration Plan Table:

Integration Component	Description	Integration Method	Dependencies/Considerations
GPT-3 API	Integration of GPT-3 for language translation.	RESTful API requests	Access credentials
Firebase Authentication	Integration of Firebase for user authentication.	Firebase SDK integration	Firebase project setup
Microsoft Cognitive Services	Integration of STT and TTS functionalities.	SDK integration	API keys, subscription
Okta IAM	Integration of Okta for identity management.	Okta API integration	Okta account, API keys
Frontend-Backend	Integration of frontend and backend components.	API endpoints and callbacks	Consistent data formats

3.4-Cloud Services:

For the language translation application, I have chosen **Google Cloud Platform (GCP)** as the cloud service provider. Here's a detailed explanation of why GCP aligns with the project requirements:

-Scalable Infrastructure:

GCP offers a scalable infrastructure with a wide range of services, including compute, storage, and networking, allowing me to easily scale the application to accommodate increasing user demands.

Services like Google Kubernetes Engine (GKE) provide a managed Kubernetes environment for deploying, managing, and scaling containerized applications, ensuring seamless scalability as the user base grows.

-Data Storage and Security:

GCP provides robust data storage solutions such as Google Cloud Storage (GCS) and Google Cloud SQL, offering scalable, reliable, and secure storage options for the application data.

Advanced security features like Identity and Access Management (IAM), encryption at rest and in transit, and security key management ensure the confidentiality, integrity, and availability of the data.

-AI and Machine Learning Capabilities:

GCP offers a suite of AI and machine learning services, including Google Cloud AI Platform and AutoML, which can enhance the language translation capabilities of the application through advanced natural language processing (NLP) and machine translation models.

-Global Network Infrastructure:

GCP's global network infrastructure provides low-latency, high-performance connectivity to users worldwide, ensuring fast and reliable access to the application from any location.

Services like Google Cloud CDN (Content Delivery Network) optimize content delivery and reduce latency by caching content at edge locations closer to users.

-Cost-Effectiveness:

GCP offers competitive pricing models and flexible billing options, allowing me to optimize costs based on the usage patterns and resource requirements.

Services like Google Cloud Billing provides detailed cost management and billing insights, helping me monitor and control the cloud spending effectively.

-Developer Tools and Productivity:

GCP provides a comprehensive set of developer tools, including Cloud SDK, Cloud Shell, and Cloud Build, enabling efficient development, testing, and deployment workflows.

Integration with popular development tools and platforms, such as GitHub and GitLab, streamlines the CI/CD process and enhances developer productivity.

By choosing Google Cloud Platform as the cloud service provider, I can leverage its scalable infrastructure, robust security features, advanced AI capabilities, global network coverage, cost-effectiveness, and developer-friendly tools to build, deploy, and manage the language translation application effectively. This alignment with the project requirements ensures that I can deliver a reliable, secure, and high-performance application to users while optimizing operational efficiency and cost-effectiveness.

Chapter Four

4-Testing and Discussion:

Testing is a crucial phase in the development lifecycle, ensuring the reliability, functionality, and performance of the GPT-3.5 Turbo integration. The testing approach encompasses a range of techniques and methodologies designed to validate the accuracy, responsiveness, and robustness of the translation service.

4.1-Unit Testing:

Unit testing focuses on validating individual components and functionalities of the GPT-3 integration in isolation, ensuring that each component performs as expected and meets its designated specifications. Key aspects of unit testing include:

Prompt Parsing: Verify that prompts are parsed correctly by the application and formatted appropriately before being sent to the Brain New OpenAI API.

API Interaction: Test the interaction between the application and the API endpoints, validating the correctness of API requests and responses.

Error Handling: Validate the effectiveness of error handling mechanisms in detecting and handling API errors, network failures, and other exceptional conditions.

