BAHIR DAR UNIVERSITY
COMPUTING FACULTY

Industrial project on **Bidirectional Language Translation**

Submitted to the faculty of computing in partial fulfillment of the requirements for the degree of

Bachelor of Science in **Software Engineering**

Group members:

No	Name	ID Number
1	Daniel Getaneh	BDU1102203
2	Yezibalem Aemro	BDU1102502
3	Yeabsira Aychiluhim	BDU1102283

Advisor ꞉ Mr. Mulugeta Muche

2015/2022

Bahir Dar University, Bahir Dar Institute of Technology

Declaration

The Project is our own and has not been presented for a degree in any other university and all the sources of material used for the project have been duly acknowledged.

Daniel Getaneh

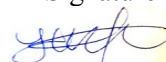Name   Signature

Yezibalem Aemro

Name   Signature

Yeabsira Aychiluhim

Name   Signature

Faculty: Computing

Program: Software Engineering

Project Title: Bidirectional Language Translation

This is to certify that I have read this project and that in my supervision and the students' performance, it is fully adequate, in scope and quality, as a project for the degree of Bachelor of Science.

Mulugeta Muche

Name of Advisor   Signature

NO.	Examining committee members	signature	Date
1			
2			

It is approved that this project has been written in compliance with the formatting rules laid down by the faculty.

Roles and Responsibilities

Table 1: List of Students with their respective Tasks

List of Tasks	List of Members		
	Daniel Getaneh	**Yezibalem Aemro**	**Yeabsira Aychiluhim**
Gathering databases	√	√	
Train the dataset by AI	√		
App drawer			√
Homepage of the app	√		
Learn Ge'ez page		√	√
Setting app page		√	
About us page	√		√

Acknowledgment

List of Acronym

- **AD –** Activity Diagram
- **AI** – Artificial Intelligence
- **BRD** – Business Rule Documentation
- **CRC** – Class Responsibility Collaborator
- **FR** – Functional Requirement
- **GB** – Giga Byte
- **iOS** – iPhone Operating System
- **Linux** – Lovable Intellect Not Using XP
- **MacOS** – Macintosh Operating System
- **NFR** – Non-Functional Requirement
- **OO –** Object Oriented
- **OOA** – Object Oriented Analysis
- **OOD** – Object Oriented Design
- **OS** – Operating System
- **PC** – Personal Computer
- **RAM** – Random Access Memory
- **SD –** Sequence Diagram
- **UI** – User Interface
- **UML –** Unified Modeling Language
- **Windows** – Wide Interactive Network Development for Office Work Solution.

<div align="center">

List of Figures

</div>

Table of Figures

List of Tables

Index of Tables

Table of Contents

Abstract

Language is a system of communication which consists of a set of sounds and written symbols which are used by the people of particular country for talking or writing. The ability to communicate in multiple languages is becoming more and more important in the increasingly integrated global community.

A language translator is a mobile application that can be utilized for translating from one language to another. It is a very important tool for anyone who is learning a new language. The problem of language difference has hindered effective information communication over the years. In Ethiopia, there are above 85 languages. But, despite the existence of many languages in the country, many are still struggle to write, speak or listen these languages properly. The study develops flutter language converter app in order to deal with this problem and make language translation easy and facilitates stress-free communication, it is advisable to use several languages learning software's. There are different flutter based language translator applications. However, the existing apps were restricted to a set of words that were hard coded into the app at the time of its development. We are developing flutter app which is compatible with **Android**, **iOS**, **Linux**, **Windows**, **MacOS** and **Web**.

This project aims to create a comprehensive translation of Ge'ez language to Amharic language and vice versa. It will include a variety of resources to help people learn and understand the language and its respective nuances. The project will be organized into different sections that cover the basics of language structure, grammar, vocabulary, and common phrases. Furthermore, it will provide a glossary of words and phrases, as well as useful examples of usage. This flutter framework based application will also include using AI transformer based model for translation of the two language. The project is designed to be comprehensive and user friendly, with the goal of making it a valuable resource for anyone looking to learn and understand the Ge'ez and Amharic language.

With this documentation, we will try to explain about our projects with three chapters, the first chapter wants to describe background, objectives, methodologies that we use during the project lifetime. The second chapter explains the system logic by comparing with the existing applications. Finally, the third chapter explains about the overall system designs of the application. We finished this documentation with the references that used during the development of this project and appendices of some useful words that is known in IT world.

1.   Chapter One: Introduction

1.1. Background

Language is a system of communication which consists of a set of sounds and written symbols which are used by the people of particular country for talking or writing. The primary purpose of language is to communicate between individuals, but language can be used in several specific ways to get certain kinds of ideas. Language allows people to: express emotion, share our ideas, thoughts, and feelings with others. The ability to communicate in multiple languages is becoming more and more important in the increasingly integrated global community.

Translator is a very important tool for anyone who is learning a new language. It is such a great app especially for the one who want to learn Ge'ez but can't. In a translator, the user can look up a word from the target language. Translation plays an important role in human life because it has made communication among people with different languages in reality. Both spoken and written translation, translation activities become a tool to create optimal communication. The way in which we communicate with people is important because it can affect the relationship we have the effectiveness of our own and others' work and of the organizations in which we work the translation is important to both language and culture because it enables better communication between two people. Because of translation as well, the equal understanding of the development of innovations in all aspects becomes easy to be understood and followed.

Translation means transferring message from source language into Target language with the nearest equivalent. The aim of translation is to find the equivalent meaning of the source language expression in the target language. Furthermore, translating a literary work into another language is creating a new literary work in another language. One can find, understand and learn the meaning of one word in another language among the two. It serves a good opportunity for student's researchers and other individuals who are on the field of language study.

1.2. Objectives

Project objectives are what you plan to achieve by the end of your project. This might include deliverables and assets, or more intangible objectives like increasing productivity or motivation.

1.2.1 General Objectives

The general objectives of this project is to create a reliable and accurate bidirectional language translation system between Ge'ez and Amharic languages, which can effectively translate text from one language to the other, while maintaining the original meaning and context of the text.

1.2.2 Specific Objectives

To achieve the general objectives of the project, the following specific objectives are formulated

  ✔ To identify the syntactic relationship between Ge'ez and Amharic languages

✔ To design an optimal language and translation AI model.

✔ Find a solution to the existing problem.

✔ Design and train a transformer-based deep learning model using the TensorFlow package in Python to effectively translate text between Ge'ez and Amharic languages.

✔ Optimize the translation model's performance on mobile devices by using techniques such as quantization, model compression, and runtime optimizations.

✔ Develop a data preprocessing pipeline within the mobile application to clean and normalize the input text, including tokenization, stemming, and lemmatization, to ensure the input is in a standard format for the deep learning model to process.

✔ Format the trained deep learning model into a .tflite file for faster and more efficient inference on mobile device and low-power systems.

✔ Integrate the dictionary and deep learning model into a user-friendly and accessible web, mobile, desktop application that can accept user input and provide translations in real-time.

✔ Evaluate the accuracy and effectiveness of the developed translation system using standard metrics such as BLEU and TER score, and compare its performance with existing translation systems.

✔ Implement a post processing pipeline within the mobile application to generate the final translated text, including detokenization, denormalization, and paraphrasing, to produce natural and fluent output text.

✔ Continuously improve and update the system by adding new words and phrases to the dictionary, fine-tuning the deep learning model, and incorporating user feedback.

✔ Develop a comprehensive dictionary of commonly used words and phrases in Ge'ez and Amharic languages, which can be used as a reference for translation purposes, and embed it into the mobile application.

✔ Implement a user-friendly interface that can accept user input and provide translations in a real-time, using Flutter's widget toolkit and UI components.

By achieving these objectives, the project will create a reliable and accurate bidirectional language translation system that can facilitate communication, promote the research with past cultures and norms and promote cultural exchange between Ge'ez and Amharic speakers.

1.3. Statement of the Problem

Ge'ez is an ancient language and many manuscripts are already archived by Ethiopian Orthodox Church as well as by the National Archival agency. Ge'ez had been known as being used in Ethiopia since the 4th century and as a spoken language close to a thousand years and had been serving as official written language practically up to the end of 19th century.

Since currently there are a lot of historical, cultural and religious documents available in Ge'ez language, there is a need to translate the manuscripts to Amharic and other Ethiopian Languages to make the decoded knowledge accessible to every especially Amharic users. On the other hand ,Ge'ez as a language being researched and taught in different Universities around the word in terms of accessing the decoded knowledge. Indirectly, Ge'ez language speakers are being created therefore; there is also a need to translate Amharic documents to Ge'ez language.

Today Ge'ez remains only as a spoken language and the liturgy language of the Ethiopian Orthodox Tewahedo Church. Whereas, Amharic is among the most spoken language in Ethiopia and the official working language of the Federal Government of Ethiopia, where it has about 30 million native and non-native speakers. The translation of Ge'ez words to Amharic will be of paramount importance in order to enable Amharic user to easily access the invaluable indigenous knowledge decoded in Ge'ez language.

1.4. Beneficiaries of the Project

The Significance of our project is to translate from the source language to other alternatives and also it is well known that a great part of our country's income comes from tourism. The second importance is reaching under resourced languages; by translating the different valuable publications; for example, from Ge'ez to Amharic it is possible to address information need of Amharic language speakers. The third importance is it solves language barriers between individuals to read and understand different publications and also it is absolutely necessary to achieve effective communication between different cultures. In the process of spreading new information, translation is something that can change history of the country.

