SRH Hochschule
Heidelberg

AI SUPPORTED LOCAL LANGUAGE TRAINING

Prepared by
Vedant Zalke
Rehaan Khan
Sumit Chaudhari
Alan K Mathew

Under guidance of
Prof. Dr. Gerd Moeckel
Prof. Dr. Josef Hermanns

06.05.2023

CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

In the age of globalisation and communication, the destiny of people, organisations, societies, and countries largely hinges on their capacity for successful interpersonal communication. And for this a lingual transparency is must. The act of communicating thoughts, feelings, information, knowledge, facts, views, and emotions with another person is known as communication. Effective communication is delivering the appropriate message through the appropriate channel at the appropriate moment to the appropriate recipient in order to alter the recipient's understanding, attitude, and behaviour. People are unable to comprehend one another due to language limitations. They frequently provide difficulties for people, teams, multinational corporations, governments, countries, and the whole planet. The research we did that a number of variables prevent or severely impede efficient communication. They result from varying word meanings and uses, symbols, images, gesture, languages, dialects, accents, linguistic ability, technical terminology or jargon, voice volume, ambiguous word use, incorrect translation, incorrect interpretation, misunderstanding, and complexity of messages, as well as from differences in the linguistic abilities of the sender and the receiver. They also result from poorly understood and inadequately explained words and messages.

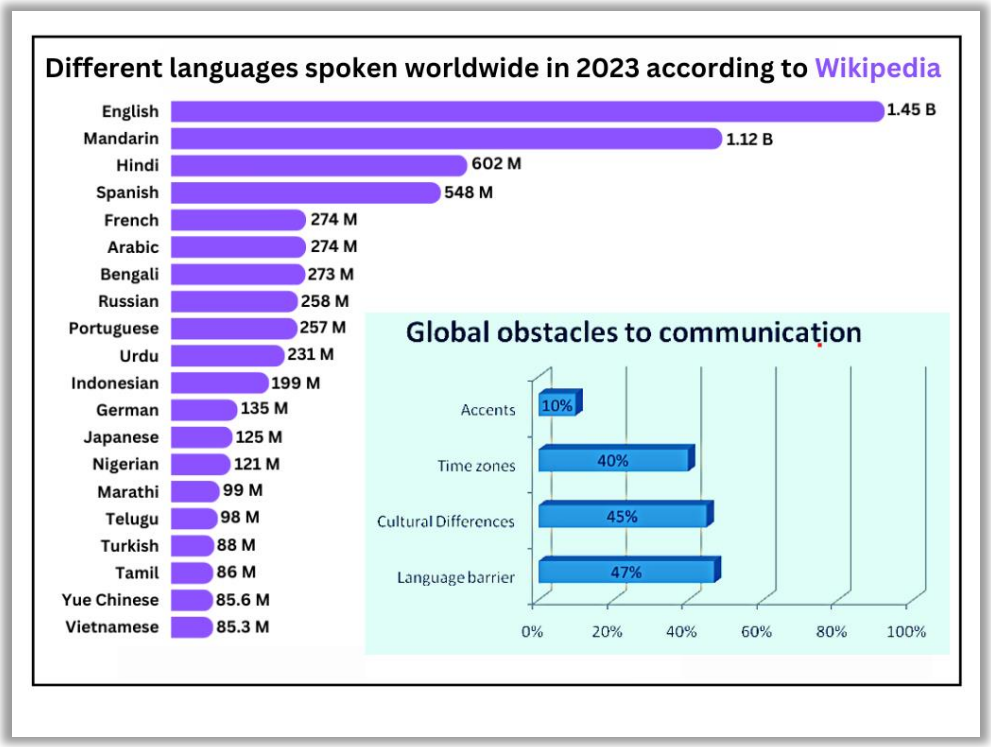

Figure 1: Different languages spoken worldwide with their obstacles to communicate

Today's AI-supported application is a ground-breaking method for bridging the gap between various languages and dialects. Understanding and using regional dialects to communicate has grown in importance as a result of globalization and greater mobility, which has made communication more crucial than ever. It is necessary to assist people in achieving this aim by offering a thorough and efficient learning environment that is simple to use and takes use of the most recent technological developments.

Tools for AI language translation come in a wide variety, each with unique capabilities and features. Elsa Speak, Google Translate, and Duo Lingo are a few well-known instances. These tools are used by individuals and organisations throughout the world to develop their language abilities for learning, employment, and leisure.

Our research intends to create an AI-supported dialect and language translation tool that can make it simple for people to pick up and comprehend regional languages. To deliver a smooth and efficient learning experience, the solution will make use of cutting-edge technologies. To guarantee that the solution is capable of effectively capturing the intricacies of regional dialects, our team will collaborate with linguists and native speakers. No matter where they are located or their level of socioeconomic standing, everyone will be able to take advantage of the solution since it will be created to be user-friendly, affordable, and accessible.

- **Task Description**:-

  AI-supported language instruction or learning to comprehend regional dialects. Learning a new language is never easy, but using it to communicate with native speakers is a whole other ballgame. The official language and regional dialects may have markedly different pronunciations, spellings, and even meanings.

The EU has set up a fund to assist speech-to-text, text-to-speech, text-to-text, and speech-to-speech translation in order to facilitate and improve comprehension. This involves taking into account regional dialects. The system should also facilitate the teaching of regional languages and dialects in addition to offering translation services. The programme aims to assist individuals in their daily lives and offer qualified translation services to organisations or governments.

Purpose

Language barriers can result in communication problems like misunderstanding, misinterpretation of messages, distorted messages, misinformation, confusion, mistrust, uncertainty, and frustration. They can also cause weak and incorrect feedback, aviation and maritime accidents and disasters, fatalities, tension, conflict, and interpersonal violence. They may result in elements that obstruct clear communication. The best way to promote effective communication and convey messages is to overcome language barriers by employing bilingual staff and qualified and licensed translators, minimising the use of jargon, avoiding ambiguous words, regional dialect, and, confirming understanding, using translation machines, avoiding shouting, and giving and receiving feedback.

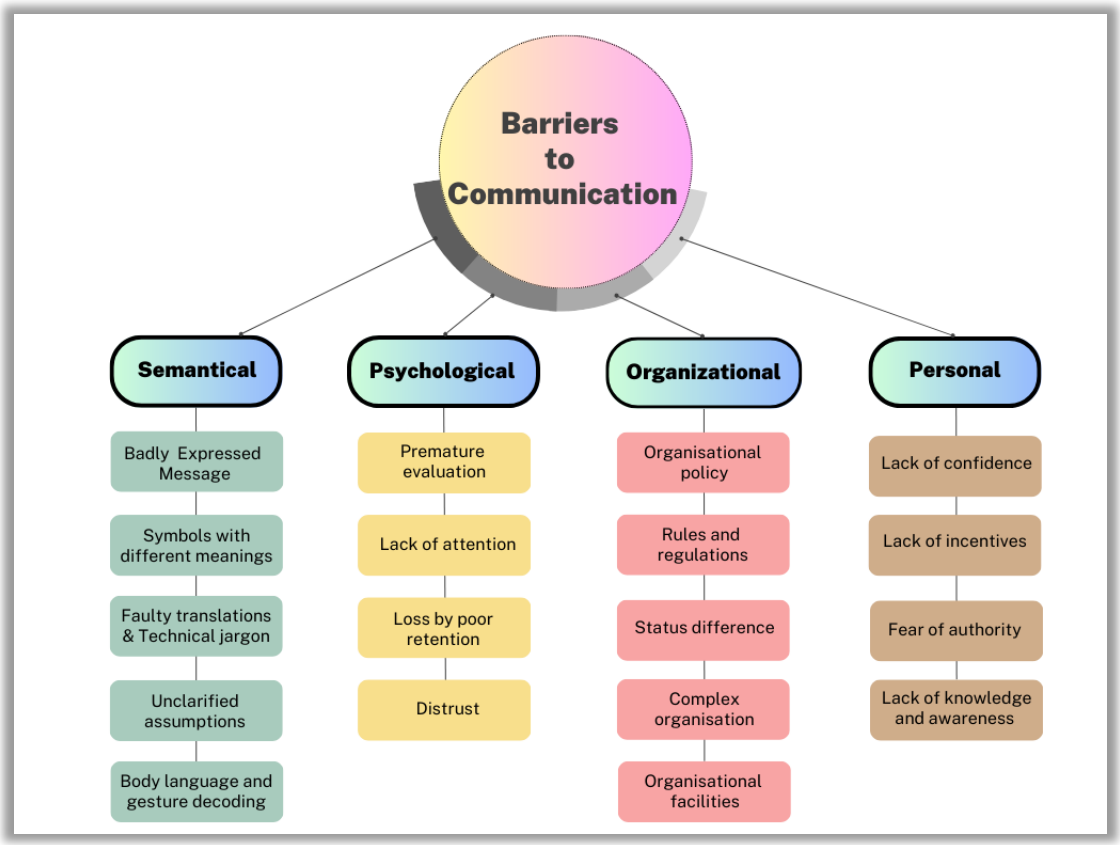

Figure 2:Types of Language barrier problems

Persona

Tobias Schmidt is a college student from Konstanz, a town in southern Germany where Swiss German is commonly spoken. However, he is now studying in a college located in a different region of Germany where

people speak a Thuringia dialect, which is quite different from what he is used to. This language barrier is causing several difficulties in his day-to-day life.

In the college, Tobias finds it difficult to follow lectures and understand the course material due to the dialect difference. He often has to ask his professors to repeat themselves or explain things in a different way, which can be quite embarrassing for him. This also makes it challenging for him to participate in class discussions and group projects.

In banks, Tobias struggles to understand the banking terminologies and instructions in the Thuringia dialect. He often has to spend more time deciphering the text or asking for help, which leaves him with less time to focus on his studies.

In the market, Tobias has difficulty communicating with the vendors and understanding the prices of items due to the dialect difference. He sometimes ends up buying things he doesn't need or paying more than he should, which can be frustrating.

In the hospital, Tobias finds it hard to understand the medical terminologies and instructions given by the doctors and nurses. This language barrier can sometimes lead to misunderstandings or mistakes in his medical care.

In sports, Tobias feels left out and excluded from the team due to the language barrier. He struggles to communicate with his teammates and often feels like he is not part of the group. This makes him feel isolated and demotivated, which affects his performance.

In the library, Tobias finds it challenging to understand the academic literature and research papers written in the Thuringia dialect. He often has to spend more time deciphering the text, which leaves him with less time to study and complete his assignments.

In the canteen, Tobias has difficulty ordering food as he is not familiar with the names of dishes and ingredients in the local dialect. This often results in him ordering something he does not like or cannot eat, which can be frustrating and disappointing.

In terms of friends, Tobias finds it challenging to socialize with his classmates outside of class as they speak the Thuringia dialect amongst themselves. He often feels excluded from conversations and gatherings and misses the camaraderie he had with his friends back home.

In his hostel, Tobias struggles to communicate with his roommates and other residents in the building. He often finds it hard to participate in group activities and discussions, which leaves him feeling lonely and disconnected from his peers.

Point of View

Due to the dialectal differences between his birthplace and college, Tobias has a number of challenges in his day-to-day activities. In a variety of contexts, including the college, banks, market, hospital, sports, library, canteen, friends, and hostel, he finds it difficult to interact with others. His mental health and academic performance are being negatively impacted by these difficulties, and he needs compassion and assistance from everyone around him.

To help Tobias overcome the language barrier and improve his daily life, an AI-powered local language and dialect training and translation tool, bot, or website may be developed. Tobias might take use of a number of features that the app, bot, or website could offer to help him understand Thuringian dialect and communicate more effectively in a range of settings.

Overall, using a local language and dialect training and translation app, bot, or website with AI support may be beneficial for Tobias and other students who have comparable language barriers. By providing specialist language study courses, translation tools, voice recognition technology, configurable settings, and community support, this program may help Tobias improve his communication skills and acquire more confidence in daily life.

EPIC AND USER STORIES

What is an EPIC?

An epic is a high-level user story that describes a large and complex feature or functionality that cannot be completed in a single iteration of agile development.

Epics are used to break down large and complex projects into smaller, more manageable pieces that can be worked on in increments. They provide a high-level view of the system requirements and help to align the development team with the overall goals and objectives of the project.

What is a USER STORY?

A user story is a simple, short description of a feature or functionality of a software system, told from the perspective of the end user or customer. It is a tool used in agile software development that helps to define the requirements of the system in a way that is easy to understand and implement.

A user story typically consists of three elements:

- The user or customer who needs the feature

- The specific feature or functionality that the user needs
- The reason why the user needs that feature or functionality

#US001

As a	Traveller
I want to	be able to quickly translate signs and menus in other dialects
So that	I can navigate and order food without difficulty.

#US002

As a	Student
I want to	Be able to translate different dialects in documents
So that	I can understand the material and learn more effectively.

#US003

As a	Business Person
I want to	Be able to quickly translate signs and menus in different dialects
So that	I can navigate and order food without difficulty.

#US004

As a	Person who wants to communicate with family members or friends who speak different dialects
I want to	Be able to translate text messages and chat conversations easily and accurately.

#US005

As a	Person with hearing loss
I want to	be able to use the translation app to transcribe speech in one dialect into written text in another dialect
So that	I can understand and communicate with others more easily.

#US006

As a	Teacher
I want to	be able to use the translation app to help my students who speak different dialects understand the material I am teaching
So that	They can learn more effectively.

#US007

As a	Doctor
I want to	be able to communicate with patients who speak different dialects
So that	I can understand their symptoms and prescribe medicine.

#US008

As a	Customer Service Representative

I want to	be able to communicate with customers who speak different dialects
So that	I can make them trust me and increase daily sales and increase loyal customer base.

Influencing Factors :-

- **User Needs and Preferences**: Understanding the needs and preferences of the application users is crucial in developing a solution that meets their expectations. User feedback and testing can help identify the most critical features and functionalities required to provide an effective and engaging learning experience.
- **Technology**: The application's success depends on the technology utilized to provide dialect training and translation. Utilizing advanced technologies such as natural language processing, machine learning, speech recognition, and voice synthesis can provide an efficient and effective learning experience.
- **Data Availability and Quality**: High-quality data is essential to develop machine learning models that provide accurate and reliable results. Therefore, the availability and quality of training data is a significant factor in the application's success.
- **Resources**: Developing an AI-supported dialect training and translation application requires a significant investment of resources, including time, expertise, and funding. Ensuring that sufficient resources are available throughout the development process is crucial to delivering a high-quality solution.
- **User Interface and User Experience**: The application's user interface and user experience design play a crucial role in the success of the solution. A well-designed interface that is intuitive and user-friendly can help improve engagement and learning outcomes.
- **Ethical Considerations**: Developing an AI-supported dialect training and translation application also requires ethical considerations, such as data privacy and security, bias mitigation, and fair use of the technology.
- **Legal Requirements**: Compliance with legal requirements, such as copyright laws and data protection regulations, is essential in the development and deployment of the application.
- **Cultural Differences**: Understanding the cultural nuances of different dialects is essential to developing a solution that is effective in facilitating communication. Cultural differences can affect the way people communicate, and the application should be designed to accommodate these differences.
- **Language Complexity**: Some dialects and languages may be more complex than others, making it more challenging to develop an effective solution. The complexity of the language can affect the accuracy of the translation and the effectiveness of the dialect training.

- **Availability of Experts**: Access to experts in the dialect or language being taught is essential to ensure that the content is accurate and culturally relevant. The availability of experts can vary depending on the dialect and the region, which can affect the development process.
- **Market Demand**: The demand for AI-supported dialect training and translation solutions can vary depending on the region, industry, and target audience. Understanding the market demand is essential to developing a solution that is relevant and useful to the target audience.
- **Integration with Existing Systems**: The application may need to integrate with existing systems, such as language learning platforms, translation tools, or communication software. Ensuring that the application can seamlessly integrate with these systems is critical to its success.
- **Competition**: There may be other solutions available in the market that offer similar functionalities. Understanding the competition and identifying the unique selling points of the application is crucial to its success.
- **Accessibility**: The application should be designed to be accessible to a wide range of users, including those with disabilities or who may have limited access to technology.
- **Language Standards**: Some dialects may not have standardized writing or grammar rules, which can affect the development of a language training and translation application.
- **Language Variations**: Dialects may have variations depending on the region or context in which they are used. The application should be designed to accommodate these variations and provide accurate translations and dialect training.
- **Data Privacy and Security**: The application may collect and store sensitive user data, such as audio recordings or personal information. Ensuring that the data is secure and that user privacy is protected is essential to building trust with users.
- **Scalability**: The application should be designed to be scalable, meaning that it can accommodate increasing numbers of users and data without sacrificing performance or functionality.
- **Budget**: Developing an AI-supported dialect training and translation application can be expensive, and the budget available can affect the development process and the features that can be included in the final solution.
- **Timeframe**: The timeframe available for development and deployment can also affect the development process and the features that can be included in the application.
- **Language Policy**: The language policy of the region or country where the application will be used can affect the development process and the content that can be included in the application.