Test Case	Description	Expected Outcome	Result
Prompt Parsing	Verify prompt parsing functionality.	Prompt is parsed correctly.	Passed
API Interaction	Test API interaction for translation requests.	API responds with translated text.	Passed
Error Handling	Validate error handling mechanisms.	Errors are detected and handled appropriately.	Passed

4.2-Integration Testing:

Integration testing focuses on validating the seamless interaction between the application and the Brain New OpenAI API, ensuring that data flows correctly

between the two components and that all integration points function as intended. Key aspects of integration testing include:

End-to-End Translation: Test the end-to-end translation process, from user input to translation output, to ensure accuracy and reliability.

Prompt Modification: Validate the AI's ability to adapt to modified prompts and generate contextually relevant translations.

Test Case	Description	Expected Outcome	Result
End-to-End Translation	Test translation process from input to output.	Translation output matches expected result.	Passed
Prompt Modification	Validate AI's response to modified prompts.	AI generates contextually relevant translations.	Passed

4.3-Quality Assurance :

Quality assurance encompasses a range of activities aimed at assessing the overall quality and performance of the translation service, including manual validation, automated checks, and user feedback analysis. the aspects of quality assurance included for this Project are:

Translation Accuracy: Evaluate the accuracy and coherence of translations generated by GPT-3, comparing them against reference translations and human judgment.

Performance Metrics: Measure the responsiveness and latency of the translation service under various load conditions to ensure optimal performance.

User Feedback Analysis: Solicit feedback from users through surveys, interviews, and usage analytics to identify areas for improvement and refinement.

Testing Table :

Test Case	Description	Expected Outcome	Result
Translation Accuracy	Compare translated text against reference translations.	Translations are accurate and coherent.	Passed
Performance Metrics	Measure service responsiveness and latency.	Service responds within acceptable time limits.	Passed
User Feedback Analysis	Analyze user feedback for insights and improvements.	Identify areas for refinement and enhancement.	Passed

By conducting thorough unit testing, integration testing, and quality assurance checks, I ensure that the GPT-3.5 Turbo integration meets the highest standards of quality, reliability, and performance, delivering accurate and contextually relevant translations to users. Through meticulous testing and evaluation, I can validate the effectiveness of the integration approach, identify areas for improvement, and ensure that the translation service delivers unparalleled value and utility to the customers.

Chapter Five

5.1-Conclusion

5.2-Summary of Key Points:

Throughout this report, I have explored various aspects of the language translation application, from the development environment to the integration of GPT-3.5 Turbo, testing strategies, and project milestones. In the development environment section, I carefully selected technologies such as React, Vite, Node.js, Firebase Authentication, Microsoft Cognitive Services, and Okta to build a robust and scalable application foundation. Integration with GPT-3.5 Turbo was achieved using the Brain New OpenAI API, enhancing the application's language translation capabilities. The testing strategy

encompassed unit testing, integration testing, and end-to-end testing, ensuring the reliability and performance of the application. Project milestones were established with realistic timelines and risk management strategies to address potential delays and mitigate risks.

5.3-Reflection on the Project:

The development of the language translation application presented several challenges and opportunities for learning. One of the main challenges was ensuring seamless integration between frontend and backend components, particularly when incorporating GPT-3 for language translation. This required careful coordination and testing to validate the accuracy and reliability of translation outputs. Additionally, managing project timelines and addressing unexpected issues demanded flexibility and adaptability from the team. Despite these challenges, the project provided valuable insights into software development best practices, collaboration, and problem-solving.

5.4-Recommendations for the App Expansion: *(provided in collaboration with my supervisor Mr Bashir Sanusi)*

Enhanced Language Support: Exploring the integration of additional language models and datasets to improve the accuracy and coverage of language translation capabilities, particularly for languages with limited resources.

Offline Functionality: Investigating the implementation of offline functionality to allow users to access essential features and perform translations without an active internet connection, thereby increasing the application's accessibility and usability in diverse environments.