1.5. Limitations of the Project

- ✔ This Application doesn't support searching words by voice.

- ✔ It may not provide accurate translations for all inputs, as it is relies on machine translation algorithm and limited number of (about 3600 bi-directional dataset) datasets.

- ✔ It only translates 2 languages.

1.6. Scope of the Project

This project is for users who are speaking/use Amharic and Ge'ez languages. And it contains the following tasks:-

- ✔ It translates words as well as sentences between the two languages.

- ✔ The system allows the user to view about us information.

- ✔ The system allows the users view about Ge'ez language and learn about the language.

- ✔ The system shows the setting of the application.

1.7. Methodology

The methodology is the analysis of the principles or procedures of inquiry in a particular field of study. When we relate it with our project it is the system on how we will perform information gathering, analyze and design, implement, test and evaluate it using different ways.

1.7.1 Requirement Gathering Methods

We have chosen the following data collection methods

✔ **Reading:** In order to obtain the required amount of parallel data, a Holy Bible Geez-Amharic translation and some other religious books (Wedase Mariam and Arganon) are used.

✔ **Internet**: It is our major source to gather information for our project. We also have been watching tutorials online about how the flutter based translator works and how to make our application.

✔ **Document analysis**: We will go through different documents that are related with our project.

1.7.2 Analysis and Design Methodology

In this project the team will use object-oriented system development methodology (OOSD). This has two phases.

✔ **Object Oriented Analysis (OOA) -** during this phase the team used to model the function of the system (use case modeling), find and identify the business objects, organize the objects and identify the relationship between them and finally model the behavior of the objects.

✔ **Object Oriented Design (OOD) -** during this phase the team uses enterprise architect software to refine the use case model, and to reflect the implantation environment, model object interactions and behavior that support the use case scenario, and finally update object model to reflect the implementation environment.

1.7.3 Implementation Methodology

Implementation is coding of all functions specified by requirement analysis and design. We have chosen object-oriented programming as our coding approach, because: -

✔ It makes it possible to extend the behavior of objects through both composition and inheritance, allowing objects to take on new life and usefulness in new settings.

✔ It is often the most natural programming approach as it models the real world.

To perform this, we will use client AI model and Dataset architecture. The AI model and Dataset provides service to the client/user and the client requests service from the AI model or Dataset. The application doesn't need any internet, and servers like database and web.

1.8. Feasibility Study

A feasibility study is an assessment of the practicality of a project or system. A feasibility study aims to objectively and rationally uncover the strengths and weaknesses of an existing business or proposed venture, opportunities and threats present in the natural environment, the resources required to carry through, and ultimately the prospects for success. In our project management used to define three areas of feasibility that determine whether a project should run or not. These characteristics are technical, time and Economical feasibility.

1.8.1 Technical Feasibility

✔ **From Developer's Perspective -** Technical feasibility involves the technical resources that are needed to develop, purchase, install, or operate the system. The technical requirements are then compared to the technical capability of the organization, in this case, our team will be considered. We expect the proposed system to be technically feasible because our team has the ability to develop this system since we have acquired the required methodologies and tools. The system is going to be developed using Object Oriented System Development technique. And the team has knowledge about flutter and Enterprise architect to design the system. Besides, the group members have enough capability to develop the project. So, the system is technically feasible.

✔ **From User's Perspective -** The system we are going to develop is also technically feasible from the user's perspective because it is going to be simple to use, it will have user friendly "user interface" and it will work effectively on any web browser and device (PC or smart phones).

1.8.2 Economic Feasibility

The degree to which the economic advantages of something to be made, done, or achieved are greater than the economic costs. We described the economic feasibility of our system as follows:

✔ **Cost -** The required cost to develop the Application, to configure it on the target system, to train user and to purchase the required hardware and software is feasible. Because our team will use free software to develop the system. The organization will not cost anything else except for one desktop. Fortunately, most of the tools that are needed are already available in the organization.

✔ **Benefit -** Our system has an advantage when it reaches its full functionality level. These benefits are:

> ➢ Availability of accurate Translation

> ➢ Increased user satisfaction

> ➢ Easy to use

> ➢ Increase knowledge about Ge'ez

To generalize, from the above statements we can conclude the proposed project is economically feasible considering the cost and the benefits.

1.8.3 Time Feasibility

A time feasibility study will take into account the period in which the project is going to take up to its completion. A project will fail if it takes too long to be completed before it is useful. Typically this means estimating how long the system will take to develop, and if it can be completed in a given time period using some methods like payback period. We develop simple language translator since by considering the time given. So, the system is time feasible.

1.9.   Organization of the Project

Here is the comprehensive and brief summary about our overall project. That helps the user to gather the info about our work easily and quickly. In chapter one, the document covers the background of the project, the statement of problems, the general and specific objectives of the system, significance and the limitation of the project , scope of the project ,the requirements gathering and implementation methodology and the feasibility of the project is included. In chapter two the existing system features and problems, the solutions of the proposed system, the functional requirements, system use case modeling and non-functional requirements are described. In the last chapter system design and detailed design, user interface design, Access control and security of the project are described.

2.   Chapter Two: System Features

2.1.   The Existing System

This part of the document will be devoted to understanding how the existing system performs its task and describe the proposed system in detail. In order to fully understand the existing system, we have started using unstructured interview, document analysis, and reading: In order to obtain the required amount of parallel data, a Holy Bible Ge'ez-Amharic and some other religious books (Wedase Mariam and Arganon) are used and we have started and developed a good working relationship with the stuff of Ge'ez language department.

The existing system functions semi –automated different words with meanings in the paper based dictionary and multi-lingual Translation (Above three languages) like Google translator.

- ✔ Users: student and others search the words meaning from the Dictionary, Google translator

- ✔ Dictionary: book in the paper based dictionary

- ✔ Multi-lingual Translation (Above three languages) like Google translator

- ✔ Free limited API from AI pre-trained models like Google translator, Microsoft and Yandex.

2.2. Proposed System

Flutter framework, which let's to develop android, iOS, Linux, Windows, MacOS and Web compatible, application will be developed using dart by using in combination of static dictionary dataset and AI trained model which mainly focuses on translating words to the target language.

This Bi-lingual translator is mobile world's platforms compatible application, designed to help individuals for word translation to users of the translator. It provides a user-friendly and attractive user interface with its setting up of the application in order to be compatible with users.

2.3. Requirement Analysis

Requirements analysis, also called requirements engineering, is the process of determining user expectations for a new or modified product. In this section we are going to describe the functional and nonfunctional requirements of the proposed system, designing the use case with its description, State chart diagram, designing the sequence diagram and activity diagram for the use case, User Interface prototype, Analysis Class Model then Logic model finally, we are going to design the interface of the system.

2.3.1 Functional Requirement

Functional requirements are product features or functions that developers must implement to enable users to accomplish their tasks. So, it's important to make them clear both for the development team and the stakeholders. Generally, functional requirements describe system behavior under specific conditions. The following are the functional requirements of the proposed system:

- ✔ The system should have setting of the application.

✔ The system should allow the user learn about Ge'ez language.

✔ The system should allow the user view about us page.

✔ The system should allow the users share application.

✔ The system should allow the user translate words as well as sentences.

Table 2: FR Description of the Application

ID	Title	Description	Priority
FR1	The system should allow the user to translate words as well as sentences.	When the user needs to translate the word as well as sentences then the system translates that they find.	High
FR2	The system should allow the users share application	When the user needs to share the application then the system allows sharing the app to different social media.	Medium
FR3	The system should allow the user learn about Ge'ez language.	When the user needs to learn about the Ge'ez language then the system display history of Ge'ez, Ge'ez alphabets, Ge'ez number to the users.	High
FR4	The system should have settings of the application.	The system has settings to allow the user to set of the application.	High
FR5	The system should allow the user view about us page.	When the user needs to know about the developer then the system should allow the user view about us page.	Medium

2.3.2 System Use case

A use case is a methodology used in system analysis to identify, clarify and organize system requirements. The use case is made up of a set of possible sequences of interactions between systems and users in a particular environment and related to a particular goal.