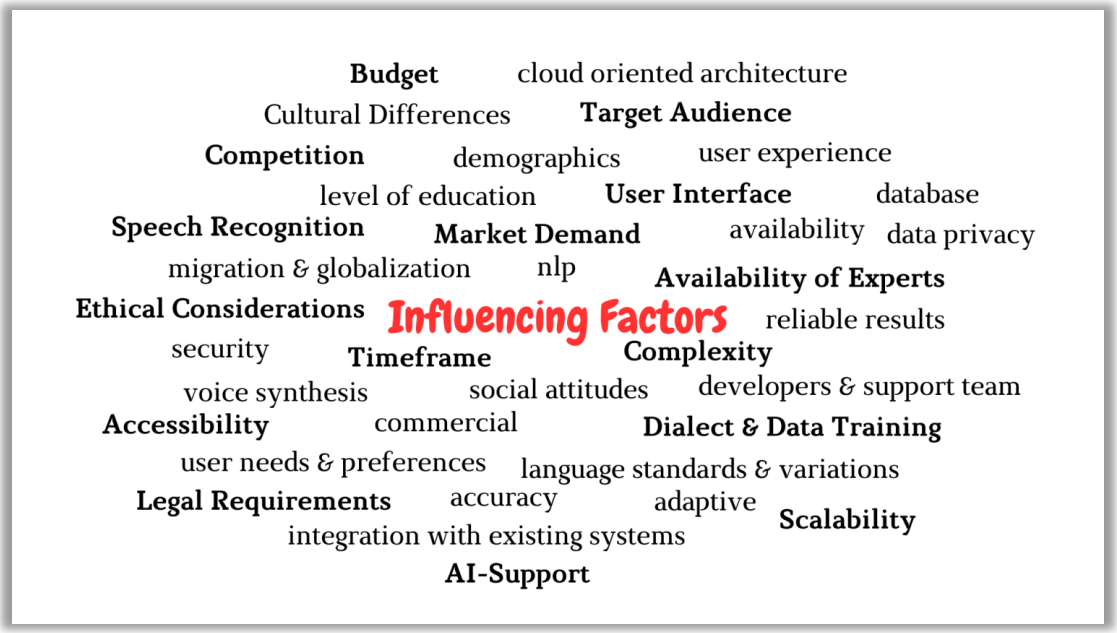

Figure 3: Influencing factors related to project goals

Project Description:

The goal of this project is to develop an AI-supported local language and dialect's training and translation program to help language learners like Tobias Schmidt overcome difficulties in day-to-day life due to dialect differences. The program will focus on translating between Swiss German and Thuringia dialect and provide language learning tools.

The following characteristics might be among them:

- **Language learning modules**: The app, bot, or website could provide language instruction that is particularly created to educate pupils like Tobias the Thuringia dialect. These modules could contain interactive activities, audio and video courses, quizzes to check his comprehension, and interactive exercises.
- **Tools for translation**: Tobias may be able to convert words and phrases from the Thuringia dialect to his own Swiss German dialect or other languages using the app, bot, or website's translation tools. In places like banks, hospitals, marketplaces, and libraries where he could run into strange terms or phrases, this could be especially helpful.
- **Voice recognition technology**: Tobias might utilise the app, bot, or website to employ voice recognition technology to work on his Thuringia pronunciation and intonation. This may be especially helpful in situations like sports when he would need to properly interact with his teammates.
- **Custom settings**: Tobias's specific requirements and preferences might be accommodated via the app, bot, or website. For instance, he may select to concentrate on particular subjects or regions of language acquisition, alter the speed at which the audio and video sessions play, or select several languages for translation.
- **Community support**: A feature that enables Tobias to connect with other students who are experiencing comparable language problems might be included in the app, bot, or website. This might be his chance to connect with like-minded others, share his experiences, and get advice.

Team Roles and Responsibilities:-

Activities	REHAN	VEDANT	ALAN	SUMIT
Research & Analysis	C	R	A	C
Target State	A	R	C	C
MOSCOW analysis	C	R	A	C
Milestones & Sprints	C	R	C	R
Requirement Analysis	A	C	R	C
Stakeholder Analysis	C	A	R	C
Market Analysis	R	C	C	A
Risk Analysis	R	C	C	A
Project Planning	A	C	C	R
Documentation	R	R	R	R
Technical Concept	R	C	R	A
Use case & User story	R	R	C	A
Budget Planning	A	R	C	R

Figure 4: Team Roles and Responsibilities

TARGET STATE

The goal of the AI-supported local language and dialect training and translation program is to give language learners who encounter challenges in day-to-day living owing to dialect variations a complete language learning solution. Accurate, user-friendly, and straightforward, the app, bot, or website will give language learners the resources they need to become fluent in their target dialect.

The people who speak Swiss German or Thuringia dialects who want to study or enhance their abilities in either dialect will be the app's/website's target audience. Students, professionals, visitors, and anybody else who needs to communicate successfully in either dialect will make up the target audience.

The app, bot, and website are intended to provide users with precise and trustworthy translations between Swiss German and the Thuringian dialect, voice recognition technology for practising pronunciation, user-customizable

settings to tailor the learning experience to their needs, and language learning modules created to help users become proficient in their target dialect.

In general, the goal of the app, bot, or website is to offer a complete language learning solution that satisfies the needs of language learners like Tobias Schmidt, empowering them to overcome challenges brought on by dialect differences in daily life and effectively communicate in their target dialect.

Target Infrastructure

The platform chosen for deployment will determine the target infrastructure for the AI-supported local language and dialect training and translation program. The infrastructure should be planned to offer the required computer power, data storage, and network connectivity to support the functionality and performance of the app, bot, or website.

The target infrastructure, for instance, may comprise cloud-based computing resources like virtual machines, containers, and serverless services if the app, bot, or website is launched as a web application. A cloud-based database for data storage and network connectivity to guarantee quick and dependable access to the app, bot, or website might also be included in the architecture.

If the program is made available as a mobile application, the target infrastructure may include network connectivity to enable quick and dependable access to the program, mobile device storage and processing capacity, and cloud-based computing resources for the app's back-end functionality.

Details:-

Computing Resources:

The target infrastructure should provide sufficient computing resources to support the program's functionality and performance. This may include:

- Cloud-based virtual machines, containers, or serverless functions for processing user requests and executing language learning modules.
- Sufficient CPU and memory resources to handle concurrent user requests and ensure fast response times.
- GPUs for accelerating machine learning algorithms for voice recognition and translation.
- Load balancers for distributing user requests across multiple instances of the program to ensure high availability and scalability.

Data Storage:

The target infrastructure should include a scalable and secure data storage solution to store user data and program data. This may include:

- Cloud-based databases such as SQL or NoSQL databases for storing user profile information, language learning progress, and other program data.
- Object storage solutions such as Amazon S3 or Google Cloud Storage for storing large media files such as audio recordings and video tutorials.

Network Connectivity:

The target infrastructure should provide fast and reliable network connectivity to ensure the program can be accessed by users from anywhere in the world. This may include:

- Content delivery networks (CDNs) for delivering program content to users quickly and efficiently.
- Load balancers for distributing user requests across multiple instances of the program to ensure high availability and scalability.
- Secure network protocols such as HTTPS and SSL/TLS to ensure secure communication between users and the program.

Security:

The target infrastructure should be designed with security in mind to ensure user data is protected from unauthorized access and cyberattacks. This may include:

- Network security measures such as firewalls, intrusion detection systems, and web application firewalls to protect against cyberattacks.
- Encryption of user data in transit and at rest to ensure data confidentiality.
- Multi-factor authentication for user logins to prevent unauthorized access to user accounts.
- Regular security audits and vulnerability assessments to ensure the program remains secure and up-to-date with the latest security best practices.

Ongoing Maintenance and Support:

The target infrastructure should be designed to enable ongoing maintenance and support to ensure the program remains up-to-date and continues to meet the target audience's language learning needs. This may include:

- Regular updates and patches to fix bugs and improve performance.
- User support channels such as chatbots or email support to address user questions and concerns.
- Monitoring and alerting systems to detect and resolve issues before they impact users.
- Regular backups of user data and program data to ensure data recovery in the event of a disaster.

MOSCOW Analysis :

Here are some additional details for each of the MOSCOW requirements:

Must-Have Requirements:

- Voice recognition and translation: The program must be able to accurately recognize spoken Swiss German and Thuringia dialects and provide real-time translations in the user's desired language.
- Multilingual support: The program could support multiple languages and allow users to switch between languages seamlessly.
- Personalized language learning modules: The program should offer language learning modules that are tailored to the user's proficiency level and preferred learning style. These modules could include vocabulary, grammar, listening, speaking, and reading exercises.

- Security: The program must follow industry-standard security practices to ensure that user data is protected from unauthorized access or cyberattacks. This includes using encryption, multi-factor authentication, and secure data storage practices.
- Accessibility: The program should be available on multiple platforms to ensure that users can access it from their preferred device. This includes mobile devices, desktop computers, and web browsers.
- User experience: The program should be designed with a user-friendly interface and intuitive navigation to ensure a seamless user experience.
- Speech recognition: The program could use speech recognition technology to help users improve their pronunciation and provide feedback on their speaking skills.
- Translation tool: The program could include a translation tool that allows users to translate words, phrases, or sentences between different languages.

Should-Have Requirements:

- Gamification and rewards: The program could incorporate gamification elements such as badges, leaderboards, and rewards to motivate users to continue language learning.
- Social component: The program should allow users to connect with other language learners and practice their language skills. This could include chat rooms, discussion forums, and language exchange programs.
- Additional resources: The program should provide users with additional resources such as cultural insights and grammar explanations to enhance their language learning experience.
- Real-time translation: The program could support real-time translation for written text and chat conversations, in addition to voice translation.

Could-Have Requirements:

- Integration with third-party tools and resources: The program could integrate with other language learning tools and resources to provide users with a comprehensive language learning experience.
- Machine learning algorithms: The program could use machine learning algorithms to personalize the language learning experience further and adapt to the user's progress over time.
- Virtual reality and augmented reality: The program could incorporate virtual reality or augmented reality technology to provide users with immersive language learning experiences.
- Subscription-based model: The program could offer a subscription-based model to access premium language learning features and resources.

Nice To Have Requirements:

- Personalized learning path: The program could create a personalized learning path for each user based on their language proficiency level and learning objectives. This could include recommended lessons, exercises, and activities tailored to the user's needs.
- Interactive language exercises: The program could include interactive language exercises such as fill-in-the-blank, multiple choice, and sentence completion activities to reinforce vocabulary and grammar.
- Cultural insights: The program could provide users with insights into the culture and customs of the language they are learning, such as holidays, traditions, and social norms.
- Progress tracking: The program could provide users with a dashboard to track their progress in learning the language, including their proficiency level, time spent practicing, and areas of improvement.
- Social learning: The program could facilitate social learning by connecting users with other language learners, native speakers, or language tutors for conversation practice and language exchange.
- Offline mode: The program could include an offline mode that allows users to access content and practice exercises without an internet connection.
- Adaptive learning: The program could use adaptive learning technology to adjust the difficulty level of lessons and activities based on the user's performance and progress.

Table 1:Sort by Complexity

S. NO.	COMPATIBILITY	PRIORITIZATION	COMPLEXITY LEVEL	USER STORY/EPIC
2	User Experience	MUST	-	EPIC
3	Multilingual Support	MUST	-	EPIC
4	Security	MUST	-	EPIC
5	Accessibility	MUST	-	EPIC
25	PAID SUBSCRIPTION MODEL	MUST	-	EPIC
22	Offline Translate Functionality	Nice-to-Have	21	US
1	Speech Recognition	MUST	13	US
18	AI Based Chat Assistant	Nice-to-Have	13	US
6	Real-time Translation	MUST	8	US
8	Document Translation	MUST	8	US
12	Social Media App for other users to connect	CAN	8	US
17	Dialect Dictionary	Nice-to-Have	8	US
21	API for Third Party Websites and Apps	COULD	8	US
7	Image to text with Translation	MUST	5	US
9	Dialect Learning	COULD	5	US
15	Chat with other users	CAN	5	US
16	Voice Chat with native users	CAN	5	US
19	Everyday lessons for Dialect Learners	Nice-to-Have	5	US
23	Website Translation	MUST	5	US

10	Favourite Translation	MUST	3	US
11	History	MUST	3	US
13	Share Translation	SHOULD	3	US
14	Search Previous Translation	SHOULD	3	US
20	Quizzes for learners	CAN	3	US
24	Log in/Log out Functionality	MUST	3	US

Table 2: Sort by Prioritization

S.NO	COMPATIBILITY	PRIORITIZATION	COMPLEXITY LEVEL	EPIC/US
13	Share Translation	SHOULD	3	US
14	Search Previous Translation	SHOULD	3	US
17	Dialect Dictionary	Nice-to-Have	8	US
18	AI Based Chat Assistant	Nice-to-Have	13	US
19	Everyday lessons for Dialect Learners	Nice-to-Have	5	US
22	Offline Translate Functionality	Nice-to-Have	21	US
1	Speech Recognition	MUST	13	US
2	User Experience	MUST	-	EPIC
3	Multilingual Support	MUST	-	EPIC
4	Security	MUST	-	EPIC
5	Accessibility	MUST	-	EPIC
6	Real-time Translation	MUST	8	US
7	Image to text with Translation	MUST	5	US
8	Document Translation	MUST	8	US
10	Favourite Translation	MUST	3	US
11	History	MUST	3	US
23	Website Translation	MUST	5	US
24	Log in/Log out Functionality	MUST	3	US
25	PAID SUBSCRIPTION MODEL	MUST	-	EPIC
9	Dialect Learning	COULD	5	US
21	API for Third Party Websites and Apps	COULD	8	US

12	Social Media App for other users to connect	CAN	8	US
15	Chat with other users	CAN	5	US
16	Voice Chat with native users	CAN	5	US
20	Quizzes for learners	CAN	3	US

MILESTONES

Research and Analysis:

- **Describe the target market's demands in terms of language learning**: To determine the language learning needs of the target audience, use surveys and focus groups.
- **Analyse the Thuringia dialect and the Swiss German dialect in depth**: Engage linguists to examine the dialects and determine how they differ linguistically.
- **Describe the features and capabilities needed to build the app, bot, or website**: To determine the essential characteristics, use competitive analysis and market research.
- **Choose the right AI technologies**: Hire AI specialists to determine the best AI solutions.

Prototyping:

- **Create mockups and wireframes for the app, bot, and website**: Mockups and wireframes may be created using design tools.
- **Create a user interface and a positive user experience**: Employ UI/UX specialists to create a simple and user-friendly interface.
- **Make working prototypes**: Software development tools may be used to build working prototypes.

App Integration & Development:

- **Create the app, bot, or website's front and back ends**: Create the app, bot, or website using programming languages like HTML, CSS, JavaScript, and Python.
- **Embrace AI technology integration**: Engage AI specialists to combine chatbots, voice recognition, natural language processing, and machine learning.

- **Create the speech recognition software, translation tools, language learning modules, and adjustable settings**: Create speech recognition technologies, translation tools, language learning modules, and adjustable settings using software development tools.

Deployment & Testing:

- Install the app, bot, or website on the proper platform: To launch the app, bot, or website on the web, Android, or iOS platform, hire software engineers.
- Conduct testing after deployment: Make sure that the app, bot, or website functions properly in the target environment by using testing tools.
- Maintain and support ongoing maintenance: Employ a specialised staff to provide continuous upkeep and support to make sure the app, bot, or website is current and continues to function properly.
- The development team can guarantee the project's success by producing a high-quality and user-friendly app/bot/website by paying attention to the project phase management aspects mentioned above.

Monitorization:

In this phase we are going to monitor our live product on app and website. If any ongoing issue is found we will repair/solve that issue via various teams (depending on what type of issue).

It seems that the monitorization service from our side will be on time and effective for product stability.

Revenue Generation:

Offer a free version of the app supported by advertisements. Advertisers can display targeted ads within the app, and you can earn revenue based on impressions, clicks, or conversions. Ensuring that the ads do not disrupt the user experience and are relevant to the app's target audience. It's important to consider the target market, user needs, and competition when selecting a revenue model.