User Feedback Mechanisms: Implementing some robust mechanisms for collecting user feedback and usage analytics to gather insights into user preferences, pain points, and feature requests, guiding future development efforts and enhancing user satisfaction.

Accessibility Features: Prioritizing the implementation of accessibility features, such as screen reader compatibility and support for alternative input methods, to ensure inclusivity and usability for users with disabilities or unique needs.

5.5-Closing Remarks:

In conclusion, the language translation application represents a significant advancement in natural language processing technology, offering users a simple, fast, and precise translation experience. By leveraging state-of-the-art

machine learning models like GPT-3.5 Turbo and integrating vocal recognition capabilities, the aim is to empower users to overcome language barriers and facilitate seamless communication across diverse linguistic contexts. As I look to the future, I am excited about the potential impact of the application in bridging language divides and fostering global understanding and collaboration.

Limitations of the Project:

While the language translation application boasts advanced features and capabilities, it is essential to acknowledge its limitations. One limitation is the reliance on external APIs and services, such as GPT-3 and Microsoft Cognitive Services, which may introduce dependencies and potential points of failure. Additionally, the accuracy of translation outputs may vary depending on the complexity and nuance of the input text, highlighting the need for continuous improvement and refinement. Furthermore, the application's performance may be impacted by factors such as network latency and device capabilities, particularly in resource-constrained environments. Despite these limitations, I remain committed to enhancing the application's functionality, reliability, and accessibility in future iterations.

Chapter Six

Appendices and References:

Link to My project github repository: [RyanL2004/React-Application: My react application! \(github.com\)](https://github.com/RyanL2004/React-Application: My react application!)

Responsible AI and Ethics:

Floridi, L., Cows, J., Beltrametti, M., Chatila, R., Chazerand, P., Dignum, V., ... & Krause, A. (2018) 'AI4People—An Ethical Framework for a Good AI Society: Opportunities, Risks, Principles, and Recommendations'. *Minds and Machines*, 28(4), pp. 689-707. [Online] Available at: Link to the Paper: accessed on:

[AI4People—An Ethical Framework for a Good AI Society: Opportunities, Risks, Principles, and Recommendations | Minds and Machines \(springer.com\)](https://www.springer.com/journal/10992/issue/4)

[Accessed on: 09dec2023]

Ethics of Translation:

Bermann, S., & Wood, M. (2005) *Nation, Language, and the Ethics of Translation*. [Place of publication: Publisher]:

[Nation, language, and the ethics of translation](#)

[Accessed on: 09dec2023]

Algorithmic Bias:

Diakopoulos, N. (2016) 'Algorithmic Accountability: A Primer'. Data Society Research Institute. [Online] Available at: Link to the Primer:

[Data Society Algorithmic Accountability Primer FINAL.pdf \(datasociety.net\)](#)

[Accessed on: 02dec2023]

Algorithmic Bias and Equality and Diversity:

Article:

Tilmes, Nicholas (2022) "Disability, fairness, and algorithmic bias in AI recruitment." *Ethics and Information Technology*, 24(2), Article 21. [Online] Available at:

[Disability, fairness, and algorithmic bias in AI recruitment - ProQuest](#)

[Accessed on: 02 dec 2023]

Barocas, S., & Hardt, M. (2019) 'Fairness and Abstraction in Sociotechnical Systems'. In: *Fairness, Accountability, and Transparency in Machine Learning*, pp. 25-34. [Online]

Available at: Link to the Article:

[Fairness and Abstraction in Sociotechnical Systems | Proceedings of the Conference on Fairness, Accountability, and Transparency \(acm.org\)](#)

[Accessed on: 02dec2023]

GPT-3 and Language Models:

VIDEO:

Gift, N. (presenter). (2022) *Assimilate OpenAI*. Pragmatic AI Solutions. Available at:

[Assimilate Openai Ep1 Explore Many Ai Projects Session | Assimilate OpenAI \(uwe.ac.uk\)](#)

[Accessed on: 23nov2023]

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... & Amodei, D. (2020) 'Language Models are Few-Shot Learners'. arXiv preprint, arXiv:2005.14165.