(a) Use case Diagram

Use-case diagrams describe the high-level functions and scope of a system. These diagrams also identify the interactions between the system and its actors. The use cases and actors in use-case diagrams describe what the system does and how the actors use it, but not how the system operates internally.
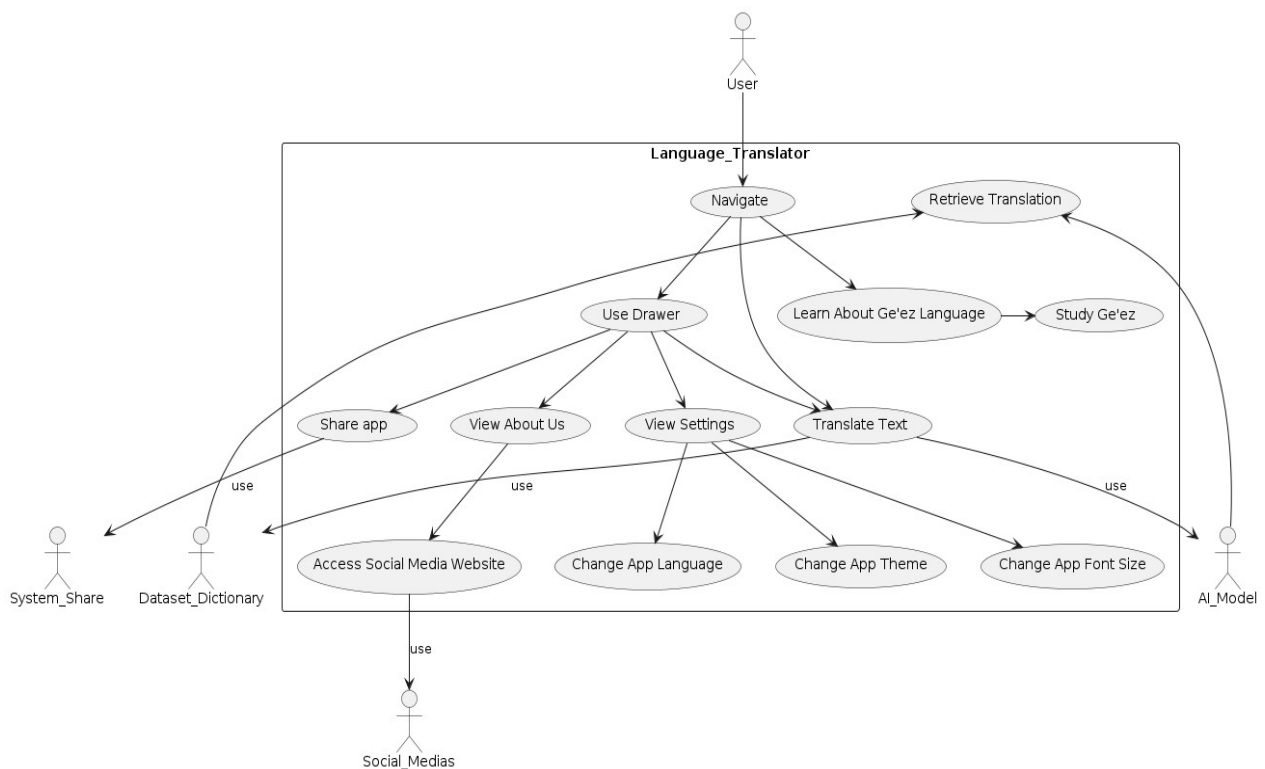
Figure 1: Use case diagram of the proposed system

(b) Use case Documentation

This section should include a use case documentation by showing use case number, name, actor, description, pre-condition, post condition, priority, basic course of action and alternate course of actions. In the basic course of action and alternate course of action, you have to indicate the user interface and the business rule if needed.

Table 3: Use case documentation of Translate Text use case

1	**Use Case Name**	Translate Text
2	**Actors**	User, AI_Model, Dataset_Dictionaries
3	**Summary**	The user navigates to the translation feature, selects the source and target languages, types or pastes the text to be translated. The translation is retrieved from both the AI_Model and the Dataset_Dictionary, and the user can view the translated text.
4	**Preconditions**	The user must have an internet connection and have access to the translation feature.
5	**Basic Flow**	the user navigates to the translation feature. The user selects the source and target languages. The use types or pastes the text to be translated. The translation is retrieved from both the AI_Model and Dataset_Dictionary →

		First search from Dataset → if not found → AI_Model. The user views and may copies the translated text.
6	**Alternate Flow**	If there is an error retrieving the translation, an error message will be displayed to the user. The user can get this page by first navigate to drawer and then tap on Get Started button.
7	**Post Conditions**	The user has viewed the translated text.

Table 4: Use case documentation of Know About Ge'ez use case

1	**Use Case Name**	Study Ge'ez
2	**Actors**	User
3	**Summary**	Users of a platform should open the know Ge'ez menu Item to know the history of Ge'ez,Ge'ez number ,Ge'ez ,alphabet and so on.
4	**Preconditions**	Navigate to the know Ge'ez navigation bar
5	**Basic Flow**	The application displays the page with drop down card title that should be clicked to view the content of them. The user then selects the his/her favorite to know about Ge'ez.
6	**Alternate Flow**	---
7	**Post Conditions**	Displays the text that teaches about basic Ge'ez language

Table 5: Use case documentation of Settings up

1	**Use Case Name**	View Settings
2	**Actors**	User
3	**Summary**	The system allows setting the application that the user wants to set by pressing setting app drop down item in the navigation bar.
4	**Preconditions**	Navigate to the drawer of the application
5	**Basic Flow**	Select the favorite(language, theme and font size) to change the settings of the application.
6	**Alternate Flow**	---
7	**Post Conditions**	Changed with the respected change of the setting of the application

Table 6: Use case documentation of About us page

1	**Use Case Name**	View About Us
2	**Actors**	User, Social Medias
3	**Summary**	The users to view the developer of the application they can be view by pressing the drop down item in the navigation bar.
4	**Preconditions**	The user should first click the drawer after the application opened.
5	**Basic Flow**	Tap the About Us button, it will redirect to About us page. Then if the user wants to view/contact the developers, user should tap his/her favorite contact social media.
6	**Alternate Flow**	---
7	**Post Conditions**	The Application should redirect to the social media of the favorite of the user.

Table 7: Use case documentation of About us page

1	**Use Case Name**	Share app
2	**Actors**	User, System Share
3	**Summary**	Users to share the application to different social media such as telegram ,email, twitter as well others app like Xender, Play store.
4	**Preconditions**	Navigate to the drawer after the application opened.
5	**Basic Flow**	Tap the Share App button, it will redirect the popup page of different types of the system application that can share the application.
6	**Alternate Flow**	---
7	**Post Conditions**	It will share to the favorite of the user to other users.

2.3.3 Business Rule Documentation

Business Rule Documentation (BRD) - This comprises a list of attributes that should be captured when eliciting and documenting business rules. These attributes provide further context to how business rules work and are important for automation.

Table 8: Business Rule documentation

Rule	Name	Description
[BR-1]	Service Access	Users ask valid request to get access service

[BR-2]	Service Provision	The system must provide appropriate responses for the users request without any unreliability.

2.3.4 User Interface Prototype

User interface requirements should be gathered with a prototype approach here and aligned with system use case documentation that shows the user interaction with the system. It should be labeled and referenced in the use case documentation.

- ✔ The user firstly interact with translate page as soon as the application launched.

- ✔ The user can navigate two of the bottom navigation bar namely translate page that one has seen after the application launched and the one that has a learning page of Ge'ez language.

- ✔ After this, the user can navigate throw the drawer which found on the top left corner of the application with three horizontal line.

- ✔ At drawer page, there are four options to choose,

  - ➢ Get Started – navigate to the translate page.

  - ➢ Settings – navigate to the settings page

  - ➢ About Us – navigate to about us page and

  - ➢ Share app – that uses the System share apps for sharing the application for other users.

- ✔ The Get Started page gets the user to the home/translate page that is found after the application is launched.

- ✔ At settings page, the user can change the application language, the application theme and the font size of the application's text.

- ✔ At about us page, a user can access the developer's contact detail with different social media link.

2.3.5 Activity Diagram

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system. Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system. The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent.

The purpose of activity diagrams is like the other four diagrams. It captures the dynamic behavior of the system. Other four diagrams are used to show the message flow from one object to another but the activity diagram is used to show message flow from one activity to another .It does not

show any message flow from one activity to another. Activity diagram is sometimes considered as the flowchart. Although the diagrams look like a flowchart, they are not.
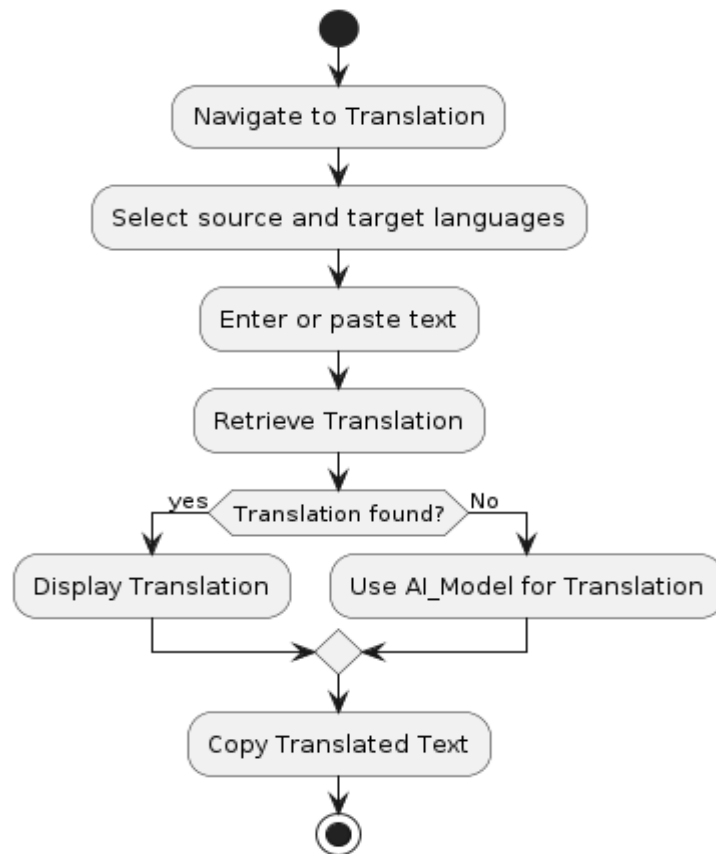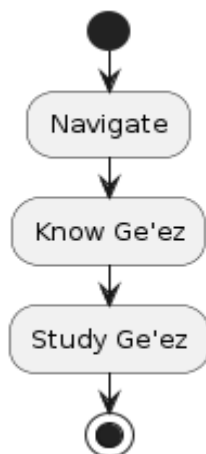