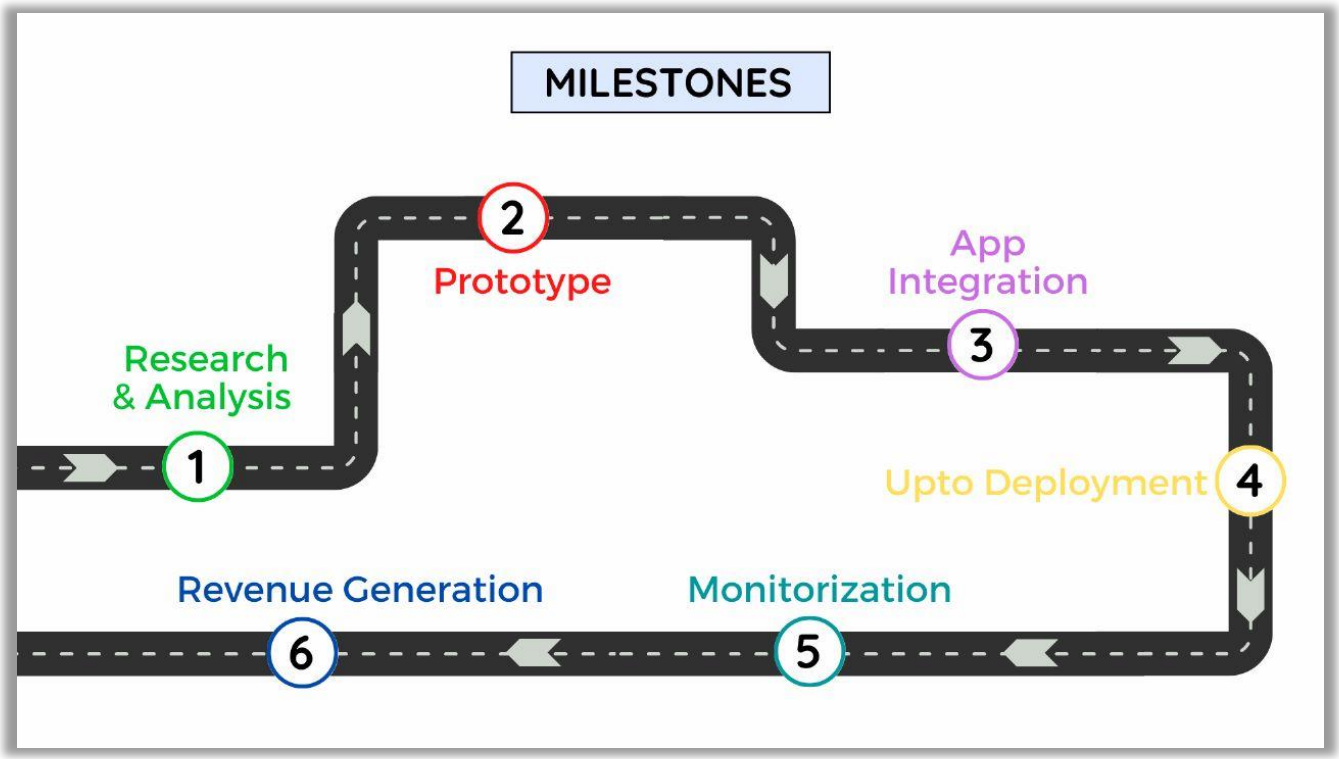

Figure 5: Milestones of our project

Eisenhower Matrix:

Table 3: Eisenhower Matrix Table

	Urgent	Not Urgent
Important	❖ Develop AI language recognition and translation technology ❖ Implement security features to protect user data and privacy ❖ Conduct user testing to ensure functionality and usability of the application ❖ Address critical bugs and errors in the application	❖ Develop new features to enhance user experience ❖ Create high-quality content and resources to support language training and translation ❖ Conduct market research to identify new target audiences and potential partnerships ❖ Establish customer support system for users to provide feedback and receive assistance

Not Important	❖ Attend meetings with stakeholders and team members ❖ Respond to non-critical emails and messages ❖ Conduct minor administrative tasks	❖ Engage in non-work-related activities ❖ Perform non-critical social media activities ❖ Conduct personal tasks unrelated to the project

ANALYSIS

REQUIREMENT ANALYSIS

The project's main objective is to develop an AI-powered dialect-to-dialect translation application that can successfully:

- ✓ Identify and analyze the source dialect.
- ✓ Translate the source dialect to the target dialect accurately.
- ✓ Offer a user-friendly interface for users to interact
- ✓ Adapt and learn from user interactions to improve the translation quality continuously.

Resource Requirements

- A powerful AI engine, possibly utilizing deep learning technologies like recurrent neural networks (RNN) or long short-term memory (LSTM) networks, to handle the complexities of the dialect translation.
- A comprehensive database of dialect examples, their translations, and linguistic information.
- A group of programmers with knowledge of creating both online and mobile applications.
- A team dedicated to quality assurance and testing of the application.

→ IT and Development Team

The IT and development team will be responsible for the development and deployment of the application using an agile methodology. The team will comprise the following roles:

- Project manager: Oversees the project's progress, resource allocation, and deadlines.
- AI specialists: Develop, train, and test the translation engine.
- Linguistic experts: Guide language, dialects, and cultural nuances.
- Mobile and web developers: Implement user interfaces and ensure cross-platform compatibility.
- Quality assurance specialists: Test, identify, and rectify issues with the application.

→ Integration Team

The integration team's role is to ensure smooth integration of the AI engine with the mobile and web applications. This team will include:

- System architects: Design the system architecture and API interfaces.

- Full-stack developers: Implement the API calls to interact with the AI engine.
- UI/UX designers: Create user-centered designs for seamless application usage.

Infrastructural Requirements

- **Mobile Applications:** The development of iOS and Android mobile applications to cater to the majority of smartphone users.
- **Smartphones:** Compatibility testing for different smartphone models to ensure that the application works smoothly across various devices.
- **Application Servers:** Robust web servers to host the AI engine and handle API requests, potentially using cloud-based services to ensure scalability and reliability.

Customer Requirements

To ensure customer satisfaction, the application should meet the following requirements:

- **User-friendly interface:** The application should be simple and easy to navigate.
- **Fast and accurate translations:** Users should receive prompt, clear translations to facilitate efficient communication.
- **Data privacy and security:** The application should be implemented as per the law of the respective country.
- **Multi-platform accessibility:** The application should be available on both mobile and web platforms, ensuring user compatibility.

Operational Requirements

For smooth operations and maintenance, the following requirements must be met:

- **Regular updates:** The application's AI engine and database should be updated regularly to improve translation quality continuously.
- **Technical support:** A dedicated team to provide timely support and assistance to users.
- **Marketing and promotion:** Focused marketing strategies to generate user interest and increase adoption.

STAKEHOLDER ANALYSIS

Table 4: Stakeholder Identification

Stakeholder	Role	Responsibility
User	**End-user of the AI-supported local language model**	**Provide feedback on user experience, report bugs or issues, and offer suggestions for improving the system.**
Competitors	**Other companies or organizations providing similar AI-supported language models**	**Monitor and stay up-to-date on developments in the field, and ensure that the product is competitively priced and offers unique value to users.**
Project Team	**Develop and train the AI-supported language model**	**Implement regular updates and improvements to the language model**
Linguistics Experts	**Professionals with expertise in the language(s) used in the AI-supported local language model**	**Provide input on language-specific nuances and ensure accuracy in translation and language processing.**
Project Manager	**Oversees the development of the AI model and manages the project team**	**Ensures that the project is on track, sets priorities and deadlines, and communicates with stakeholders about the project's progress.**
Local Communities	**Residents of the area where the AI-supported local language model will be deployed**	**Provide feedback on language usage and local customs and preferences, and help to promote the use of the product within the community.**
Investors	**Individuals or organizations investing in the development of the AI-supported local language model**	**Provide financial resources to support the project, and expect a return on investment.**

Project Owner	**The organization or individual responsible for the development and deployment of the AI-supported local language model**	**Sets the project vision and goals, ensures that the project remains aligned with broader organizational objectives, and takes responsibility for the success or failure of the project.**
Marketing Team	**Responsible for promoting the AI-supported local language model to potential users**	**Develop marketing strategies and materials, organize events and other promotional activities, and collaborate with other stakeholders to ensure that the product is well-promoted and widely adopted.**
AI Developers	**Professionals with expertise in the development of artificial intelligence systems**	**Responsible for developing the AI model, using programming languages and algorithms to process language accurately and efficiently.**
Language Technology Developers	**Professionals with expertise in the development of language technology, such as speech recognition and natural language processing**	**Responsible for ensuring that the AI-supported local language model uses the latest and most effective language technology to provide accurate and reliable translations.**

Figure 6: Stakeholder Map

MARKET ANALYSIS

Goal and Agenda

- Our goal is to solve dialect barriers and help people in sophisticated professions such as nurses, doctors, lawyers, customer facing jobs and other legal professions as well.
- Specifically we want our app to translate different German-speaking dialects such as Bavarian, Hochdeutsch and other dialogues as well.
- We would also like to offer API-based services to other technological-based businesses and further monitise it to create a new revenue stream.
- Implementing artificial intelligence services such as sentiment analysis, chat bot capability, question answering, basically natural language processing on different German dialects.
- Also we would like to provide speech recognition technology in our app which can help users in real-time situation with real-time translation it can further be also provided as another application programming interface to sectors such as banking, telecommunication, marketing, tourism, booking and managing orders, appointment for doctors, lawyers and for emergency services as well.

Target Audience

Our target audience include anyone who is above the age of 13, it can be used in following situations:

- It can help **doctors and nurses** to improve patient communication and improve the quality and accuracy of translation, it is very important to communicate accurately in critical conditions or medical emergencies or even while discussing different treatment options as it can help them to enhance the patient care by providing personalized solution, it can help the doctors to understand patients medical history, symptoms, concerns which can help in their diagnosis and treatment decision ,it can help in reducing miscommunication due to direct barriers that might lead to medical errors which can have serious consequences for patients ,further it will help doctors with increased accessibility to speak with patients who speak different dialects and it would also help in saving time and resources in busy environment such as health care where time and resources are very limited.

- It can help **customer service representatives** by improving customer satisfaction by providing better services to customers who speak different dialects by providing customers with a way to communicate in the on direct customer service representatives can build trust and report with customers using customers direct will help in increasing the customer base by removing language barriers and hence businesses can tap into new markets and reach customers further more it will also help to reduce the cost of businesses and hence reduce the amount of workforce acquired to provide multilingual customer services by using our platform they can automate translation and provide multi Kal dialect-based translation and increase sufficiency by eliminating manual translation and handle vast customer interaction in lesser time.

- It can also help people in **day-to-day activities** such as travelling by helping communicating with locals, reading menus, understanding signs and navigating the way around, it can also be used to learn a new dialect quickly and get immediate feedback on your pronunciation, furthermore it can help in doing business that can help people from other other regions who speak different directs to communicate with their partners or clients in their native dialect which can help build better relationship and increase trust it can also be used as a source of entertainment to enjoy foreign language movies ,TV shows, books and music by providing real-time translation. Overall, dialect translation apps can help break down dialect barriers and make communication easier and more accessible in a variety of day today activities.

- It can help businesses in following ways:

  - **Communication**

    It can help businesses communicate more effectively with customers and partners who speak different dialects. This can help build better relationships and increase customer satisfaction.

  - **Marketing**

    It can also help businesses create more effective marketing campaigns by translating their messages into different dialects. This can help businesses reach a wider audience and increase sales.

  - **Localization**

It can help businesses localize their products and services for different markets. For example, a business can use a dialect translation app to translate their website into a local dialect, which can help increase engagement and sales in that market.

- **Employee Training**

  It can be helpful in employee training programs. By using a dialect translation app, businesses can provide training materials in different dialects, which can help improve employee engagement and understanding.

Advertisement:

There are several forms of internet advertising, including:

- **Display Ads**

  These are visual ads that appear on websites, typically in the form of banner or image ads.

- **Search Ads**

  These are ads that appear in search engine results pages (SERPs) when users search for specific keywords or phrases.

- **Video Ads**

  These are ads that appear before, during, or after online video content, such as on YouTube or other video streaming platforms.

- **Native Ads**

  These are ads that blend in with the content on a website or social media platform, often appearing as sponsored content.

- **Social Media Ads**

  These are ads that appear on social media platforms such as Facebook, Twitter, and Instagram.

- **Email Marketing**

  These are ads that appear in email newsletters or other types of marketing emails.

- **Influencer Marketing**

  This is a form of advertising that involves working with social media influencers to promote a product or service.

- **Content Marketing**

Create blog posts, videos, or other content that showcases your app's features and benefits. You can then promote this content through social media, email marketing, or other channels.

- **App Install Campaigns**

  These are visual ads that appear on websites, typically in the form of banner or image ads.

Market Prediction and Future Trends

The future of translation-specific apps is promising, as advances in artificial intelligence and natural language processing are improving the accuracy and functionality of these apps. Here are some potential developments we can expect to see.

- **Improved accuracy**

  Machine learning and neural network algorithms are making language translation apps more accurate, with fewer errors and better context understanding. We can expect to see further improvements in accuracy as these technologies continue to evolve.

- **Real-time translation**

  Real-time translation apps that allow two or more people to communicate in different languages in real-time are becoming more common. These apps use speech recognition and translation algorithms to provide instant translations of spoken language, making communication across language barriers easier than ever before.

- **Increased dialect support**

  As dialect translation apps become more advanced, we can expect to see support for a wider range of dialects. This will enable more people to communicate and access information across dialect barriers.

- **Integration with other technologies**

  Dialect translation apps are likely to become more integrated with other technologies, such as virtual assistants and smart home devices. This will allow people to communicate and control their devices using voice commands in different dialects.

- **Personalization**

  Dialect translation apps may become more personalised , allowing users to customise their language preferences and receive translations tailored to their needs and interests.

- **Integration with augmented reality**

  Augmented reality technology may be used to provide real-time translations of written dialects in the user's environment, such as street signs or menus.

Overall, the future of language translation apps is exciting, and we can expect to see continued improvements and innovations as technology continues to advance.

Competitor Analysis

Yes, there are some apps that specialize in translating different dialects of a language. Here are a few examples:

Dialectic Translator

Dialect Translator is an app that translates different dialects of Chinese, including Mandarin, Cantonese, and Shanghainese. It uses voice recognition and translation technology to provide accurate translations of spoken language.

Dialectic

Dialectic is an app that translates different dialects of English, including American, British, Australian, and New Zealand English. It provides translations of words, phrases, and idioms, and includes audio pronunciation and usage examples.

Dialecto

Dialecto is an app that translates different dialects of Spanish, including Mexican, Argentine, and Colombian Spanish. It provides translations of words and phrases, and includes audio pronunciation and usage examples.

Dialect Coach

Dialect Coach is an app that provides guidance and training on different dialects of English. It includes audio samples and exercises to help users improve their pronunciation and understanding of different dialects.

These are just a few examples of apps that specialize in dialect translation. There may be other apps available for different languages and dialects. It's important to note that dialects can vary widely within a language, so it's important to choose an app that specifically supports the dialect you need.

There are some apps that can convert German dialects. Here are a few examples:

PONS Dialect Translator

PONS Dialect Translator is an app that can translate German dialects, including Bavarian, Swabian, and Low German. It provides translations of words and phrases, and includes audio pronunciation and usage examples.

Sächsisch Deutsch Übersetzer

Sächsisch Deutsch Übersetzer is an app that can translate between Standard German and the Saxon dialect, which is spoken in the state of Saxony in Germany. It provides translations of words and phrases, and includes audio pronunciation and usage examples.

Bairisch Deutsch Übersetzer

Bairisch Deutsch Übersetzer is an app that can translate between Standard German and Bavarian, a dialect spoken in Bavaria and Austria. It provides translations of words and phrases, and includes audio pronunciation and usage examples.

Schwäbisch Deutsch Übersetzer

Schwäbisch Deutsch Übersetzer is an app that can translate between Standard German and Swabian, a dialect spoken in the region of Swabia in Germany. It provides translations of words and phrases, and includes audio pronunciation and usage examples.

These are just a few examples of apps that can convert German dialects. It's important to note that dialects can vary widely within a language, so it's important to choose an app that specifically supports the dialect you need.

Legal Aspect

Language translation apps can raise several legal issues, particularly when used in a professional or commercial context. Here are a few key legal aspects of language translation apps to consider:

- **Accuracy**

  Depending on the context and purpose of the translation, there may be legal requirements for the translation to be accurate. For example, in legal documents or contracts, inaccurate translations could lead to disputes or legal liabilities. Therefore, it's important to use translation apps that are reliable and accurate.

- **Confidentiality**

  If the translations involve sensitive or confidential information, such as personal data, trade secrets, or legal documents, there may be legal requirements to ensure the confidentiality of the translation. Translation service providers should have appropriate safeguards and agreements in place to protect the confidentiality of the translation.

- **Intellectual property**

  Translations can involve copyrighted or trademarked materials, such as literary works, software, or brand names. Depending on the jurisdiction and the purpose of the translation, there may be legal requirements to obtain permission or licenses to use or translate these materials.