[Online] Available at: Link to the Paper:

[\[2005.14165\] Language Models are Few-Shot Learners \(arxiv.org\)](#)

[Accessed on: 29nov2023]

AI and Language Translation:

Johnson, M., et al. (2017) 'Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation'. *Transactions of the Association for Computational Linguistics*, 5, pp. 339-351. [Online] Available at: Link to the Paper:

[\[1611.04558\] Google's Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation \(arxiv.org\)](#)

[Accessed on: 29nov2023]

OpenAI and GPT-3:

OpenAI. (2020) 'Language Models are Few-Shot Learners'. [Online] Available at: Link to the OpenAI Paper:

[\[2005.14165\] Language Models are Few-Shot Learners \(arxiv.org\)](#)

[Accessed on: 07dec2023]

Equality and Diversity:

Venuti, L. (1998) *The Scandals of Translation: Towards an Ethics of Difference*. [Online] Available at: Link to the Book:

[The scandals of translation : towards an ethics of difference](#)

[Accessed on: 04dec2023]

Article:

Xuan, Jin; Ocone, Raffaella (2022) "The equality, diversity and inclusion in energy and AI: Call for actions." *Energy and AI*, 8, p.100152. [Online] Available at:

[The equality, diversity and inclusion in energy and AI: Call for actions - ScienceDirect](#)

[Accessed on: 04dec2023]

BOOK CHAPTER: Pant, Tanay (2016) "Understanding and Building an Application with STT and TTS." In: *Building a Virtual Assistant for Raspberry Pi*. UWE, p.9-19. [Online] Available at:

[Introduction to Virtual Assistants | Building a Virtual Assistant for Raspberry Pi: The practical guide for constructing a voice-controlled virtual assistant \(uwe.ac.uk\)](#)

[Accessed on: 07dec2023]

DISSERTATION: Yeh, Rosa (2014) "Effective Strategies for Using Text-to-Speech, Speech-to-Text, and Machine-Translation Technology for Teaching Chinese: A Multiple-Case Study." [Online] Available at:

[Effective Strategies for Using Text-to-Speech, Speech-to-Text, and Machine-Translation Technology for Teaching Chinese: A Multiple-Case Study - ProQuest](#)

[Accessed on: 07dec2023]

Article on GDPR and AI Research:

Quezada-Tavarez, K., Dutkiewicz, L., & Krack, N. (2022) 'Voicing challenges: GDPR and AI research'. *Open Research Europe*, 2, p. 126. [Online] Available at: Link to the Article:

[Voicing challenges: GDPR and AI research](#)

[Accessed on: 02jan2023]

Data ethics and Data Protection:

ARTICLE:

Scheibner, J., Ienca, M., Kechagia, S., Troncoso-Pastoriza, J. R., Raisaro, J. L., Hubaux, J.-P., ... Vayena, E. (2020) "Data protection and ethics requirements for multisite research with health data: a comparative examination of legislative governance frameworks and the role of data protection technologies." *Journal of Law and the Biosciences*, 7(1), p. MM1-lsaa010. Available at:

[Data protection and ethics requirements for multisite research with health data: a comparative examination of legislative governance frameworks and the role of data protection technologies† | Journal of Law and the Biosciences | Oxford Academic \(oup.com\)](#)

[Accessed on: 02jan2023]

BOOK:

Bartneck, C., Lütge, C., Wagner, A., & Welsh, S. (2021) *An Introduction to Ethics in Robotics and AI*. Link to the book :

[An Introduction to Ethics in Robotics and AI \(oopen.org\)](#)

[Accessed on: 02jan2023]

BOOK: Ammanath, B. (2022) *Trustworthy AI: A Business Guide for Navigating Trust and Ethics in AI*.