Figure 2: Translate page AD
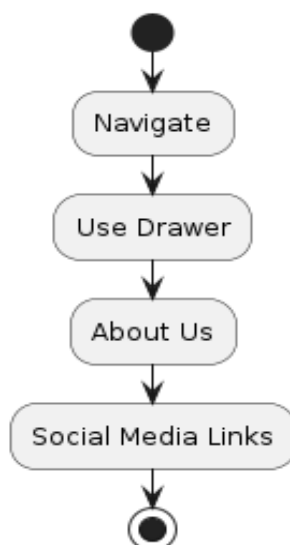

Figure 3: Learn Ge'ez AD

Figure 4: About Us AD
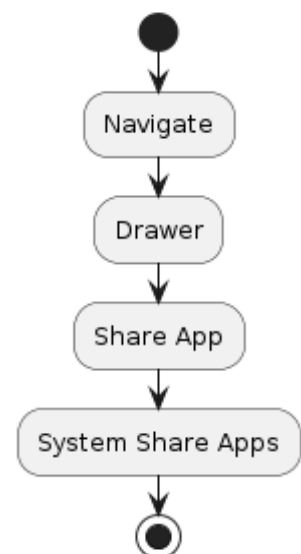
Figure 5: Share App AD

Figure 6: Settings Page AD

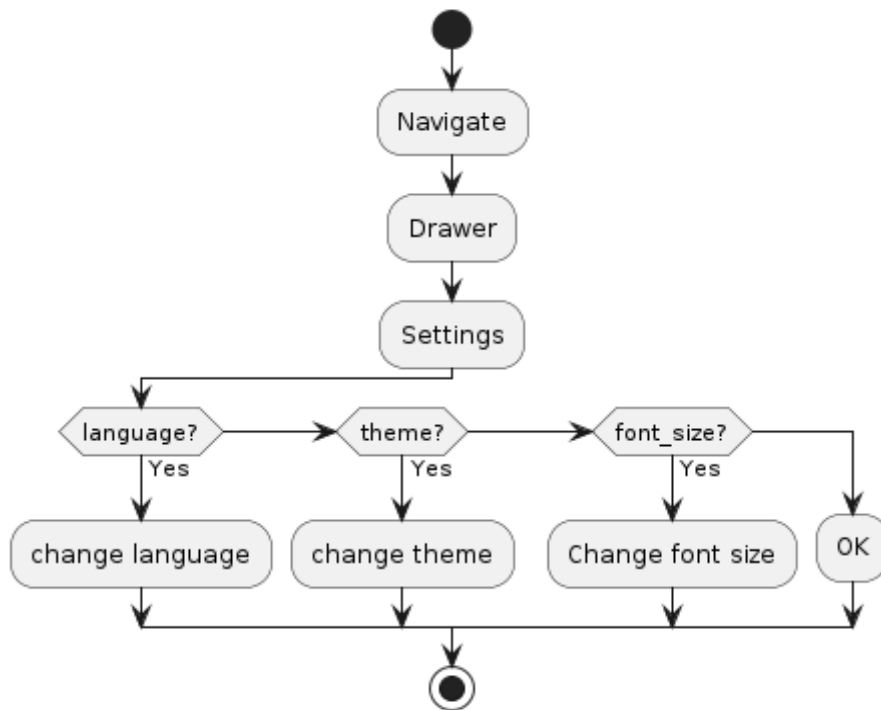### 2.3.6 Sequence Diagram

Sequence diagrams should be drawn for each use case to show how different objects interact with each other to achieve the functionality of the use case. Show how objects operate with one another and in what order (chronological) to respond to the actors.
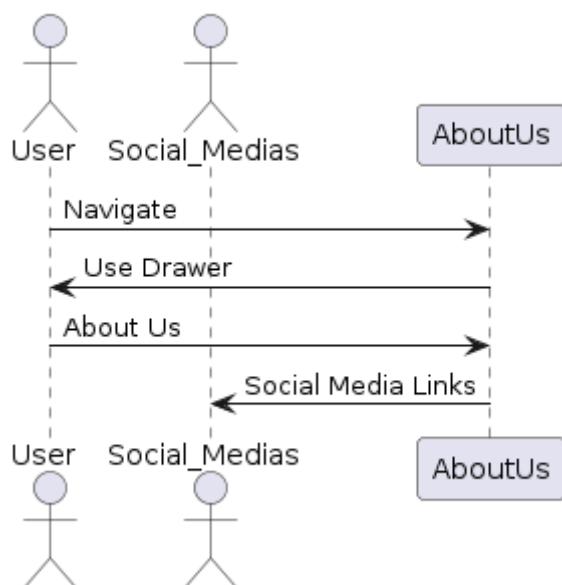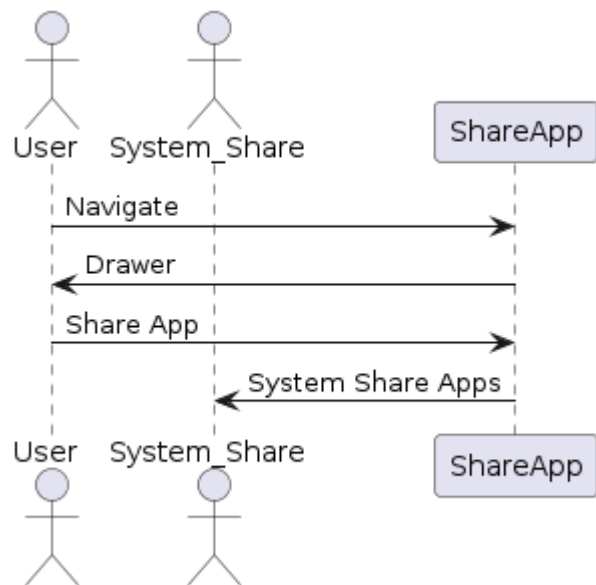

Figure 8: About Us AD
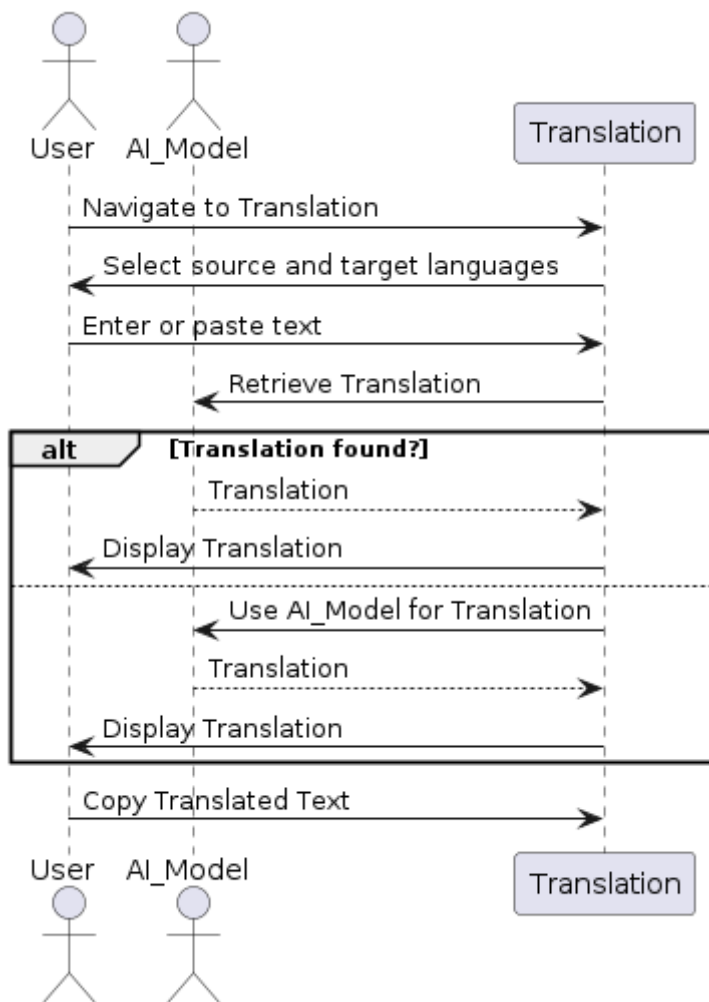

Figure 7: Share App SD

Figure 9: Settings SD

Figure 10: Translate Text Page SD
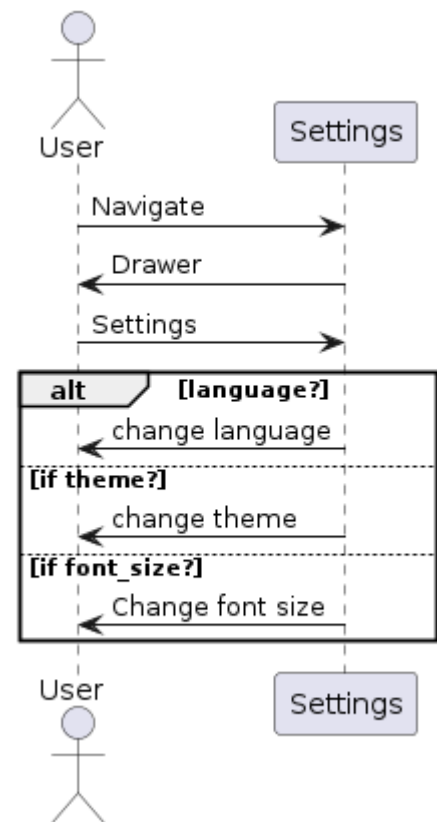
2.3.7 Analysis Class Model