- **Regulatory compliance**

  In some industries or contexts, there may be regulatory requirements for translations to comply with certain standards or guidelines. For example, in the healthcare industry, translations of medical documents or labels may need to comply with regulatory requirements for accuracy and clarity.

- **Liability**

Depending on the context and purpose of the translation, there may be legal liabilities for inaccurate or misleading translations. For example, in the context of marketing or advertising, inaccurate translations could lead to false or misleading claims, which could result in legal liabilities or regulatory fines.

These are just a few examples of the legal aspects of language translation apps. It's important to understand the legal requirements and potential risks associated with using translation apps, particularly in professional or commercial contexts.

RISK ANALYSIS

PROJECT RISK

- **Accuracy and Quality**

  → One of the most significant risks with dialect translation is inaccurate or poor-quality translations. This can occur due to limitations in the app's algorithms or programming, lack of context, or errors in the source text.

- **Compatibility**

  → Another risk is the compatibility of the dialect translation with various devices, operating systems, or software versions. This can affect the app's functionality, usability, or accessibility, and may require additional testing and troubleshooting.

- **Security and Privacy**

  → Dialect translation may also pose security and privacy risks, particularly if they access sensitive or confidential information, such as personal data or business communications. We must ensure that the app's data protection policies and procedures are in line with regulatory requirements and best practices.

- **Linguistic and Cultural Nuances**

  → Dialect translation may not always accurately capture linguistic and cultural nuances, such as idioms, slang. This can lead to misunderstandings or miscommunications, which can negatively impact the user experience or business outcomes.

- **Maintenance and Support**

  1. Dialect translation requires ongoing maintenance and support to ensure they remain functional, up-to-date, and user-friendly. The lack of adequate maintenance and support can lead to technical issues, user dissatisfaction, or even app failure.

It is crucial to identify these risks early in the project planning phase and develop appropriate risk management strategies to minimise their impact or avoid them altogether. This may involve measures such as extensive testing, quality assurance processes, and regular updates and upgrades to the app's software and algorithms.

TECHNICAL RISK

- **Integration with other systems**

  Translation apps may need to integrate with other systems or platforms, such as content management systems or e-commerce platforms. Incompatibilities or technical issues may arise during the integration process, leading to delays or functional limitations.

- **Scalability**

  As usage of the translation app grows, the technical infrastructure supporting it must be scalable to handle increased traffic and data processing requirements. Failure to properly plan for scalability can lead to performance issues, downtime, or data loss.

- **Natural Language Processing (NLP)**

  NLP is a core technology used by language translation apps to understand and translate human language. However, NLP can be complex and prone to errors, especially when dealing with complex or ambiguous phrases. This can result in inaccurate translations, which can negatively impact user experience.

- **Data Security**

  Language Dialect translation apps may process sensitive data, such as personally identifiable information or confidential business information. Adequate measures must be taken to ensure the security and privacy of this data, including encryption and access controls.

- **User Experience**

  The success of a dialect translation app is highly dependent on the user experience it provides. Technical issues, such as slow response times or clunky user interfaces, can negatively impact user experience and lead to user dissatisfaction.

BUSINESS RISK

- **Accuracy and Quality**

  The accuracy and quality of dialect translation apps are crucial to their success. If the translations are inaccurate or of poor quality, it can lead to misunderstandings, confusion, and even legal liability. This can damage the reputation of the app and the company that produces it.

- **Privacy and Security**

  Dialect translation often requires users to provide access to their personal information and data, such as contacts, messages, and emails. This can be a significant privacy and security risk, especially if the app is not secure or if the data is not encrypted.

- **Cultural Sensitivity**

Dialect translation must be culturally sensitive to avoid offending or alienating users. This can be particularly challenging in languages where there are different dialects and cultural nuances that must be taken into account.

- **Intellectual Property Rights**

  The use of dialect translation can raise intellectual property issues, particularly if copyrighted material is translated without permission or attribution. This can result in legal action against the app and the company behind it.

- **Reliance on Technology**

  Dialect translation relies heavily on technology, which can be unpredictable and subject to errors or malfunctions. If the technology fails to work correctly, it can cause significant business disruptions and lost revenue.

- **Competitors**

  There are some dialect translation applications available in the market, which can create competition for market share. We must constantly innovate and improve our accuracy to stay ahead of our competitors.

RISK MANAGEMENT

Table 5: Risk Analysis

Risk	Category	Likelihood	Impact	Priority
Medical document is translated incorrectly	System	Likely	High	High
Taking care of cultural sensitivity	System	Likely	High	High
Data Breach	Security	Likely	High	High
Wrong Image-to-text conversion	System	Likely	High	High
Sloppy grammar and informal speech	Linguistic	Likely	High	High
Unwritten/Unknown Dialects	Linguistic	Likely	Medium	Medium
Wrong Pronunciation, Morphological Complexity	Pronunciation	Very Likely	Medium	Medium
Compound words	Linguistic	Likely	Medium	High
Untranslatable Words	Linguistic	Very Likely	High	Medium

Risk	Category	Likelihood	Impact	Priority
Speech Synthesis for older users	Linguistic	**Very Likely**	**High**	**High**
Issue of translating long and complex sentences	Linguistic	**Likely**	**High**	**Medium**
Natural Sound of Speech Synthetic System	Linguistic	**Likely**	**High**	**High**
Delay in App Development	Technical	**Very Likely**	**High**	**Medium**
Data not stored properly	Technical	**Very Likely**	**High**	**Medium**
Political Correctness	Logical	**Likely**	**High**	**High**
Religious Views	Logical	**Likely**	**High**	**High**
High Processing Time	System	**Highly Likely**	**High**	**High**
High Market & User Acquisition Costs	Budget	**Likely**	**High**	**Medium**
Aesthetic Word And Phrase Translation	Logical	**Highly Likely**	**High**	**High**

RISK MANAGEMENT STRATEGIES

#RMS001

Risk	**Challenges in translating scientific, technical, medical terminologies**
Strategy	**Mitigation**

Risks such as this need to be catered with a lot of attention as it can cause serious consequences such as inaccurate diagnosis and treatment. To mitigate such risk, we can work with subject matter experts who have a deep understanding of medical terminology and can help ensure accurate translations and establish a review process that includes multiple levels of review and verification, including review by a medical expert. This can help to ensure that translations are accurate and free from errors.

#RMS002

Risk	**Taking care of cultural sensitivity.**
Strategy	**Prevention**

Cultural sensitivity is an important consideration when developing any product or service that will be used by people from different cultures. Failure to consider cultural sensitivity can lead to misunderstandings, offence, and even loss of business. We can tackle this risk by conducting research to understand the cultural norms and expectations of our target audience. This can help us avoid potential misunderstandings. Moreover, we can consult with cultural experts or members of the target audience to ensure that our translation is culturally appropriate. This can help us avoid unintentional cultural in-sensitivities.

#RMS003

Risk	Data breach
Strategy	Mitigation

Data breaches can have significant consequences, including damage to reputation, and loss of customer trust. We can mitigate this risk by implementing strong security measures, such as encryption, firewalls, and access controls, to protect sensitive data and develop and test a plan for responding to data breaches, including how to notify customers and regulators, and how to recover from the breach.

#RMS004

Risk	Wrong Image-to-Text Conversion
Strategy	Mitigation

To mitigate the risks associated with wrong image to text conversion, we can train our OCR system with a large dataset of accurate images and text. This can help improve the accuracy of the OCR conversion and reduce the risks of errors. Also, we should continuously monitor the OCR conversion process and make improvements as needed. Regularly reviewing and analysing the OCR conversion results can help us identify and address any issues before they become larger problems.

#RMS005

Risk	Sloppy grammar and informal speech
Strategy	Mitigation

To mitigate the risks associated with Sloppy grammar and informal speech we should use a style guide in our language model to ensure consistent and professional language across our application regarding rules for grammar, spelling and punctuation. Moreover, we can implement spell checkers to identify errors and suggest automatic corrections or typos. This is a long term process for which we need to constantly monitor feedback and make necessary changes.

#RMS006

Risk	Unwritten/Unknown Dialects
Strategy	Delegation

There are a lot of dialects which are unknown, in order to tackle this, we have to monitor feedback from users and identify issues with unknown dialects. Working with linguistic experts can help us develop a better understanding of how to handle unknown dialects and improve accuracy of the overall system

#RMS007

Risk	Wrong Pronunciation, Morphological Complexity
Strategy	Mitigate

Wrong Pronunciation in speech translation can lead to confusion and misunderstanding, The only best possible way to deal with such a situation is to train our system with large dataset of accurate pronunciation samples as this can improve accuracy of the system and implement quality control checks and provide an option for user to correct pronunciation and thereby improve accuracy.

#RMS008

Risk	Compound Words
Strategy	Mitigate

Compound words can be a challenge for language translation because the meaning of the compound word can be different from the sum of its individual parts. To mitigate such risks, we can use a comprehensive dictionary that includes compound words and their meanings to ensure accuracy in translation and conduct user testing to identify any areas where the translation of compound words may be unclear or inaccurate, and make adjustments accordingly. It's important to prioritise accuracy in translation, and to continually test and refine the translation process.

#RMS009

Risk	Untranslatable Words
Strategy	Accept

Untranslatable words can pose a big challenge in translation, We can provide footnotes or glossaries to help the reader better understand meaning and significance of word and it's context oe we can try translating using a similar word of phrase that conveys a similar meaning, or creating a new word or phrase that accurately captures the essence of original word.

#RMS010

Risk	Speech Synthesis for older users
Strategy	Mitigation

When it comes to speech synthesis for older people, there are several risks to consider, such as hearing impairments, difficulty understanding synthesized speech, and confusion or frustration due to unfamiliar technology. We can Mitigate this risk by using clear and simple language that is easy to understand for older adults to follow and provide adjustable volume control so that older adults can adjust the volume to a comfortable level based on their hearing needs.

#RMS011

Risk	Issue of translating long and complex sentences
Strategy	Mitigation

Long and complex sentences can pose a challenge for language translation as they can be difficult to translate accurately and can result in a loss of meaning. To Mitigate such problems which might arise in future, we can limit the amount of text that will be translated later into speech and break down long and complex sentences into shorter and simpler sentences that are easier to synthesize and translate into speech.

#RMS012

Risk	Natural Sound of Speech Synthetic System
Strategy	Mitigation

Natural sounding speech is essential for any speech synthesis system, especially in language translation where the accuracy and naturalness of the translated speech can affect the comprehension and effectiveness of communication. To mitigate this we can train the speech synthesis system with a diverse dataset that includes a range of accents and speech styles and with that we have to incorporate intonation and stress into the synthetic speech to help convey meaning and context. It's important to prioritise the naturalness and accuracy of the synthetic speech, and to continually test and refine the system to ensure it meets the needs of users.

#RMS013

Risk	Data Collection & Training
Strategy	Mitigation

Developing a dialect translation app requires a significant amount of data collection and training to ensure that the app accurately recognizes and translates the unique features of the dialect. This can be a time-consuming and resource-intensive process. We can mitigate this risk by conducting thorough research on the target language and dialect to identify the unique features that need to be captured. This can help to ensure that the collected data is relevant and useful. We can use POS tagging models can be used to assign a grammatical tag to each word in a sentence, which can help to identify patterns and relationships in the text data.

#RMS014

Risk	Delay in App Development
Strategy	Mitigation

Delay in App development can be mitigated by prioritising tasks based on their level of importance and allocating resources accordingly can help to ensure that the most critical tasks are completed on time. It is also important to have a contingency plan(backup) in place in case of resource constraints or unexpected delays and maintaining effective communication among team members, stakeholders, and customers will help to ensure that everyone is aware of project status, potential issues, and changes to the project timeline. This can be possible by breaking down the project into smaller, more manageable tasks that can help to ensure that progress is being made and that milestones are being met. This can also make it easier to identify potential delays and take corrective action before they impact the overall timeline.

#RMS015

Risk	Data not stored properly and securely
Strategy	Mitigation

To mitigate the risks associated with data not being stored properly, we can use secure data storage systems to ensure that the data is protected from unauthorised access or cyber attacks. This can include measures such as encryption, multi-factor authentication, regular backups and we can implement access control mechanisms such as role-based access control can help to ensure that only authorized users have access to sensitive data. We can use anonymization techniques such as tokenization (breaking down a piece of text into smaller units) or pseudonymization can be used to protect the privacy of individuals whose data is being stored.

Pseudonymization is the process of replacing personally identifiable information (PII) with artificial identifiers, or pseudonyms, that do not reveal the individual's identity. Pseudonymization is often used in healthcare and other industries where sensitive personal data is collected, such as financial services and government agencies. It is also a requirement of the EU General Data Protection Regulation (GDPR), which mandates that organisations pseudonymised personal data to protect privacy.

#RMS016

Risk	Political Correctness
Strategy	Mitigation

Mitigating the risk of political correctness in dialect translation can be challenging, as it requires a nuanced understanding of cultural and social contexts. We can have a diverse team of translators, reviewers, and subject matter experts. This can help ensure that different perspectives and cultural sensitivities are taken into account during the translation process. It's important to stay up to date with cultural and political issues in the countries or regions where the app will be used. This can help avoid inadvertently using language or terminology that is deemed insensitive or inappropriate.

#RMS017

Risk	Religious Views
Strategy	Mitigation

Mitigating the risk of religious views in translation requires a combination of cultural sensitivity, religious expertise, and technical solutions such as implementing a review process where translations are reviewed by multiple people to ensure that they are accurate, culturally appropriate, and religiously sensitive. The review process should include a range of perspectives, including religious leaders, subject matter experts, and other stakeholders. Machine learning algorithms can be trained to identify and flag potentially sensitive or offensive language in translations related to religious views, allowing for manual review and adjustment.

#RMS018

Risk	High Processing Time
Strategy	Mitigation

Mitigating the risk of long processing time in language translation apps can improve user experience and increase overall efficiency. We can employ pre-processing techniques which can improve performance by reducing the amount of data that needs to be processed. Techniques such as data compression and filtering can significantly reduce processing time. Huffman coding technique can be used to assign variable-length codes to each symbol in the data based on its frequency of occurrence. Huffman coding is commonly used for compressing text files.

#RMS019

Risk	High Market & User Acquisition Cost
Strategy	Mitigation

Creating a translation app involves various costs, which can pose budget risks if not managed properly. The cost of marketing the app and acquiring users can be a significant budget risk, especially if the app is new. We can determine the target audience and develop a marketing strategy that focuses on reaching and engaging them. This could include social media advertising, content marketing, and influencer partnerships. We should continuously monitor the performance of marketing campaigns and optimize them to maximize ROI.

#RMS020

Risk	Aesthetic Word And Phrase Translation
Strategy	Mitigation

To accurately translate aesthetic or literary words or phrases, then it's important to have a team of skilled and knowledgeable translators who can properly interpret the intended meaning and cultural context. We need to conduct thorough reviews of the translations to ensure accuracy and consistency. This can include having multiple translators review the same text and compare their translations to ensure consistency. It can be helpful to test translations with a sample audience to ensure that the translations are well received and accurately capture the intended meaning. This can help identify any issues or areas for improvement before the translations are released to a wider audience.

TECHNICAL CONCEPT

TECHNICAL METHODOLOGY

Hidden Markov Model

A Hidden Markov Model (HMM) is a statistical model used to describe a sequence of observations, where each observation is assumed to be generated by an underlying unobservable process. The HMM is a type of generative model, which means that it models the probability distribution of the observed sequence given some underlying state sequence.

An HMM consists of two main components: a set of hidden states and a set of observable symbols. The hidden states represent the unobservable process that generates the observed symbols. The observable symbols represent the sequence of observations that we are interested in modeling.

The HMM assumes that the hidden state sequence is a Markov process, meaning that the probability of moving from one state to the next depends only on the current state and not on any previous states. The probability of emitting an observable symbol depends on the current hidden state.

The HMM is trained using the Baum-Welch algorithm, which is a variant of the Expectation-Maximization algorithm. The algorithm starts by randomly initializing the model parameters and iteratively updates them to maximize the likelihood of the observed sequence. This involves computing the forward and backward probabilities, which are used to update the transition and emission probabilities.

Once the model is trained, it can be used for a variety of tasks, such as sequence classification, sequence prediction, or sequence generation. One common application of HMMs is in speech recognition, where the observed sequence is a sequence of audio signals, and the hidden states represent phonemes or other linguistic units.

Baum-Welch Algorithm

The Baum-Welch Algorithm is a variant of the Expectation-Maximization (EM) algorithm, used to estimate the parameters of Hidden Markov Models (HMMs) from a set of observed data. The algorithm proceeds in two main steps: the E-step and the M-step.