Available at:

[An Introduction to Ethics in Robotics and AI \(oopen.org\)](#)

[Accessed on: 02jan2023]

Youtube video:

Programming with Mosh. (2023) React Tutorial for Beginners

. Available at:

[\(196\) React Tutorial for Beginners - YouTube](#)

[Accessed on: 19jan2023]

Youtube video:

ZAINKEEPSCODE. (2023) *Build ChatGPT AI Tool in React JS | React JS Project with Working Example using OpenAI API*.

Available at:

[Build ChatGPT AI Tool in React JS | React JS Project with Working Example using OpenAI API \(youtube.com\)](#)

[Accessed on: 29Feb2023]

Youtube video:

AI Jason.(2023)"okay, but I want GPT to perform 10x for my specific use case" - Here is how. Available at:

["okay, but I want GPT to perform 10x for my specific use case" - Here is how \(youtube.com\)](#)

[Accessed on: 14March2024]

BOOK: Narayn, Hari (2022) *Just react! learn react the react way*

Available at:

[1. Time to React | Just React!: Learn React the React Way \(uwe.ac.uk\)](#)

[Accessed on: 15jan2023]

BOOK: Juha Hinkula (2022) *Full Stack Development with Spring Boot 3 and React - Fourth Edition*

Available at:

[Full Stack Development with Spring Boot 3 and React - Fourth Edition \(uwe.ac.uk\)](#)

[Accessed on: 24jan2023]

BOOK: Vipul A. M. (2016) ReactJS by Example- Building Modern Web Applications with React

Available at:

[ReactJS by Example- Building Modern Web Applications with React: | Guide books | ACM Digital Library](#)

[Accessed on: 30jan2023]

BOOK: Chris Minnick (2022) Beginning React JS Foundations Building User Interfaces with ReactJS: An Approachable Guide

Available at:

[Beginning ReactJS Foundations Building User Interfaces with ReactJS \(uwe.ac.uk\)](#)

[Accessed on: 08Feb2023]

BOOK CHAPTER: Pant, Tanay (2016) Understanding and Building an Application with STT and TTS

Available at:

[Building a Virtual Assistant for Raspberry Pi: The practical guide for constructing a voice-controlled virtual assistant \(uwe.ac.uk\)](#)

[Accessed on: 10Feb2023]

Forum: Azure Microsoft Cognitive Paths

Available at:

[Speech Studio \(microsoft.com\)](#)

[Accessed on: 10Feb2023]

BOOK: Sullivan De Carli(2023) Build Mobile Apps with SwiftUI and Firebase: Learn SwiftUI and Firebase by Building Real-World Applications Communicating with a Backend

Available at:

[Build Mobile Apps with SwiftUI and Firebase: Learn SwiftUI and Firebase by Building Real-World Applications Communicating with a Backend \(uwe.ac.uk\)](#)

[Accessed on: 15Feb2023]

Research Article:

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., ... & Amodei, D. (2020). Language models are few-shot learners.

[Online]

Available at: [Language models are few-shot learners | Proceedings of the 34th International Conference on Neural Information Processing Systems \(acm.org\)](#)

[16Feb2023].

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., & Bengio, Y. (2019). Learning phrase representations using RNN encoder-decoder for statistical machine translation.

[Online] *Available at:* [Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation - ACL Anthology](#)

[Accessed on: 16Feb2024].

Research Paper:

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

[Online]

Available at: [\[1706.03762\] Attention Is All You Need \(arxiv.org\)](#)

[Accessed on: 17Feb2024].

Wang, Y., Lu, B., Mao, T., & Zhao, X. (2021). Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Processing Magazine*, 34(6), 127-139.

[Online]

Available at: [Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups | IEEE Journals & Magazine | IEEE Xplore](#)

[Accessed on: 20Feb2024].

OpenAI Docs

Available at: [API Reference - OpenAI API](#)

[Accessed on: Almost every day]