Model Classes with their attributes, methods, relationship, multiplicity and Role
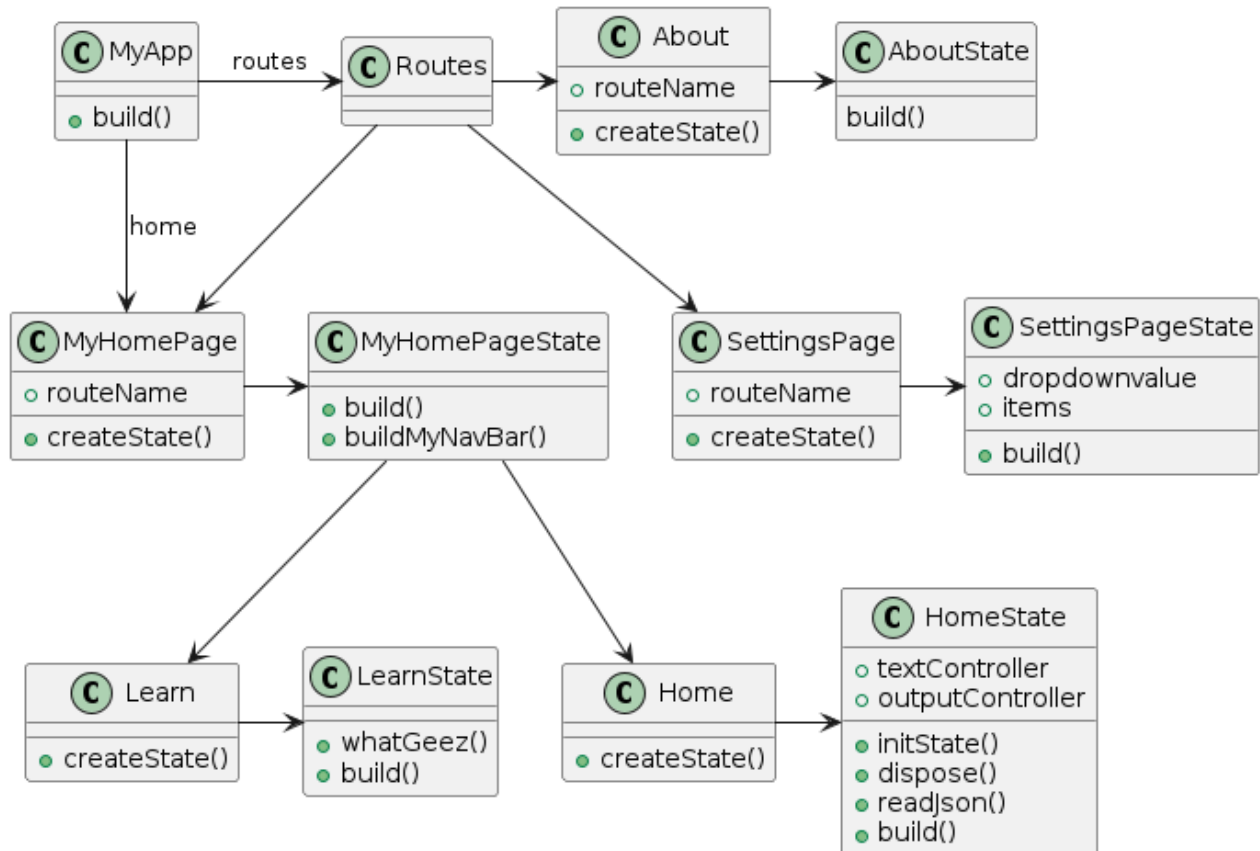
Figure 11: Class Analysis Diagram

2.3.8 Logic Model

It is an informal way of programming description that does not require any strict programming language syntax or underlying technology considerations. It is used for creating an outline or a rough draft of a program. Pseudo code summarizes a program's flow, but excludes underlying details. System designers write pseudo code to ensure that programmers understand a software project's requirements and align code accordingly.

Pseudo code is not an actual programming language. So it cannot be compiled into an executable program. It uses short terms or simple English language syntax to write code for programs before it is actually converted into a specific programming language. This is done to identify top level flow errors, and understand the programming data flows that the final program is going to use.

Table 9: Logic Model description of functionality of the App

1	The user opens the application and selects the source and target languages.

2	The user inputs the text they want to translate.
3	The application sends the input text first to Dataset, if not found sends to the AI model, which uses machine learning algorithms to analyze the text and determine the most likely translation.
4	The AI model returns the translation to the application.
5	The application displays the translation to the user.
6	If the user is not satisfied with the translation, the user can ask for alternative translations.
7	The application sends the input text to the AI model which return the alternative translations.
8	The application displays the alternative translations to the user.
9	The user can select the best translation or ask for another alternative.
10	The application stores the user's selection to improve the AI model in the future.

Pseudo code for Know Ge'ez Page

1) Begin

2) Open Homepage

3) Navigate to Learn Ge'ez button

4) Displays Information about Ge'ez language

5) Stop

Pseudo code for Settings

1) Begin

2) Navigate to drawer of the app after the application is opened.

3) Tap on **Settings** button

4) Choose the preference to change the application setting

5) Stop

Pseudo code for About Us

1) Begin

2) Navigate to Drawer after the application opened.

3) Tap on **About Us** button

4) Use the button link to contact/view the profile of developers.

5) Stop

Pseudo code for Share App

1) Begin

2) Navigate to drawer of the application

3) Tap on **Share App** button

4) Overlay pop up window that contains different application that can share a link will appear

5) Select the application that user want share the app to other user.

6) Send to the other users

7) Stop

2.4. Non-Functional Requirement

Non-Functional Requirement (NFR) specifies the quality attribute of a software system. They judge the software system based on Responsiveness, Usability, Security, accuracy and other non-functional standards that are critical to the success of the software system. The non-functional requirements of this system include the following

- ✔ **Reliability:** The system consistently performs its intended function. Example the important functions are validated. So the system is reliable.

- ✔ **Efficiency**: The system is efficient based on the concept of resource Consumption. Example it only needs a computer that has a browser to access the system.

- ✔ **Robustness** (Error handling requirement): The system have error handling mechanisms that is, as errors occur it will not stop functioning rather provide error manages and back to the previous page to give chance to reenter data and process the task by beyond the error.

- ✔ **Resources:** The system is compatible with specified hardware and software environment

- ✔ **Platform:** our system supports any operating system and all browsers.

- ✔ **Performance requirement:** Response time**: -** depending on the strength of available network the system should be response in short period of time.

- ✔ **Availability**: The system is available 24 hours /7 days to everyone.

- ✔ **Correctness**: The application should never allow unauthorized users to delete or insert evaluation data from the database.

✔ **Accuracy:** The level of accuracy in the proposed system will be better due to the reduction of error. The system is implemented in a way that gives correct output for the users when they give correct input.

✔ **Usability:** Our system user interface will be designed in a simple and very attractive manner in order to make the usability of the system comfortable.

2.5.   System Requirement

We will use different types of software and hardware tools to build this multi-platform compatible application. We have showed all requirement as the below table.

Table 10: System Requirement(Software and Hardware)

No.	Software Requirement	Hardware Requirement
1	Android Studio Code, Visual Code	Computer with 4 – 8 GB installed RAM, 64-bit system type
2	GitHub/Git	Flash Disk
3	PlantUML/Graphiz, Edraw max, App Diagrams, Figma	
4	Colab Research for AI Model Training	

2.6.   Key Abstraction with CRC Analysis

A Class Responsibility Collaborator (CRC) model is a collection of standard index cards that have been divided into three sections. A class represents a collection of similar objects, a responsibility is something that a class knows or does, and a collaborator is another class that a class interacts with to fulfill its responsibilities.

2.6.1 Persistent Modeling

A persistent model is a way of storing data in a long-term, durable manner, such as in a database. In a class diagram, a persistent model would typically be represented by a separate class or set of classes that handle the database operations, such as connecting to the database, querying data, and updating data.

We have not used any database in this application. We have prepared a dataset in the form of a dictionary to be used as a dataset, and in addition, we train this dataset with a transformer AI model to solve definitions that the dictionary cannot solve. Because we don't have a database, we didn't make any kind of persistent model here either; For we have not used it.