In the E-step, the algorithm computes the expected sufficient statistics of the hidden state sequence given the observed data and the current estimate of the model parameters. Specifically, it computes the forward and backward probabilities, which represent the probability of being in a particular hidden state at a particular time, given the observed sequence up to that point. These probabilities are used to compute the expected number of transitions between states and the expected number of emissions of each symbol, for each pair of hidden states.

In the M-step, the algorithm updates the model parameters based on the expected sufficient statistics computed in the E-step. Specifically, it computes the maximum likelihood estimates of the transition

and emission probabilities, which maximize the likelihood of the observed data given the current estimate of the model parameters. This involves normalizing the expected sufficient statistics by the total probability of the observed sequence, and using them to update the transition and emission probabilities.

The algorithm iterates between the E-step and the M-step until convergence, i.e., until the change in the log-likelihood of the observed data is below a certain threshold.

The Baum-Welch algorithm is used to estimate the parameters of an HMM from a set of observed data, without knowing the true state sequence that generated the data. It is a type of unsupervised learning algorithm, since it does not require labeled data, but rather learns the structure of the underlying process that generates the data. The algorithm is widely used in speech recognition, natural language processing, bioinformatics, and other fields where sequence data is common.

FRONT END

REQUIREMENT GATHERING

→ User Interface Design

Table 6:UI Requirement

Input Method	**Source and Target Dialects**	**Translation Output**	**Navigation and Controls**
The input method for the text to be translated should be clear and easily accessible. Users should be able to input text through a variety of methods, such as typing, voice recognition, or by taking a photo of text.	The source and target dialects should be clearly visible and easy to switch between. It is important to provide a wide range of languages to choose from, especially for popular languages used worldwide.	The translated output should be displayed clearly and prominently, with options to listen to the translation or see an alternative translation.	The navigation and controls should be intuitive and easy to use. Users should be able to easily navigate between pages, select different dialects, and switch between input methods.

Visual Design	**Feedback and Error Handling**	**Personalization**	**Accessibility**
The visual design of the app should be clean, modern, and visually appealing. It should use colour and typography to create a clear and easy-to-read layout.	The app should provide clear feedback to the user during the translation process, such as progress indicators or status messages. It should also handle errors gracefully and provide clear instructions on how to resolve any issues.	The app should provide options for personalization, such as saving favourite translations or customising the app's interface.	The app should be designed with accessibility in mind, such as providing alternative text for images and ensuring that it is compatible with assistive technologies.

→ Accessibility

Table 7:Accessibility Requirement

Screen Reader Support	**Text Size**	**Language Support**	**Colour Contrast**
Our app should be designed in a way that it can be easily read by a screen reader, which is a software that reads out the text displayed on the screen for visually impaired users.	The tool should allow users to adjust the text size, as many users with visual impairments may need larger text to read.	The tool should support a wide range of languages and dialects.	Our app should use colour schemes that provide sufficient contrast between foreground and background colours to make it easier for users with visual impairments to read the content.

→ Performance Optimization

- Our app should be optimised for speed and performance to ensure that they load quickly and provide a smooth user experience. This includes optimising images and other media, minimising HTTP requests, and using caching techniques.

→ **Version Control**

- Our project should be managed using version control systems like Git to ensure that changes to the codebase are tracked and managed effectively. This enables developers to collaborate on the project and roll back changes if necessary.

→ **Testing and Debugging**

- Our app should be thoroughly tested and debugged to ensure that they work correctly and are free of errors. This includes using automated testing tools and debugging tools to identify and fix issues.

PROTOTYPING UI/UX

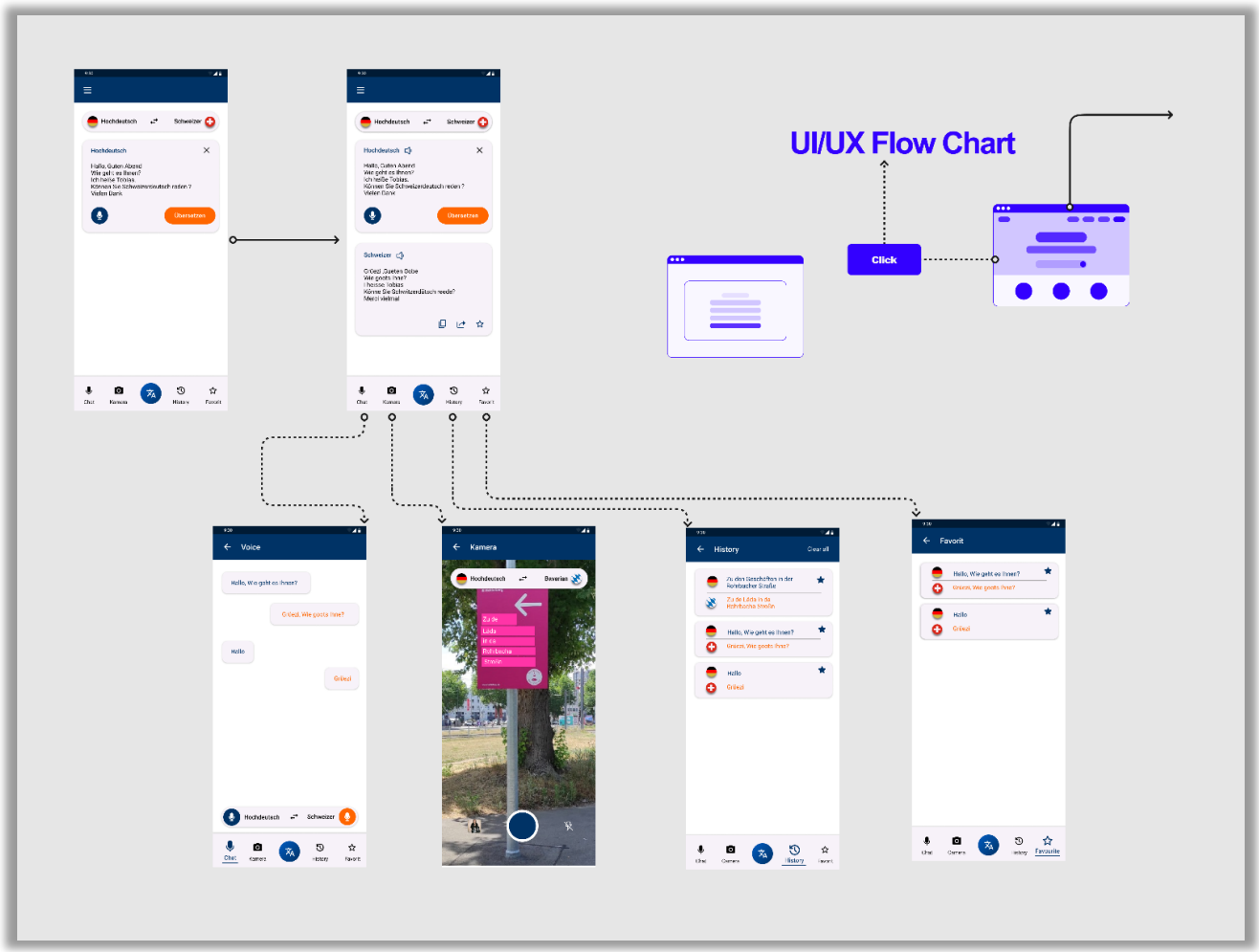

Figure 7: UI/UX Flow Chart

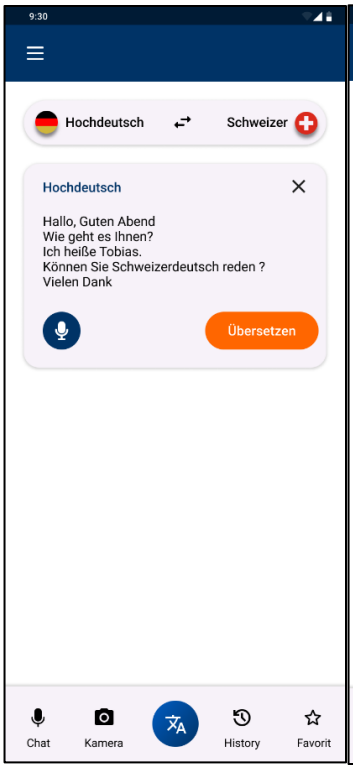

Figure 8: Screen 1

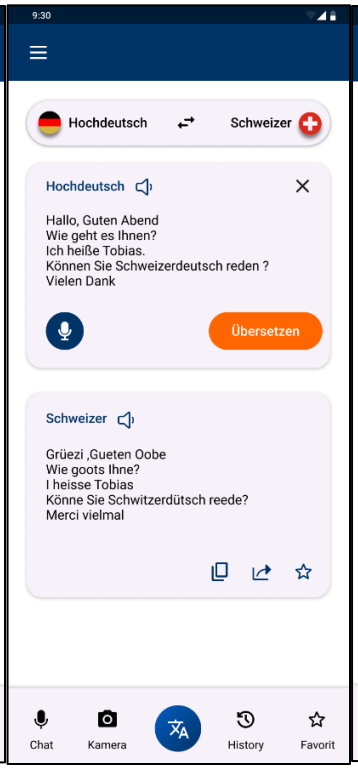

Figure 9: Screen 2

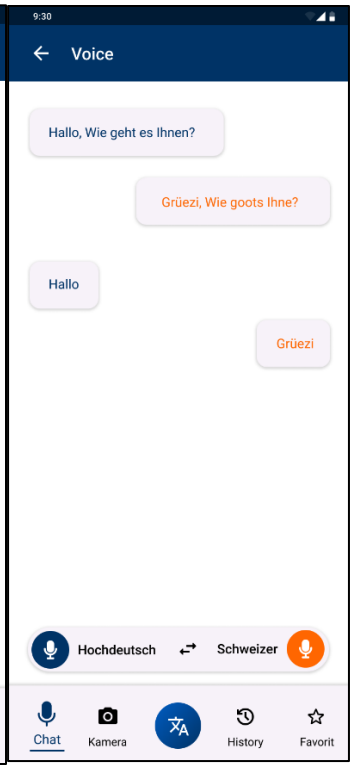

Figure 10: Screen 3

Figure 11: Screen 4

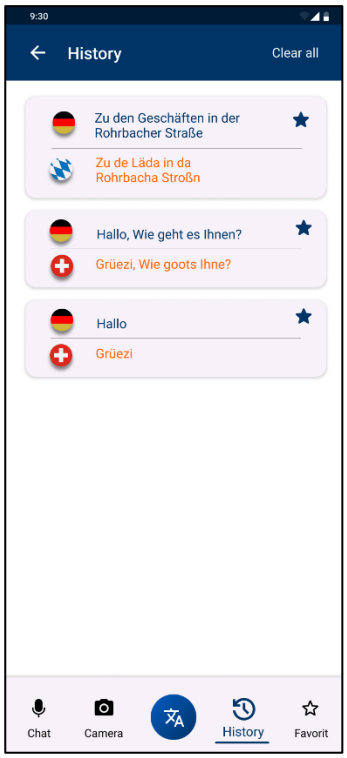

Figure 12: Screen 5

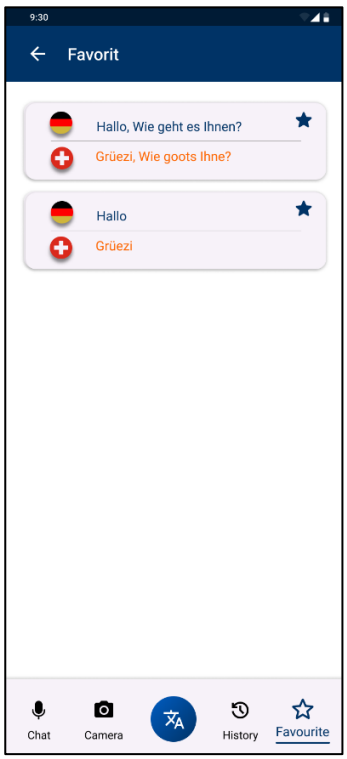

Figure 13:Screen 6

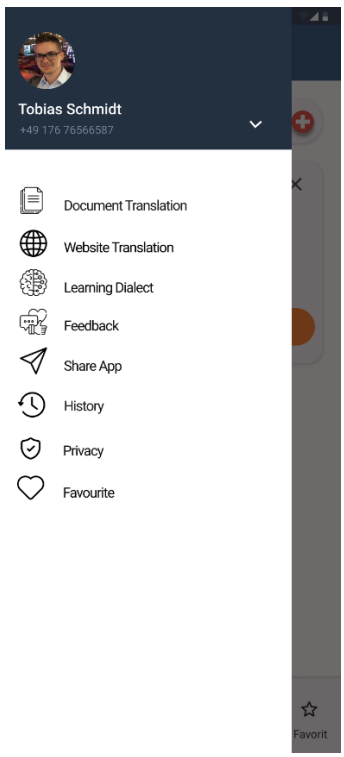

Figure 14: Screen 7

QUALITY ASSURANCE

Table 8: Quality Assurance Steps

1. Develop a testing plan Define the scope of testing and create a plan that includes test cases, test scenarios, and test data.	**2.Conduct functional testing** Conduct tests to ensure that the app's features, including translation accuracy and speed, work as expected.	**3.Perform compatibility testing** Test the app's compatibility with different devices, operating systems, and screen sizes.
4.Conduct usability testing Test the app's usability, including ease of navigation, readability, and user satisfaction.	**5.Perform localization testing** Test the app's functionality for different languages and ensure that the translations are accurate.	**6.Conduct performance testing** Test the app's performance under different conditions, including network speed, device memory, and battery life.

MAINTENANCE & SUPPORT

Table 9:Maintenance & Support Steps

1.Develop a maintenance plan Develop a plan that outlines the tasks and activities required to maintain and support the app, including bug fixes, security updates, and feature enhancements.	**2.Assign roles and responsibilities** Assign roles and responsibilities for maintaining and supporting the app, including developers, testers, and customer support staff.	**3.Set up a bug tracking system** Set up a system for tracking and resolving bugs and issues in the app, such as Jira or Bugzilla.
4.Monitor performance Monitor the performance of the app, including user feedback, usage metrics, and performance benchmarks, to identify areas for improvement.	**5.Keep up-to-date with technology** Keep up-to-date with the latest technology and industry trends, and incorporate new features and functionality into the app as needed.	**6.Provide customer support** Provide responsive and helpful customer support, including a helpdesk or support forum, to help users troubleshoot issues and get the most out of the app.
7.Conduct regular testing Conduct regular testing, including functional, compatibility, performance, and security testing, to identify and fix issues before they affect users.	**8.Release updates and patches** Release updates and patches to address issues and improve the app's functionality, usability, and security.	**9.Gather user feedback** Gather feedback from users to identify areas for improvement and prioritise feature requests.

BACK-END

The backend was developed using Python-based Flask framework, which helps in creating a RESTful API to allow communication between the front end and the back end. It receives the user query from the front end and processes it to generate the most relevant results.

The backend architecture is divided into three parts:

Natural Language Processing (NLP)

1) NLP Engine

The NLP engine of the backend is responsible for processing the user queries and extracting the requisite information for generating effective search results. In this engine, the data received from the user query undergoes several processes to make it more understandable for the system.

2) Pre-Processing Module

The pre-processing module of the NLP engine cleans the user query, and certain local language-specific rules are applied based on the local culture, usage, and context. The pre-processing module further segments the data into individual units like words and phrases, enabling better processing of the query by the system.

3) Machine Language (ML) Model

The core of the NLP engine is based on an ML model that helps the system understand the user's intent and context behind the query. The ML model is trained with an extensive corpus of data related to the local language, allowing it to recognize various language patterns, syntax, and usage. The ML model processes the user query and applies the learned language patterns to extract information accurately.

Information Retrieval Engine

The Information Retrieval Engine is responsible for making sense of the processed user query and returning the most relevant search results based on the query's context and intent. The Information Retrieval Engine employs several techniques like keyword matching, semantic analysis, and ranking algorithms like TF-IDF and BM25 to generate better search results.

- **Keyword Matching**

  The keyword-matching technique matches the user query with a pre-defined list of relevant keywords. This technique helps in identifying the query's intent and narrowing down the search results to a more specific set.

- **Semantic Analysis**

Semantic Analysis is used to understand the meaning behind the text beyond simple keyword matching. The semantic analysis technique employs ML models to associate the user query with a particular semantic category to generate more refined search results.

- **Ranking Algorithms**

  Ranking Algorithms are used to sort out the search results based on their relevance to the user query. Techniques like TF-IDF and BM25 use statistical calculations to determine the document's similarity to the query and rank them accordingly.

Database layer

The database layer of an AI-supported local language model project plays a critical role in storing and retrieving data. Key functions of the database layer include

- **Storing User Data**: The database layer stores user input data that is relevant to the project.
- **Storing Output Data**: Once processing of user input is complete and the desired output is generated, it is stored in the database.
- **Retrieving User Data**: The database layer retrieves user data as required to support backend processing.

The database will be developed using SQL, a widely used database management system. The database stores all the information required for the backend to generate search results. It consists of two main tables: one for storing the text data corpus, and the other for storing the user data like user profiles, search history, and feedback. They are given below

4) Corpus Table

The Corpus Table is used to store a vast collection of data relevant to the local language. This corpus data is extracted from various sources like books, newspapers, websites, and other public domain sources. The Corpus Table is used to provide a better understanding of the language to the AI models, allowing them to generate more appropriate search results.

2) User Data Table

The User Data Table is used to store the personal data of the users like profiles, search history, preferences, and feedback. The User Data Table helps in making the search results more personalized and relevant to the user. By analysing the user's search history, the AI models can learn about the user's interests and preferences and generate search results tailored to their needs.

Services and Technologies:

The working of a backend in the AI-supported local language model involves the integration of several services and technologies that work together to deliver the most relevant search results. Some of the essential services and technologies used in this model are

- **Python Flask Framework**: The Python Flask framework helps in developing a RESTful API that allows communication between the front end and the back end of the model.

- **Natural Language Toolkit (NLTK)**: The NLTK is a widely used open-source platform that provides various tools and resources for performing NLP tasks effectively.
- **TensorFlow**: TensorFlow is an open-source ML platform developed by Google used in training and developing ML models for NLP.
- **SQL/NoSQL Databases**: SQL databases like MySQL or Postgres are used to store structured data, while NoSQL databases like MongoDB or Cassandra are better suited for unstructured data.
- **AWS EC2**: The AWS EC2 is a secure cloud-based hosting service that provides high-speed server instances essential for deploying the AI model and managing user data.

As stated above the backend and database of this model play a crucial role in generating relevant search results and personalizing the user experience. The backend processes the user query, extracts the necessary information, and retrieves the most relevant search results based on the query's intent and context. The database's role is to store the local language corpus and user data to provide a more comprehensive understanding of the language and personalize the search results. In this model, several services and technologies like Python Flask, NLTK, TensorFlow, SQL Server, and AWS EC2 are used to ensure efficient and accurate processing of user queries.

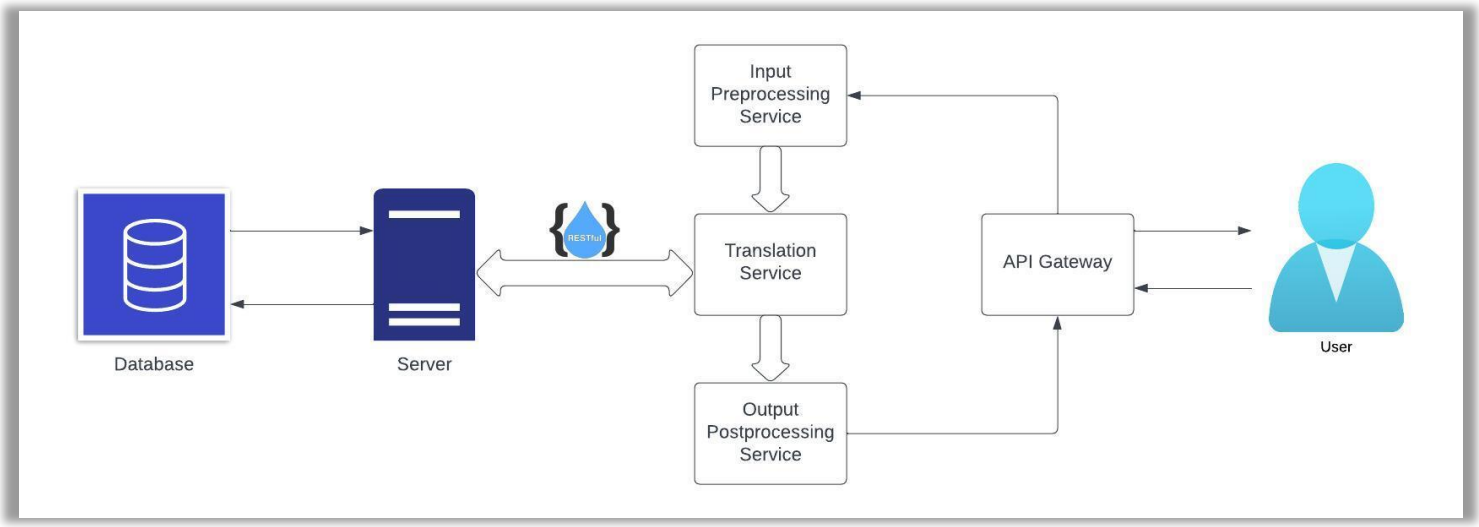

Figure 15: User-to-Server Flow Chart

Machine Translation Engine

Software that can translate texts from one language to another is called a machine translation engine.

These technologies have become more accurate as a result of the application of artificial intelligence. Currently, they are able to analyse vast volumes of data and turn them into information in order to make precise translations, including those that take speaker intent and semantics into account.

The integration of technologies like machine learning and deep learning increases the potential of a machine translation engine. These methods enable machine learning to be used by translation engines, resulting in results that are continually improved. However, excellent training is necessary to improve translation quality.

1. Incorporation of the base data:

Open-source software such as Pangeanic's ECO, together with NLP (Natural Language Processing) experts, have allowed organizations to create their own artificial intelligence and machine translation processes.

It is also possible to train systems using non-text data, however labelling of picture and video data during training must be accurate. A technique for creating suitable annotation and data segmentation must be developed.

Another sort of data that may be used to train a machine translation engine is voice data. Automatic speech recognition systems need a lot of high-quality audio data gathered in many circumstances and environments, therefore this is a unique approach.

2. Data cleaning and normalization:

Following the acquisition of raw data, unclean data should be cleaned up, and the data should be normalised. For instance, this method entails consistently utilising the appropriate quote marks for both languages. The required information may now be provided to the translation engine.

When uploading files to be trained, ECO automatically cleans the data and just requires that it be in the common XML-based translation format known as TMX (Translation Memory Exchange), a translation memory.

3. *Possibility of sentiment analysis*:

The ability for translation engines to grasp and account for the real meaning of a document or the speaker's purpose when translating is being made possible by more sophisticated technology. NLP and machine learning are coupled for this aim.

To enhance the quality of translation results, messages are categorised (as positive, negative, or neutral) and labelled when they are analysed (for example, obtained from social networks) to ascertain the mood or views of users.

PROJECT PLANNING

PROJECT MANAGEMENT METHODOLOGY

The Agile Method is a specific project management methodology used in the software development industry. This approach helps teams adapt to the volatility of software development. It employs regularly used incremental, iterative work sequences called sprints.

Additionally, the Agile methodology's flexibility makes it simpler to include requirement changes even during the development phase. Additionally, it offers chances to find application bugs during the early stages of development, improving delivery quality. Regular product reviews lead to higher customer satisfaction, which in turn boosts the value of the company.

The choice of the agile approach was made in order to have an incremental and iterative process for development and quick delivery of a functional software product. Additional benefits of agile are listed below.

- It is crucial to engage with customers in order to understand their needs and show them a functional software demo.
- It emphasises adapting quickly to change and ongoing improvement.
- It promotes teamwork and cross-training above solitary pursuits.

SCRUM PROCESS:

Scrum is a popular Agile methodology that uses a set of roles, events, artifacts, and rules to guide the development of a project. It is designed to be highly flexible and adaptable to changing requirements.

SCRUM places a strong focus on teams working closely with businesses to provide business values and involve them in the process so that we can produce excellent results. This team completes tasks at intervals of time known as sprints. In the daily stand-ups, the team can replan and report. Every sprint has a retrospective meeting when additional enhancements or optimisations are explored.

Below three roles are in SRCUM.

Product Owner: He or she is in charge of regularly informing the development team of the vision and properties. He or she is also the one who responds to all of the team's inquiries.

SRCUM master: He or she serves as a mediator between the team and the product owner, not as the team's manager.

Team: The team is autonomous and accountable for achieving the sprint's objectives. It incorporates programmers, analysts, QA specialists, software engineers, and software architects.

In order to identify unfavourable variations, Scrum users must often check Scrum artefacts and progress towards a Sprint Goal. Their examination shouldn't be conducted so frequently that it hinders their ability to do their job. When diligently carried out by qualified inspectors at the place of work, inspections are most useful.

Scrum users frequently need to monitor the status of a Sprint Goal and the Scrum artefacts in order to spot undesirable deviations. They shouldn't be examined so regularly that it interferes with their capacity to do their duties. Inspections are most beneficial when they are carefully conducted on-site by certified inspectors.

In order to identify unfavourable deviations, Scrum users typically need to keep an eye on the state of a Sprint Goal and the Scrum artefacts. They shouldn't be tested so often that it hinders their ability to do their job. The best inspections are those that are meticulously carried out on-site by trained inspectors.

PROJECT MANAGEMENT PROCESS MODEL

THE AGILE MODEL

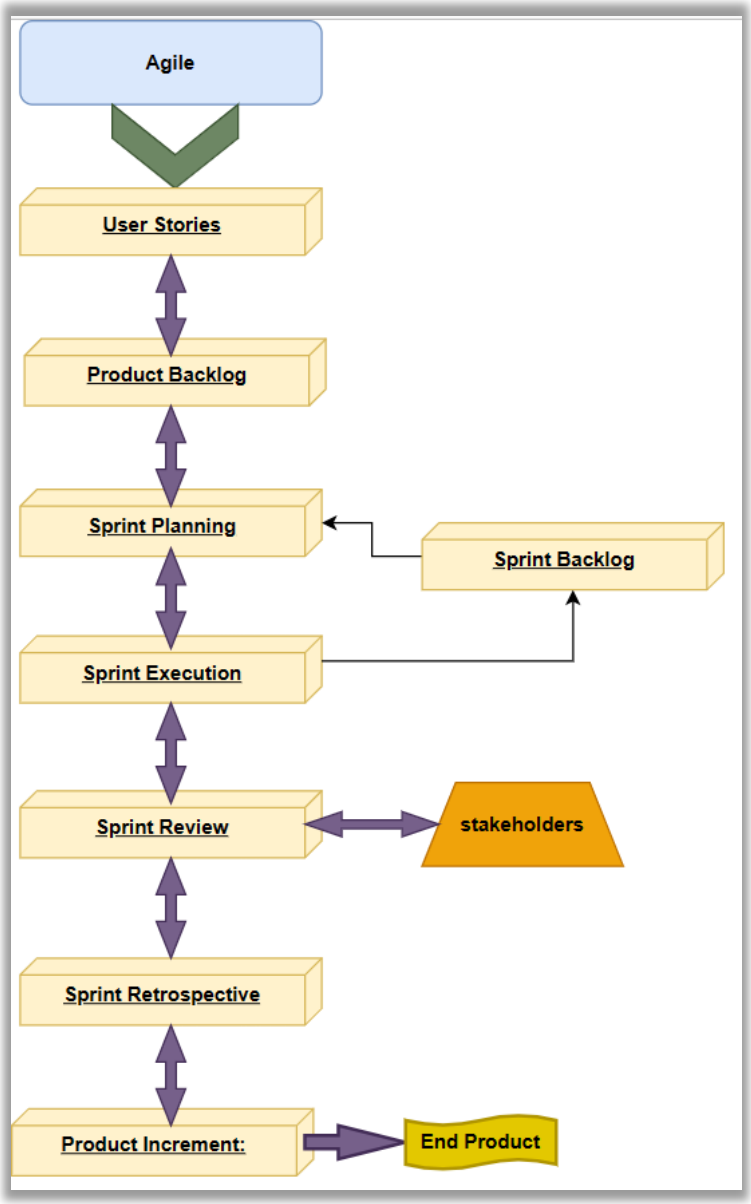

Figure 16: Agile Model

1. User Stories: The user stories are represented as input to the system. These stories are the starting point for the agile development process and represent the requirements for the system.

2. Product Backlog: The product backlog is the central repository for all user stories. It represents the prioritised list of user stories that will be worked on in the current sprint.

3. SCRUM Ceremonies -:

- **Sprint Planning:** During sprint planning, the product backlog is reviewed and the team selects the user stories that will be worked on during the current sprint.
- **Sprint Backlog:** The selected user stories are then moved to the sprint backlog. This represents the set of tasks that the team will work on during the current sprint.
- **Sprint Execution:** During the sprint execution, the team works on the tasks in the sprint backlog. As tasks are completed, they are marked as done and the progress is updated in the sprint backlog.
- **Sprint Review:** At the end of the sprint, a review is held to evaluate the progress and performance of the team. The review includes a demo of the completed tasks and feedback from stakeholders.
- **Sprint Retrospective:** After the review, the team holds a retrospective to discuss what went well and what could be improved for the next sprint. The retrospective can help the team to identify areas for improvement and to implement changes to the development process.

2. **Product Increment:** At the end of each sprint, a product increment is delivered. This represents the portion of the system that has been completed during the sprint and is ready for release to stakeholders.

Agile Methodology within Implementation Phase

Why Agile for IT development?

Agile methodology has been chosen for the app development and web-app development because:

- → The iterative feedback using this process will enhance the user experience and also improve quality.
- → It is open to improvisational changes. In today's technologically innovative and competitive environment, being open to changes is becoming a necessity rather than choice in IT developments. A daily stand-up call will be scheduled at the beginning of each business day in order to align the team and the product owners, and also probably set up short-term milestones.

Guidelines for the AGILE Methodology

- → Meet the customer's needs, continuously improve the software, and take advantage of shifting specifications to provide the client a competitive edge.
- → Focus on often providing functional software. The quickest time frame will be given consideration for delivery.
- → The entire process requires collaboration between businesspeople and developers.
- → Motivated individuals must be the foundation of all projects. Give them the environment and assistance they require. You should have faith in them to do their tasks.
- → Direct communication is the most effective technique to provide information inside and across teams. The main indicator of progress is working software.

PROJECT STRUCTURE PLAN

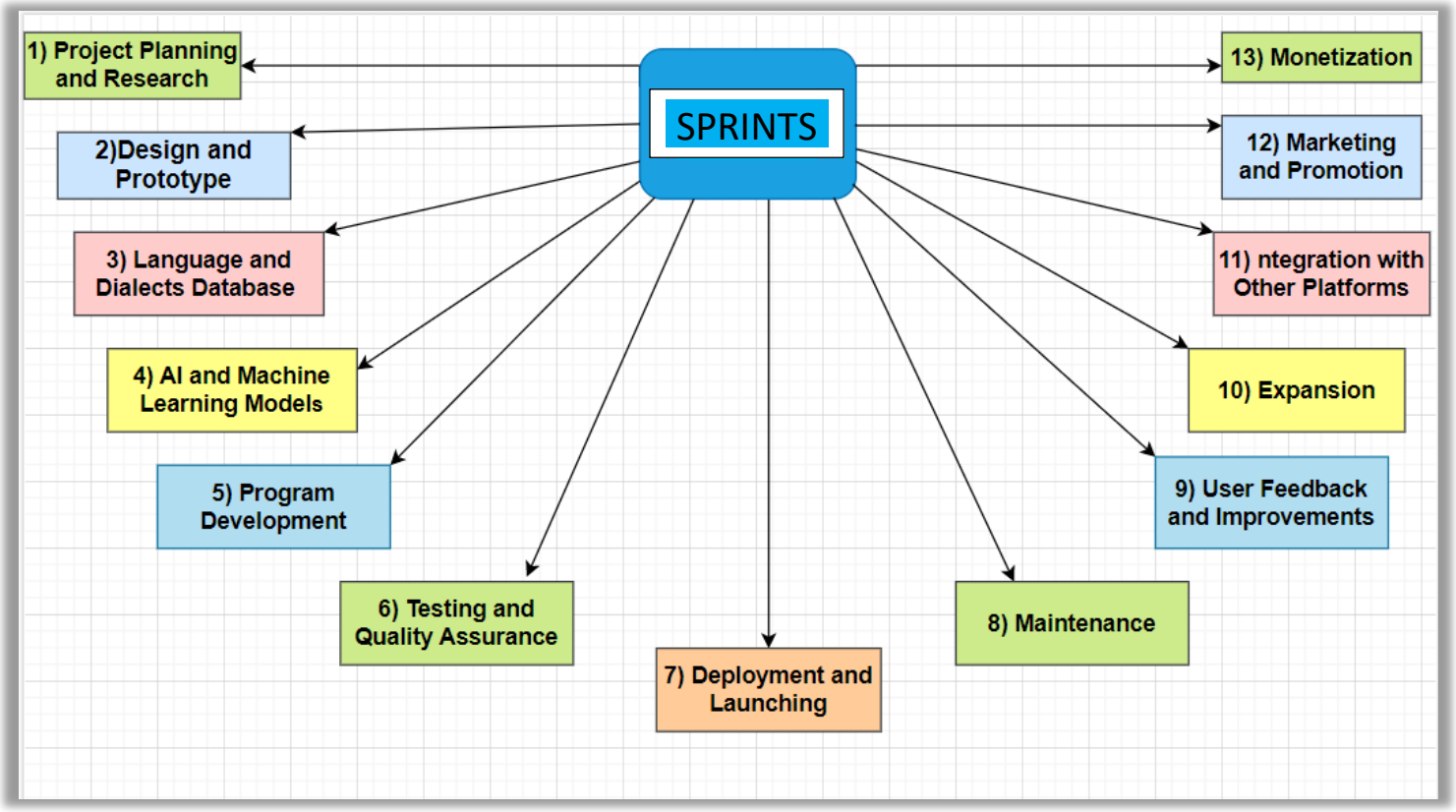

Figure 17: Project Structure Plan

Sprints :