2.6.2 Conceptual Modeling – Class Diagram

Class models are the mainstay of the OO analysis and design. Class models show the classes of the system, their interrelationships (including inheritance, aggregation, and association) and the operations and attributes of the classes. The conceptual models are used to depict your detailed understanding of the problem space and solution for your system. The easiest way to begin conceptual modeling is to convert the CRC (as a base) directly to UML class diagram. While a CRC model provides an excellent overview of a system, it doesn't provide the details needed to actually build it.
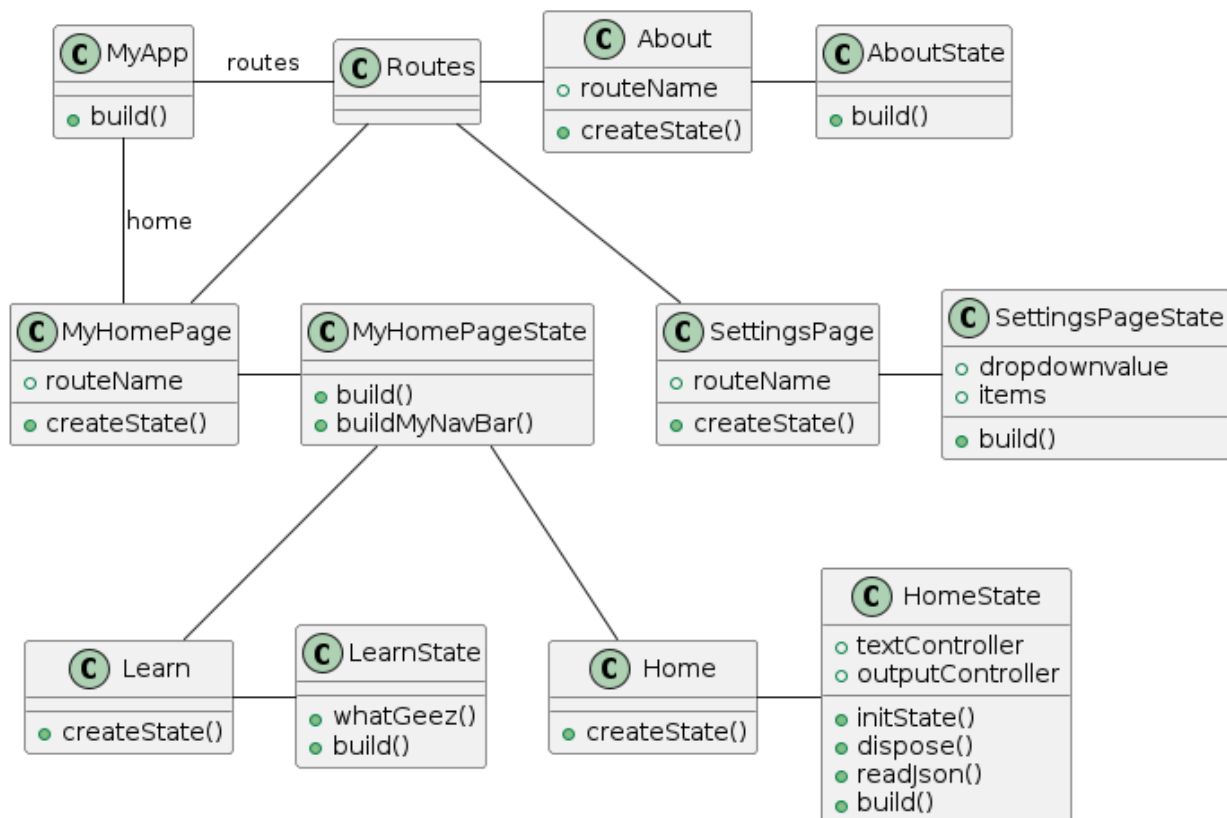
Figure 12: CRC class model of the Application

2.6.3 Identify Change Case

Change cases are used to describe potential modifications requirements to the system.

- ✔ **Comments**: - When crucial comments are raised from the advisor, friends, and examiners that should be included and excluded the developing team assesses the project again.

- ✔ **Lack of time**: - If the given time to develop many system activity is not enough the developing team also restructured the content of the project.

3.   Chapter Three: System Design

3.1. Introduction

Systems design is the process of defining elements of a system like modules, architecture, components and their interfaces and data for a system based on the specified requirements. It is the process of defining, developing and designing systems which satisfies the specific needs and requirements of a business or organization. The system design activity starts after the system requirements analysis has been completed. It is based on an understanding of the model the Application built on system design also focuses on decomposing the system into manageable parts.

Design is important part of the system development process. This is actually where the requirements mentioned before come to the ground to the finished software product. It is also useful in assessing risks that can be encountered. Most importantly when we talk about designing it is the place where we deal about the quality of the system so assessing quality will not be difficult when there is a design of the system.

The objective of design is to model a system within a qualified way. Implementing of the high quality system depends on the nature of the design .In the process of going from the analysis model to building the system design, design goals must be identified. These will identify qualities that the system must focus on, so that design decisions can be based on a specific set of criteria. The overall design goals should be decided upon early in the system design process, since they will contain priorities that the rest of the system design must consider it .The goal of the system design is to manage complexity by decomposing the system into manageable pieces. Some designs goals will come right from the non-functional requirements others must be elicited from the client. Some are listed below.

- ✔ Performance
- ✔ Dependability
- ✔ Maintenance
- ✔ End user

Performance:-
The system should respond fast with high throughput, i.e. it should perform the task quickly as possible. The system performs its task within a user acceptable time and space. This includes the following:-

- ✔ Response time: - depending on the strength of available network the system should be response in short period of time.

Dependability: - our system includes the following dependability criteria:

- ✔ **Reliability**: our system should be reliable.
- ✔ **Fault Tolerance**: - our system should be fault tolerant to loss of connectivity with the service.
- ✔ **Availability**: - as long as there is an internet connection the system will be available 24 hours a day

Maintainability:-To be maintainable the system should meet the following maintenance criteria:

> ➢ **Modifiability:** our system should be modifiable for further modification and enhancement of the system.
> ➢ **Portability: -** the system is developed to be viewed and retrieved from any web browser regardless of their version and platform it resides in it.
> ➢ **Extensibility: -** if it is needed to add new functionality to the system, this must be achieved by only making a separate page and integrate this page with the existing system.

End User Criteria:-

The system should provide user-friendly graphical user interface that eases the interaction of the user with the system so that the user can access it without any difficult.

3.2. Architectural Design

The architecture plays a main role in modularizing the different components of a system. The layered architectural style focuses on the grouping of related functionality within an application into distinct layers that are stacked vertically on top of each other.

3.2.1 Component Modeling

Systems may be built from components in component based architecture. Component diagram shows how objects (classes) in the system are grouped together and form components. The components interact with each other either in giving service to other components or requesting service from other component.
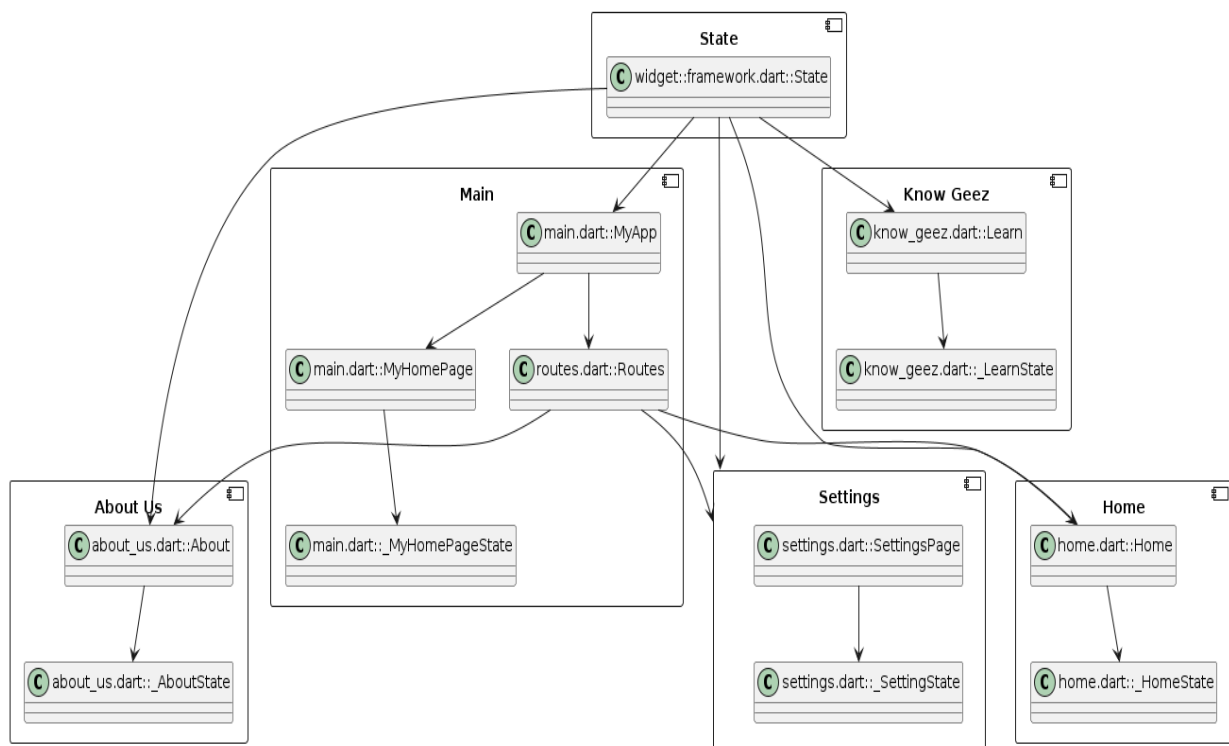

Figure 13: Component Diagram of the Application

3.3. Deployment Modeling

Deployment diagram show how the system is deployed on computers. In other words, it shows which component of the software is installed on which machine and how they communicate with each other if they are on different machines. Indicate where each component will be located, on what servers, machines or hardware. The deployment model should clearly show the physical communication links between hardware and software items.