Sprint 1: Project Planning and Research

- Conduct extensive research to determine the needs and expectations of the target audience, including identifying common dialect differences and linguistic challenges they face.
- Define the scope of the project, including the languages and dialects that will be included in the program
- Establish project objectives and develop a project plan that outlines the timelines, budgets, and resources required for the project.
- Identify and allocate the necessary resources, including the team members, technology, and tools required for the project.
- Assign roles and responsibilities to the team members and establish communication protocols to ensure effective collaboration throughout the project.
- Deliverables: Research report, feature list, and design mockups.
- Timeline: 4 weeks

- Upcoming sprint planning on next day

Sprint 2: Design and Prototype

- Develop a user interface (UI) design that is intuitive, user-friendly, and appealing to the target audience.
- Create a wireframe and mockup of the program to provide a visual representation of the final product.
- Develop a prototype of the program to test the UI design, user flow, and functionalities.
- Conduct user testing to gather feedback on the prototype and refine the design based on user feedback.
- Deliverables: Working front-end prototype.
- Timeline: 6 weeks
- Upcoming sprint planning on next day

Sprint 3: Language and Dialects Database

- Collect and compile a comprehensive database of the targeted languages and dialects, including regional variations and sub-dialects.
- Categorize the languages and dialects according to their regions and sub-regions to ensure accurate and efficient searching and filtering.
- Establish a database management system that can handle the volume and complexity of the data effectively and efficiently.
- Develop a data cleaning and verification process to ensure the accuracy and completeness of the data.
- Deliverables: Working Back-end prototype.
- Timeline: 8 weeks
- Upcoming Sprint planning on next day

Sprint 4: AI and Machine Learning Models

- Develop and implement machine learning models for speech recognition, translation, and pronunciation analysis, customized to handle the specific dialects and language variations targeted in the program
- Train the machine learning models using the collected data to ensure they can accurately recognize and analyze dialects and language variations
- Fine-tune the models based on user feedback and evaluation results to improve accuracy and effectiveness
- Test and validate the models to ensure their accuracy and reliability across a wide range of dialects and language variations
- Deliverables: Working Backend prototype.
- Timeline: 4 weeks
- Upcoming Sprint planning on next day

Sprint 5: Program Development

- Develop the program using the selected programming language and framework.
- Integrate the AI and machine learning models into the program to provide accurate translations and language analysis.
- Develop a user management system that allows users to create accoun.ts, log in, and track their progress and learning history.
- Implement personalized learning paths and progress tracking features to provide tailored language learning experiences for each user.
- Deliverables: Working prototype with integrated front-end and back-end.
- Timeline: 6 weeks
- Upcoming sprint planning on next day

Phase 6: Testing and Quality Assurance

- Conduct extensive testing of the program to identify and fix any bugs or errors that may arise during use
- Conduct usability testing with real users to gather feedback and improve user experience.
- Conduct security testing to ensure the program is secure and free from vulnerabilities.
- Conduct performance testing to ensure the program can handle a high volume of users and traffic and deliver accurate translations and language analysis quickly and efficiently.
- Deliverables: Bug reports and fixes, performance improvements, and user experience feedback.
- Timeline: 6 weeks
- Upcoming sprint planning on next day

Sprint 7: Deployment and Launch

- Deploy the program to a hosting environment that can handle the expected traffic volume and provide reliable and secure service to users.
- Configure the program for optimal performance and security, including optimizing database queries and encryption protocols.
- Launch the program and promote it through various channels such as social media, advertising, and partnerships.
- Monitor and evaluate user engagement, feedback, and performance metrics after launch to identify opportunities for improvement and expansion.
- Deliverables: Production-ready app and launch plan.
- Timeline: 3 weeks
- Upcoming sprint planning on next day

Sprint 8: Maintenance

- Description: Ensure the app is running smoothly and update it as necessary.
- Deliverables: Regular maintenance updates and bug fixes.
- Timeline: Ongoing after the app's launch.

Sprint 9: User Feedback and Improvements

- Description: Gather user feedback and make necessary improvements to the app.
- Deliverables: Updated app with improvements based on user feedback.
- Timeline: Ongoing after the app's launch.

Sprint 10: Expansion

- Description: Expand the app's capabilities to include more dialects/languages and improve its accuracy.
- Deliverables: Updated app with additional language/dialect support and improved accuracy.
- Timeline: After the app has been successfully launched and tested, and user feedback has been gathered and implemented.

Sprint 11: Integration with Other Platforms

- Description: Integrate the app with other platforms, such as social media and messaging apps.
- Deliverables: Updated app with integration capabilities.
- Timeline: After the app has been successfully launched, tested, and expanded.

Sprint 12: Marketing and Promotion

- Description: Develop a marketing and promotion strategy to increase app awareness and usage.
- Deliverables: Marketing and promotion materials, social media campaigns, and user acquisition strategies.
- Timeline: After the app has been successfully launched and tested, and user feedback has been gathered and implemented.

Sprint 13: Monetization

- Description: Develop a monetization strategy for the app, such as in-app purchases or advertising.
- Deliverables: Monetization plan and implementation.
- Timeline: After the app has been successfully launched, tested, and marketed.

  Overall, these milestones are designed to ensure the app is developed, tested, and launched successfully, while also providing ongoing maintenance and improvements to meet the needs of users.

SPRINT PLANNING

Table 10:Sprint Planning

Milestone	Tasks	Deliverables	Resources
Project Planning and Research	Conduct market research, identify target audience, define project scope, establish project timeline and budget, identify risks and constraints	Market research report, project scope document, project timeline and budget, risk and constraint matrix	Project Manager, Business Analyst
Design and Prototype	Develop wireframes and user interface design, create mockups and prototypes, get feedback from stakeholders, iterate design based on feedback	Wireframes, UI design, mockups and prototypes, stakeholder feedback report	UI/UX Designer
Language and Dialects Database	Identify languages and dialects to be included in the database, research language nuances and variations, collect and organize data	Language and dialects database, data organization plan	Linguists, Data Analyst
AI and Machine Learning Models	Develop and train AI and machine learning models, implement natural language processing algorithms, test and refine models	AI and machine learning models, natural language processing algorithms, model testing and refinement report	Data Scientist, AI Engineer
Program Development	Develop frontend and backend of the application, integrate AI models and language database, create user accounts and login functionality	Functional application with frontend and backend, user account and login functionality	Frontend Developer, Backend Developer
Testing and Quality Assurance	Conduct functional testing, performance testing, security testing, and usability testing, fix bugs and issues, ensure compliance with regulations and standards	Testing and quality assurance report, bug and issue log, compliance certification	Quality Assurance Engineer
Deployment and Launch	Prepare the application for deployment, launch the application on app stores and online platforms	Deployed and launched application	DevOps Engineer
Maintenance	Provide ongoing maintenance and support, fix bugs and issues, ensure scalability and stability	Maintenance and support plan, bug and issue log, scalability and stability report	Support Team
User Feedback	Collect user feedback and suggestions, prioritize	User feedback report, improvement plan, implemented improvements	Product Manager

and Improvements	improvements based on feedback, implement improvements		
Expansion	Expand the language and dialects database, add new features and functionalities, identify new target markets	Expanded language and dialects database, new feature and functionality plan, identified new target markets	Linguists, Product Manager
Integration with Other Platforms	Integrate with other language learning platforms, social media platforms, and translation tools	Integration plan, integration with other platforms	Integration Engineer
Marketing and Promotion	Develop marketing and promotional strategies, create marketing materials, run advertising campaigns	Marketing plan, marketing materials, advertising campaign report	Marketing Team
Monetization	Identify monetization strategies, implement monetization model, measure revenue and profitability	Monetization plan, implemented monetization model, revenue and profitability report	Business Analyst

SPRINTS

Table 11:Sprints

Tasks	Deliverables	Timeframe	Resources	Start Date	End Date
Project Planning and Research (SPRINT-1)		4 weeks (20 Days working)	Project Manager, Business Analyst	weekend dates not included	weekend dates not included
Conduct resources and market research	Market research report,	5 Days		01/05/2023	05/05/2023
Identify target audience and requirements	project scope document,	4 Days		08/05/2023	12/05/2023
Identification of stakeholders	stakeholders	3 Days		15/05/2023	17/05/2023
define project scope	project timeline and budget,	2 Days		18/05/2023	19/05/2023
establish project timeline and budget	risk,	4 Days		22/05/2023	26/05/2023
identify risks and constraints	constraint matrix ,	5 Days		26/05/2023	31/05/2023
Infrastructure hiring and training	hiring teams and training infrastructure	5 Days		01/06/2023	06/06/2023
Research & Analysis	Report & Training of project teams	1 Day		01/06/2023	01/06/2023

Sprint Planning Day					
Design and Prototype (SPRINT-2)		6 weeks (30 Days working)	UI/UX Designer	weekend dates not included	weekend dates not included
Develop wireframes and user interface design	Wireframes and UI design,	10 Days		07/06/2023	20/06/2023
create mockups and prototypes	mockups and prototypes,	10 Days		07/06/2023	20/06/2023
get feedback from stakeholders	stakeholder feedback report	4 Days		21/06/2023	26/06/2023
iterate design based on feedback	asked changes applied	6 Days		27/06/2023	04/07/2023
Prototype	Finalising UI/UX	1 Day		05/07/2023	05/07/2023
Sprint Planning Day					
Language and Dialects Database (SPRINT-3)		8 weeks (40 Days working)	Linguists, Data Analyst	weekend dates not included	weekend dates not included
Identify languages and dialects to be included in the database	Language and dialects database	18 Days		05/07/2023	28/07/2023
research language nuances and variations	report of variation	10 Days		05/07/2023	18/07/2023
collect and organize data	data organization plan	12 Days		31/07/2023	11/08/2023
Sprint Planning Day					
AI and Machine Learning Models (SPRINT-4)		4 weeks (20 Days working)	Data Scientist, AI Engineer	weekend dates not included	weekend dates not included
Develop and train AI and machine learning models	AI and machine learning models,	8 Days		14/08/2023	23/08/2023
implement natural language processing algorithms	natural language processing algorithms,	8 Days		14/08/2023	23/08/2023
test and refine models	model testing and refinement report	4 Days		24/08/2023	29/08/2023
Sprint Planning Day					
Program Development (SPRINT-5)		6 weeks (30 Days working)	Frontend Developer, Backend Developer	weekend dates not included	weekend dates not included
Develop frontend of the application	Functional application with frontend	10 Days		30/08/2023	12/09/2023
Develop backend of the application	Functional application with backend,	10 Days		30/08/2023	12/09/2023

integrate AI models and language database	AI models with database management	5 Days		13/09/2023	19/09/2023
create user accounts and login functionality	user account and login functionality	5 Days		20/09/2023	26/09/2023
App Integration	Integrating back-end, front-end and cloud	1 Day		26/09/2023	26/09/2023
Sprint Planning Day					
Testing and Quality Assurance (SPRINT-6)		6 weeks (30 Days working)	Quality Assurance Engineer	weekend dates not included	weekend dates not included
Conduct functional testing	assurance report	8 Days		27/09/2023	06/10/2023
performance testing		8 Days		27/09/2023	06/10/2023
security testing		8 Days		09/10/2023	18/10/2023
usability testing		8 Days		09/10/2023	18/10/2023
Sprint Planning Day					
Deployment and Launch (SPRINT-7)		3 weeks (15 Days working)	DevOps Engineer	weekend dates not included	weekend dates not included
Prepare the application for deployment	Deployed application	8 Days		19/10/2023	30/10/2023
Upto Deployment		1 Day		30/10/2023	30/10/2023
Deployment of application on app stores and online platforms	launched application	7 Days		31/10/2023	08/11/2023
Sprint Planning Day					
Maintenance (SPRINT-8)		Ongoing	Support Team	weekend dates not included	weekend dates not included
Provide ongoing maintenance and support	Maintenance and support plan			08/11/2023	Till the future
fix bugs and issues	bug and issue log	Weekend Activity		13/11/2023	14/11/2023
ensure scalability and stability	scalability and stability report			15/11/2023	16/11/2023
Sprint Planning Day					
User Feedback and Improvements		Ongoing	Product Manager	weekend dates not included	weekend dates not included

(SPRINT-9)					
Collect user feedback and suggestions	User feedback report,			17/11/2023	Till the future
implement improvements accordingly	implemented improvements			17/11/2023	Till the future
Sprint Planning Day					
Expansion (SPRINT-10)		Ongoing	Linguists, Product Manager	weekend dates not included	weekend dates not included
Expand the language and dialects database	Expanded language and dialects database,	Weekend Activity		08/01/2024	09/01/2024
add new features and functionalities	new feature and functionality plan,	Weekend Activity		15/01/2024	16/01/2024
identify new target markets for collaboration	identified new target markets	Weekend Activity		22/01/2024	23/01/2024
Sprint Planning Day					
Integration with Other Platforms (SPRINT-11)		Ongoing	Integration Engineer	weekend dates not included	weekend dates not included
Integrate with other language learning platforms	Integration plan	Weekend Activity		05/02/2024	06/02/2024
Integrate with social media platforms	integration with other platforms	Weekend Activity		19/02/2024	20/02/2024
Integrate with translation tools and softwares	integration with other tools	Weekend Activity		26/02/2024	27/02/2024
Monitorization	User feedback Integration	1 Day		27/02/2024	27/02/2024
Sprint Planning Day					
Marketing and Promotion (SPRINT-12)		Ongoing	Marketing Team	weekend dates not included	weekend dates not included
Develop marketing and promotional strategies	Marketing plan			31/10/2023	Till the future
create marketing materials	marketing materials,			01/12/2023	25/02/2024
run advertising campaigns	advertising campaign report			01/03/2024	01/04/2024
Sprint Planning Day					
Monetization (SPRINT-13)		Ongoing	Business Analyst	weekend dates not included	weekend dates not included

Identify monetization strategies	Monetization plan,			01/04/2024	15/04/2024
implement monetization model	implemented monetization model,			16/04/2024	23/04/2024
Revenue Generation	Managing Revenue Income	1 Day		17/04/2024	17/04/2024
measure revenue and profitability	revenue and profitability report			24/04/2024	30/04/2024

GANTT CHART

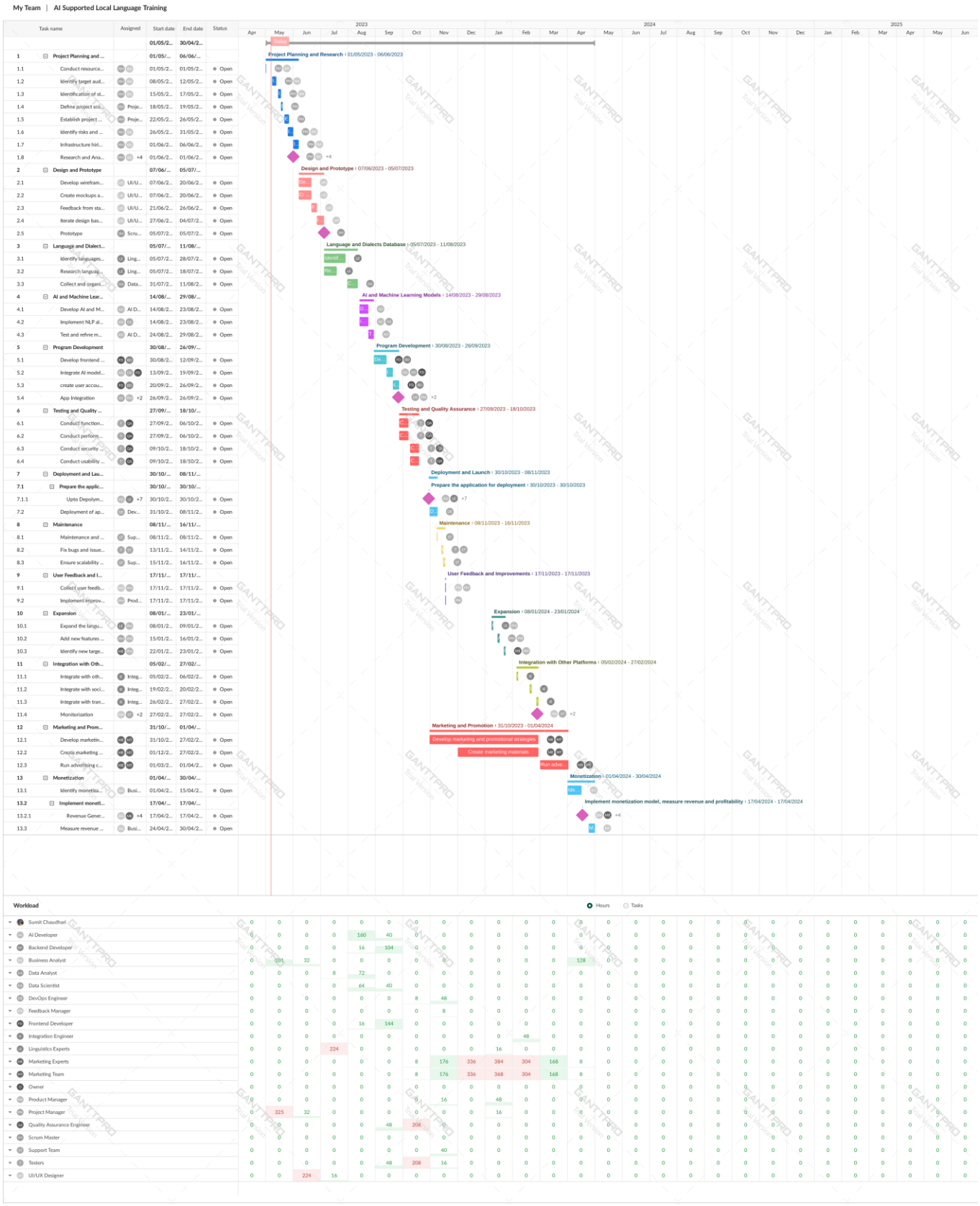

Figure 18: Gantt Chart

NETWORK PLANNING

In order to guarantee that the programme may be accessible by users from any location on Earth, the target infrastructure must offer quick and dependable network connectivity.