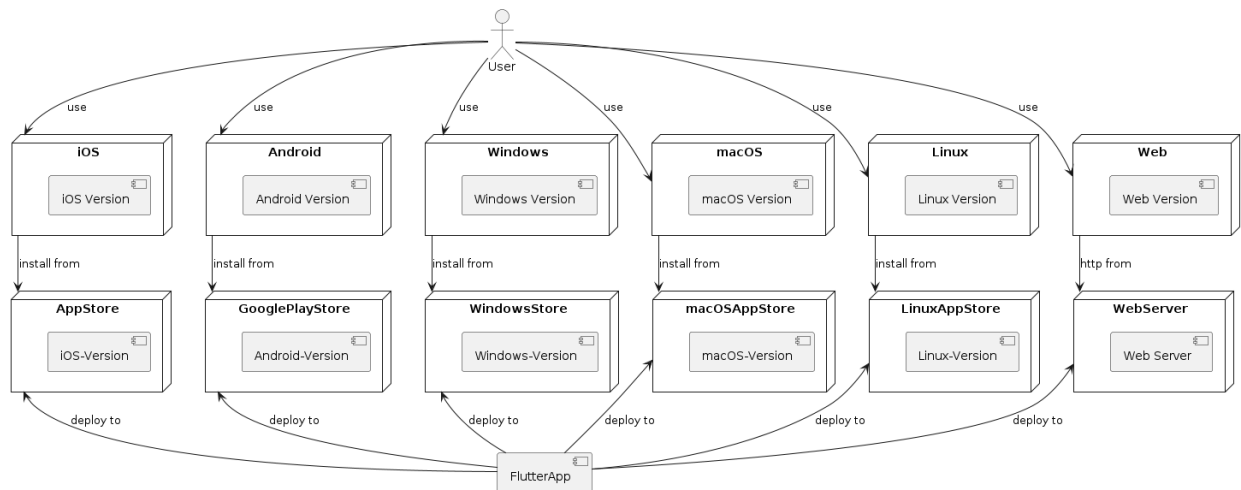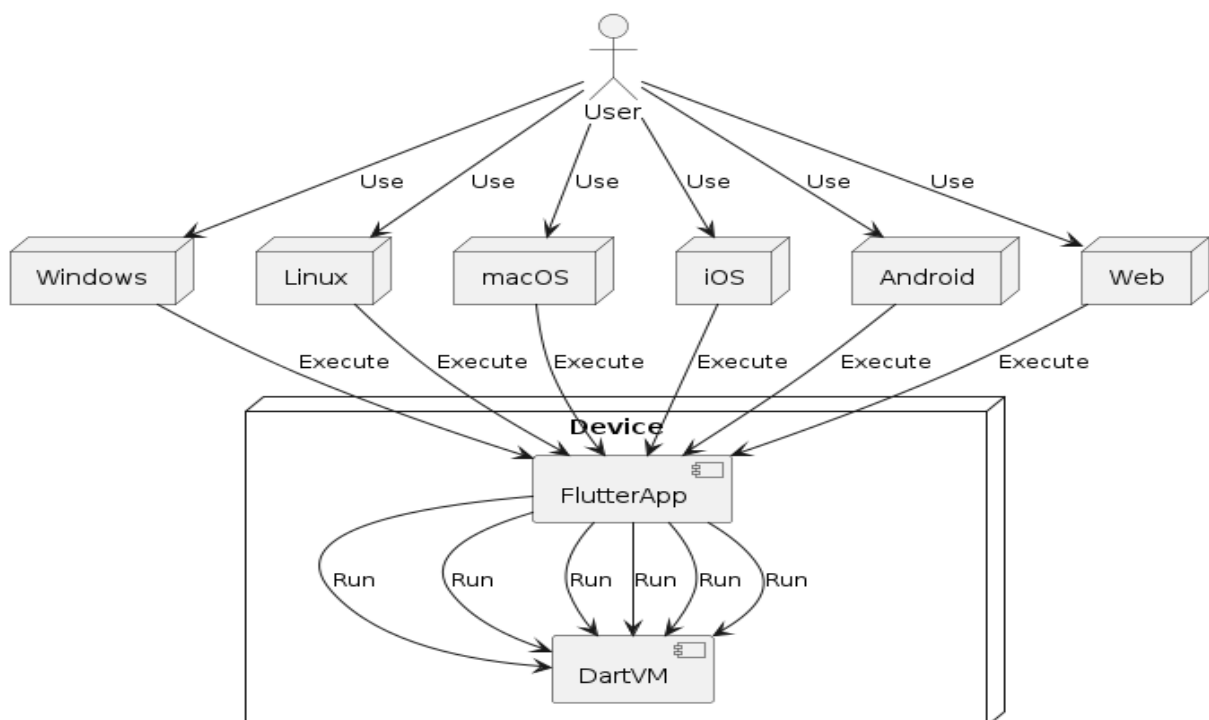*Figure 14: Deployment Diagram of the Application*

Figure 15: Usage and its process of Execution of the Application

3.4. Detail Design

3.4.1 Design Class Model

The Design Class Model is classes with their attribute data types, methods with their return types and arguments and their data types, relationships, Access visibility, multiplicity and role.
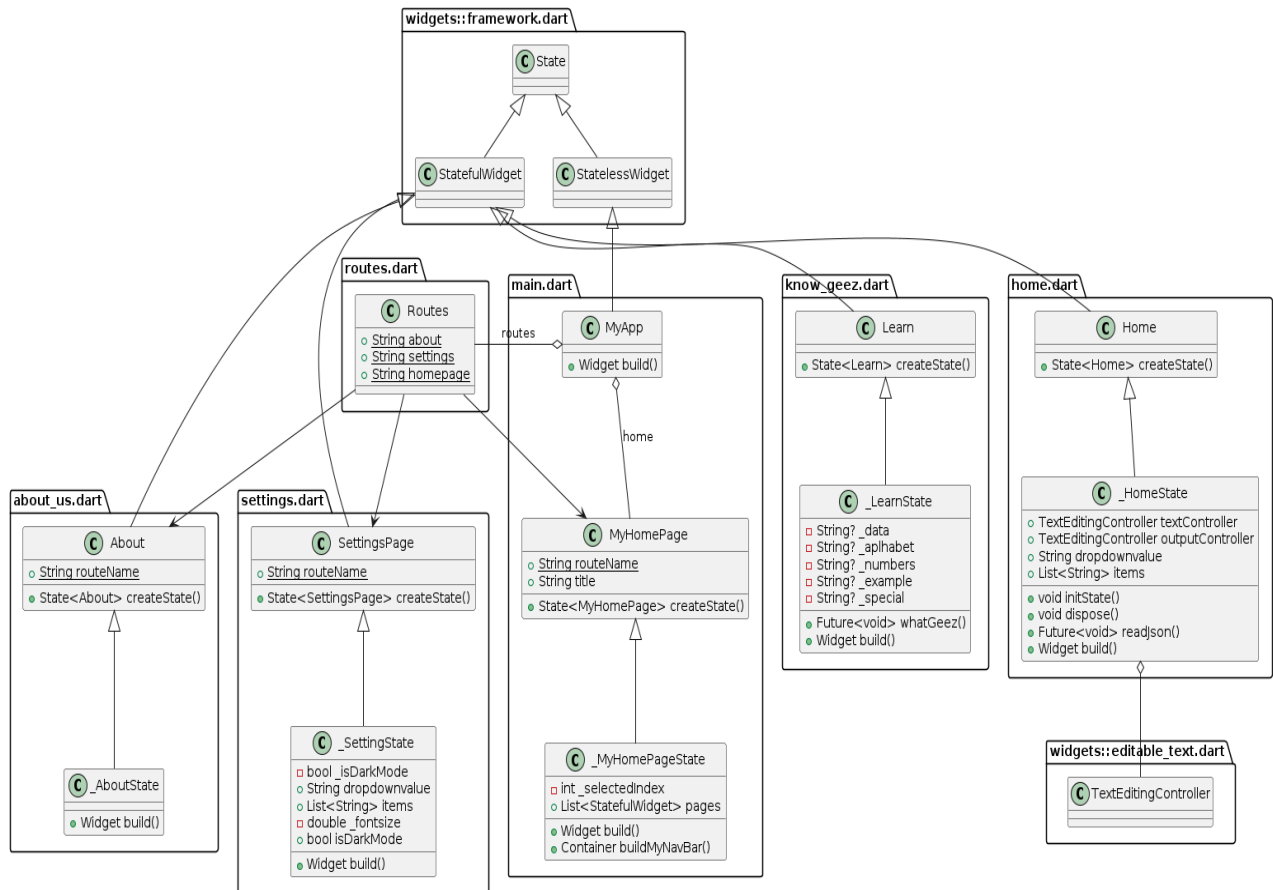
Figure 16: Class Diagram of the Application

3.5. User Interface Design

Figure 17: Complete UI of the Application

3.6. Access Control and Security

Table 11: Access control and security of the Application

ID	Responsibilities	Actors
		Users
1	Translate Text	√
2	Setting up application like change language, font size and theme	√
3	Learn about Ge'ez Language	√
4	View and Access the Developers of the Application via their Social Media	√
5	Share the app to other users	√