- This could include: Content delivery networks (CDNs) for quickly and effectively delivering programme content to users.
- Load balancers for allocating user requests among many instances of the programme in order to provide high availability and scalability.
- HTTPS and SSL/TLS safe network technologies to provide secure connection between users and the programme.

TEAM CONSTELLATION

1. Project Management:

The project is supervised by the project management group. To assure the success of a project, they plan, manage, and control the resources, schedules, budgets, and timetables. Additionally, they are in charge of updating stakeholders on project status, reporting progress, and resolving any problems that may arise. They want to make that the project achieves its objectives, adheres to its budget, and is delivered on schedule.

2. IT and Development Team

Software development, testing, and deployment are the responsibilities of the IT development team. To make sure requirements are met, they collaborate closely with the project management team and end users. They also guarantee the best performance, scalability, and dependability of software. They are essential in keeping the program up to date and resolving any problems.

The IT and development team will be responsible for the development and deployment of the application using an agile methodology.

The team will comprise the following roles:

→ Project manager: Oversees the project's progress, resource allocation, and deadlines.
→ AI specialists: Develop, train, and test the translation engine.
→ Linguistic experts: Guide language, dialects, and cultural nuances.
→ Quality assurance specialists: Test, identify, and rectify issues with the application.

3. Integration Team

The integration team is in charge of integrating the program with external systems including servers, databases, and applications from other companies. They make that the program integrates easily with other tools and that data is synced between platforms. Additionally, they test the software in various settings to make sure it works well in a variety of situations

The integration team's role is to ensure smooth integration of the AI engine with the mobile and web applications. This team will include:

- **System architects**: Design the system architecture and API interfaces.
- **Full-stack developers**: Implement the API calls to interact with the AI engine.
- **UI/UX designers**: Create user-centred designs for seamless application usage.

4. Marketing Team:

The marketing group is in charge of publicising the program and luring potential clients. To generate leads and build brand awareness, they develop marketing plans, programs, and materials. To make sure that the marketing message is in line with the features and benefits of the product, they collaborate closely with the IT development team and the project management team.

BUDGET PLANNING

The cost of creating an AI-supported dialect training and translation application can have an impact on the development process and the features that can be added to the finished product.

Table 12:Budget Planning

Resources	Duration in months	Units	Cost per Month per Unit (Euros)	Total Cost(Euros)
Business Analyst	6	2	4970	59.641
UI Designer	6	2	3775	45.301
Project Manager	11	1	5416	59.576
Scrum Master	10	1	5416	54.161
legal advisor	8	1	6438	51.504
Developers (AI,Frontend,Backend)	7	6	4545	190.891

Quality Assurance Engineer	6	3	4111	73.998
Data Analyst	11	4	4583	201.652
DevOps Engineer	8	2	6250	100.001
Testers	3	4	4560	54.721
Linguistics Experts	11	6	3753	247.698
Marketing Experts	6	2	3984	47.808
Technical Architect	6	1	6997	41.982
Support Team	6	4	5784	138.816
Content Administrator	10	1	3333	33.331
Work Space	11	1	600	6.601
Internet / WLAN	11		1100	12.101
Total				**1.419.783**

PROJECT MANAGEMENT

1.Risk Management:

The first step is to identify potential risks that could impact the AI based project's success. This may include risks related to data quality, algorithmic bias, cybersecurity, and privacy.

The risk management plan should be implemented throughout the project, and risks should be continuously monitored and evaluated to ensure that the plan is effective.

As the project progresses, new risks may emerge or existing risks may change. It's important to regularly review and update the risk management plan to ensure that it remains relevant and effective.

Documenting all the steps taken for risk management and making it accessible to relevant stakeholders can help in transparency and accountability.

Overall, effective risk management for an AI-based project requires a proactive approach and ongoing monitoring to identify and mitigate potential risks throughout the project's lifecycle.

2.Communication Management:

To maintain efficient teamwork throughout the project, assign roles and tasks to the team members and develop communication procedures.

Develop a communication plan that outlines the stakeholders, their communication needs, and the frequency and mode of communication. This plan should be updated as the project progresses and new stakeholders are identified.

Clearly define the roles and responsibilities of each team member in the communication plan. This includes identifying the project manager, project sponsor, technical leads, and other stakeholders who will be responsible for different aspects of the project.

Use collaborative tools like project management software, chat platforms, and video conferencing tools to facilitate communication and collaboration among team members. These tools can help streamline communication and ensure that everyone is up-to-date on the project's progress.

Encourage stakeholders to provide feedback and address concerns as they arise. This can help ensure that any issues are addressed in a timely manner and that the project stays on track.

Overall, effective communication management is critical to the success of an AI project. By establishing clear roles and responsibilities, using collaborative tools, holding regular meetings, sharing updates, and addressing concerns and feedback, project teams can ensure that all stakeholders are aligned on the project goals and progress.

3.Configuration management:

Configuration management is a process that involves identifying, organising, and tracking changes to the AI system's configuration, including software, hardware, and data.

Optimise the program's security and speed by configuring it to use efficient database searches and encryption techniques.

Version control: This involves creating a version history for each component of the system, including the model, data, and code, and tracking changes over time.

Change management: This involves documenting change requests, assessing the impact of changes, and obtaining approval before making changes to the system.

Configuration identification: Each component of the AI system should be identified and catalogued, including the model, data, and code. This enables changes to be tracked and managed throughout the system's life cycle.

Configuration control: This involves establishing roles and responsibilities for making changes, implementing access controls, and monitoring changes to the system.

Configuration status accounting: The status of the AI system's configuration should be tracked and reported throughout its lifecycle. This includes information such as the current version of the system, the status of changes, and any issues or defects that have been identified.

Configuration audit: Regular configuration audits should be conducted to ensure that the AI system's configuration is accurate and up-to-date. This involves reviewing the system's configuration against its documentation, identifying any discrepancies, and taking corrective action as necessary.

4.Test Management:

Developing a comprehensive test strategy is crucial for effective test management in AI. This involves defining the testing objectives, identifying the testing scenarios, and determining the types of tests that will be performed. This involves selecting relevant data sets, pre-processing the data to ensure it is consistent, and partitioning the data for training and testing.

Performing tests on AI models involves running simulations or experiments to validate their performance. This includes functional testing, performance testing, and reliability testing.

automated testing can help accelerate the testing process and improve accuracy. This involves developing test scripts and tools that can be used to automate various testing activities.

Test results need to be documented and reported in a clear and concise manner. This involves creating reports that summarise the test results, highlighting any issues or defects that were identified during testing.

Conduct functional testing: To guarantee that all features and functions operate as intended, use testing tools.

Conduct performance testing: To make sure the app, bot, or website works without latency, use performance testing tools.

Conduct user acceptance testing: To get input from the target audience and make the required modifications, hire focus groups.

Conduct post-deployment testing: Use testing tools to ensure the app/bot/website works correctly in the target environment.

To make sure the app, bot, or website functions properly and satisfies the demands of the target audience for language learning, the development team will carry out functional testing, performance testing, and user acceptability testing throughout the testing process.

To verify the models' correctness and dependability across a variety of dialects and linguistic variants, test and validate them.

EXECUTIVE SUMMARY

Today's AI-supported application is a ground-breaking method for bridging the gap between various languages and dialects. Understanding and using regional dialects to communicate has grown in importance as a result of globalisation and greater mobility, which has made communication more crucial than ever. It is necessary to assist people in achieving this aim by offering a thorough and efficient learning environment that is simple to use and takes advantage of the most recent technological developments.

Our research intends to create an **AI-supported language tool** that can make it simple for people to pick up and comprehend regional languages. To deliver a smooth and efficient learning experience, the solution will make use of cutting-edge technologies. To guarantee that the solution is capable of effectively capturing the intricacies of regional dialects, our team will collaborate with linguists and native speakers. No matter where they are located or their level of socioeconomic standing, everyone will be able to take advantage of the solution since it will be created to be user-friendly, affordable, and accessible.

Our Project will aid users who need to translate signs and menus, transcribe one dialekt into written text form in another dialekt, provide real time voice translation in different dialects and detecting text in an image and converting into another dialects.

There are various risks associated with dialect translation such as challenges in translating scientific, technical and medical terminologies. It's obvious to take care of cultural sensitivity, political and religious correctness, data breaches and security, informal speech and slang recognization, wrong pronounciation, speech synthesis for older users and issue of translating long and complex sentences.

Our Sprints include Project Planning & Research, Design & Prototype, Language & Dialect Database, AI & Machine Learning Models, Program Development, Testing & Quality Assurance, Deployment & Launch, Maintenance, User Feedback and Management, Expansion, Integration with Other Platforms, Marketing & Promotion and Monetization.

The project duration will last almost **one year (01.05.2023-30.04.2024).** We'll use **Agile Methodology** as Agile Methodology encourages collaboration between team members, including developers, testers and stakeholders. This can help us ensure that everyone is aligned with project goals, as well as coordinate our development process well.

The project costs are estimated at **EUR 1.419.783**. Potential risks include inaccuracies in translation, security and privacy issues, cultural sensitivity and reliance on frequent technological development in Natural Language Processing. We will implement risk monitoring and control systems to ensure that risks are addressed in a timely manner and that the development team remains responsive to changing risks throughout the development process.

REFERENCES

Table 13:References External Link

S. NO	PAPER NAME	LINK	DATE	TIME
1	SPEECH RECOGNITION SYSTEMS – A COMPREHENSIVE STUDY OF CONCEPTS AND MECHANISM	https://www.researchgate.net/publication/331679755_SPEECH_RECOGNITION_SYSTEMS_-_A_COMPREHENSIVE_STUDY_OF_CONCEPTS_AND_MECHANISM	22-04-2023	21:41
2	Speech and Language Processing.	https://web.stanford.edu/~jurafsky/slp3/A.pdf	22-04-2023	21:49
3	Hidden Markov Model (HMM) For NLP Made	https://spotintelligence.com/2023/01/05/hidden-markov-model-hmm-nlp/	22-04-2023	21:55
4	Sanskrit Speech Recognition using Hidden Markov Model Toolkit	https://www.semanticscholar.org/paper/Sanskrit-Speech-Recognition-using-Hidden-Markov-Pokhariya-Mathur/8b2b84790d16d2728b4418cd22cefff19f0ff1c6	24-04-2023	01:34
5	A Review on Automatic Speech Recognition System in Indian Regional Languages	https://www.semanticscholar.org/paper/A-Review-on-Automatic-Speech-Recognition-System-in-Siddharth-Prashantkumar/b09faa4989fd40c5a846317dfc6fe7f1c2748a9c	24-04-2023	01:35
6	Toward an automatic speech recognition system for amazigh-tarifit language	https://www.semanticscholar.org/paper/Toward-an-automatic-speech-recognition-system-for-Ouahabi-Atounti/bb6048ea84b9b7728b6693b5f1db887823c6a81f	24-04-2023	02:07
7	A survey of voice translation methodologies — Acoustic dialect decoder	https://www.semanticscholar.org/paper/A-survey-of-voice-translation-methodologies-%E2%80%94-Krupakar-Rajvel/f51ac7420962e9e566991b59229dd77e76aaa847	24-04-2023	02:08
8	A Comparison of Speech-to-Speech Neural Network Methodologies for Digit Pronunciation	https://www.semanticscholar.org/paper/A-Comparison-of-Speech-to-Speech-Neural-Network-for-Quintana-Bernal/9c7c8b25c279d931b3f83f5ae05b8dc0746c677b	24-04-2023	02:15
9	Addressing the Rare Word Problem in Neural Machine Translation	https://www.semanticscholar.org/paper/Addressing-the-Rare-Word-Problem-in-Neural-Machine-Luong-Sutskever/1956c239b3552e030db1b78951f64781101125ed	24-04-2023	02:19
10	Neural Machine Translation by Jointly Learning to Align and Translate	https://www.semanticscholar.org/paper/Neural-Machine-Translation-by-Jointly-Learning-to-Bahdanau-Cho/fa72afa9b2cbc8f0d7b05d52548906610ffbb9c5	25-04-2023	04:59

11	Speech synthesis techniques. A survey	https://www.semanticscholar.org/paper/Speech-synthesis-techniques.-A-survey-Tabet-Boughazi/da477d8e3a8101df3b042aa299e5be92bb4fbcf6	25-04-2023	05:01
12	Performance evaluation and comparison of multilingual speech synthesizers for Indian languages	https://www.semanticscholar.org/paper/Performance-evaluation-and-comparison-of-speech-for-Jeeva-Ramani/949af9ca3090a2d45d7fa2bb2e1ad8e7adae5827	25-04-2023	05:09
13	A Survey :On Text To Speech Synthesis	https://www.semanticscholar.org/paper/A-Survey-%3AOn-Text-To-Speech-Synthesis-Anand-Bhadran/75d874177c001ed79ab259100436a4257b2eaf97	26-04-2023	05:26
14	High-quality text-to-speech synthesis : an overview	https://www.semanticscholar.org/paper/High-quality-text-to-speech-synthesis-%3A-an-overview-Dutoit/36cb0e3d47a76bc1e87529cbb2fd965a37ade12a	26-04-2023	05:34
15	Review of text-to-speech conversion for English.	https://www.semanticscholar.org/paper/Review-of-text-to-speech-conversion-for-English.-Klatt/5657f5888e198fecf4612ff04c4b0bdef972147c	27-04-2023	08:26
16	Speech synthesis using stochastic Markov graphs	https://www.semanticscholar.org/paper/Speech-synthesis-using-stochastic-Markov-graphs-Eichner-Wolff/9ce35dc0b28bd8e8ab9ead2c85194f5deb891d04	27-04-2023	20:30
17	A study on automatic speech recognition toolkits	https://www.semanticscholar.org/paper/A-study-on-automatic-speech-recognition-toolkits-Ganesh-Sahu/15de894645bde5e63677ccff47a97fd65bd26f86	28-04-2023	20:46
18	Artificial Intelligence Based Language Translation Platform	https://www.researchgate.net/publication/350555260_Artificial_Intelligence_Based_Language_Translation_Platform	28-04-2023	21:14
19	Exploring the Benefits and Challenges of AI-Language Learning Tools	https://www.researchgate.net/publication/366957798_Exploring_the_Benefits_and_Challenges_of_AI-Language_Learning_Tools	29-04-2023	11:30
20	The Role of Artificial Intelligence in Translation Services: Revolutionizing the Way We Communicate	https://www.daytranslations.com/blog/the-role-of-artificial-intelligence-in-translation-services/	30-04-2023	12:02