4. Chapter Four: Implementation

```
import 'package:flutter/material.dart';
import 'dart:convert';
import 'package:flutter/services.dart';
import 'package:dropdown_button2/dropdown_button2.dart';
import 'package:fluttertoast/fluttertoast.dart';
import 'classifier.dart';

class Home extends StatefulWidget {
  const Home({Key? key}) : super(key: key);

  @override
  State<Home> createState() => _HomeState();
}

class _HomeState extends State<Home> {
  // the textfield's controllers
  final textController = TextEditingController();
  final outputController = TextEditingController();

  final _classifier = Classifier();

  // Initial Selected Value
  String dropdownvalue = "Ge'ez to Amharic";

  // List of items in our dropdown menu
  var items = ["Ge'ez to Amharic", "Amharic to Ge'ez"];

  @override
  void initState() {
    super.initState();
    textController.addListener(translate);
  }

  @override
  void dispose() {
    // Clean up the controller when the widget is removed from the
widget tree.
    // This also removes the _printLatestValue listener.
    textController.dispose();
    super.dispose();
  }
```

The functionality of Translation

```
// Fetch content from the json file
  Future<void> translate() async {
    final String response =
        await rootBundle.loadString('assets/files/geez_amharic.json');
    Map lang = await json.decode(response);

    lang.forEach((key, value) {
      if (dropdownvalue == "Ge'ez to Amharic") {
        if ('$key' == textController.text) {
          outputController.text = '$value';
        }
        else {
          _translate(textController.text);
        }
      } else {
        if ('$value' == textController.text) {
          outputController.text = '$key';
        }
        else {
          _translate(textController.text);
        }
      }
    });
  }

  void _translate(String inputText) {
    final output = _classifier.classify(inputText);
    setState(() {
      outputController.text = output;
    });
  }
```

```
@override
 Widget build(BuildContext context) {
  return ListView(
   children: [
    Padding(
     padding: const EdgeInsets.all(10.0),
     child: Center(
      child: DropdownButtonHideUnderline(
       child: DropdownButton2(
        // Initial Value
        value: dropdownvalue,
       // Down Arrow Icon
        icon: const Icon(Icons.keyboard_arrow_down),
        // Array list of items
        items: items.map((String items) {
         return DropdownMenuItem<String>(
          value: items,
          child: Text(items),
         );
        }).toList(),
        // After selecting the desired option,it will
        // change button value to selected value
        onChanged: (String? newValue) {
         setState(() {
          dropdownvalue = newValue!;
         });
        },
        iconEnabledColor: Colors.teal,
        buttonWidth: 300,
        buttonPadding: const EdgeInsets.only(left: 15, right: 15),
        buttonDecoration: BoxDecoration(
         borderRadius: BorderRadius.circular(15),
         border: Border.all(
          color: Colors.teal,
         ),
        ),
        itemHeight: 40,
        itemPadding: const EdgeInsets.only(left: 15, right: 15),
        dropdownWidth: 300,
        offset: const Offset(0, -3),
        dropdownDecoration: BoxDecoration(
         borderRadius: BorderRadius.circular(15),
         border: Border.all(
          color: Colors.teal,
        ), ), ), ), ), ),
```

Input Text Field

```
Padding(
    padding: const EdgeInsets.all(10.0),
    child: TextField(
      obscureText: false,
      maxLines: 18,
      cursorColor: Colors.teal,
      decoration: InputDecoration(
        suffixIcon: IconButton(
          onPressed: () async {
            ClipboardData? data = await Clipboard.getData("text/plain");
            String? text = data?.text;
            textController.text = text!;

            // Show the toast message
            Fluttertoast.showToast(
              msg: "Text pasted from the clipboard",
              toastLength: Toast.LENGTH_SHORT,
              gravity: ToastGravity.BOTTOM,
              timeInSecForIosWeb: 2,
              fontSize: 16,
              textColor: Colors.blue,
              backgroundColor: Colors.white,
            );
          },
          icon: const Icon(
            Icons.paste,
            color: Colors.blue,
          ),
        ),
        border: const OutlineInputBorder(),
        hintText: "Enter word(s) or sentence(s) for translation...",
        hintStyle: const TextStyle(color: Colors.teal)),
      controller: textController,
    ),
  ),
```

Output Text Field

```
Padding(
    padding: const EdgeInsets.all(10.0),
    child: TextField(
      obscureText: false,
      maxLines: 18,
      readOnly: true,
      decoration: InputDecoration(
        suffixIcon: IconButton(
          icon: const Icon(
            Icons.copy_all,
            color: Colors.blue,
          ),
          onPressed: () {
            Clipboard.setData(
                ClipboardData(text: outputController.text));

            // Show the toast message
            Fluttertoast.showToast(
              msg: "Text copied to the clipboard",
              toastLength: Toast.LENGTH_SHORT,
              gravity: ToastGravity.BOTTOM,
              timeInSecForIosWeb: 2,
              fontSize: 16,
              textColor: Colors.blue,
              backgroundColor: Colors.white,
            );
          },
        ),
        border: const OutlineInputBorder(),
        hintText: "Translated items will appear here",
        hintStyle: const TextStyle(color: Colors.teal)),
      controller: outputController,
    ),
  ),
 ],
);
}
}
```

For the complete code of the project see the GitHub repository through <u>Bidirectional Language Translator</u>

5.   Chapter Five: Testing and Evaluation

Introduction:

The bidirectional language translation project aims to provide users with a convenient and accurate way to translate text between Ge'ez and Amharic languages. The project uses a combination of dictionary lookup and machine learning inference to achieve this goal. In this part of document, we will discuss the testing and evaluation process for this project.

Testing:

The testing phase of the project is divided into two parts: unit testing and integration testing.

Unit Testing: Unit testing involves testing individual components of the project to ensure that they function correctly. The following components will be tested:

- ✔ **Dictionary lookup function**: The dictionary lookup function should correctly return the corresponding translation for a given input text. We will test this function with a variety of inputs to ensure that it returns the correct output in all cases.

- ✔ **Machine learning inference function:** The machine learning inference function should correctly translate input text from Ge'ez to Amharic and vice versa. We will test this function with a variety of inputs to ensure that it returns the correct output in all cases.

Integration Testing: Integration testing involves testing the interaction between different components of the project to ensure that they work together correctly. The following interactions will be tested:

- ✔ **Dictionary lookup + machine learning inference**: We will test the integration between the dictionary lookup function and the machine learning inference function. Specifically, we will test cases where the input text is not found in the dictionary and must be passed to the machine learning model for inference.

Evaluation: The evaluation phase of the project involves assessing the accuracy and performance of the translation system.

- ✔ **Accuracy:** To evaluate the accuracy of the translation system, we will use a set of test cases with known translations. We will compare the translations generated by the system with the known translations to calculate the accuracy of the system. We will also perform error analysis to identify common errors made by the system and areas for improvement.
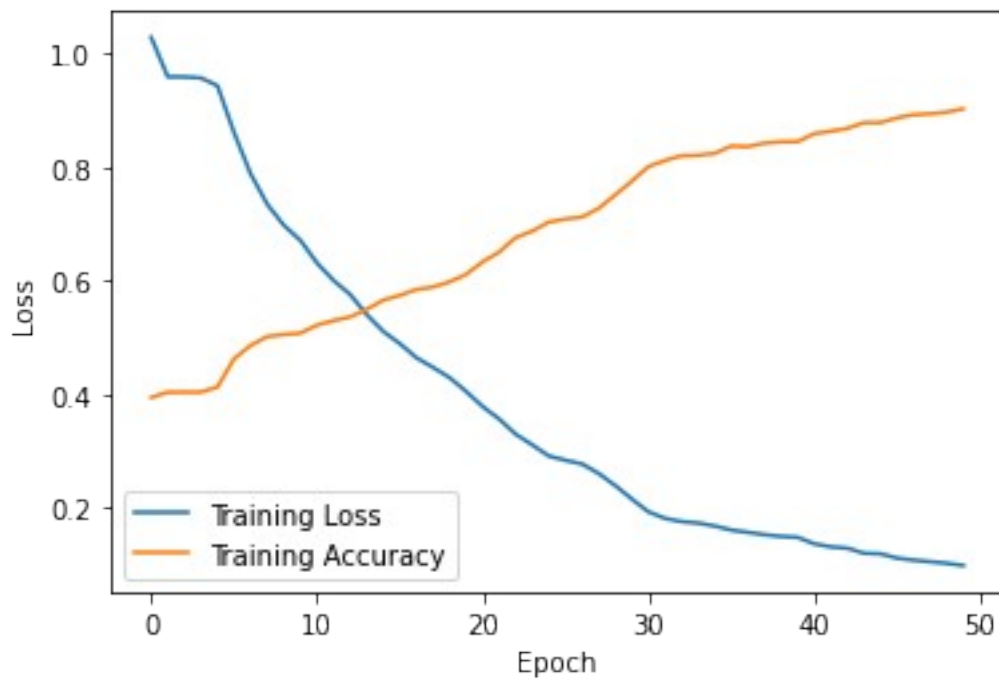
Figure 18: Accuracy and Loss of the AI training Data

✔ **Performance:** To evaluate the performance of the translation system, we will measure the time taken to generate translations for a set of inputs. We will also measure the memory usage of the system during inference to identify areas for optimization.

Conclusion: In this document, we have outlined the testing and evaluation process for the bidirectional language translation project. By thoroughly testing and evaluating the system, we can ensure that it provides accurate and reliable translations to users.

References

- [BiT Project Documentation format of the students](#)
- [Flutter Project Documentation](#)
- [Flutter Package Documentation](#)
- [Dart Programming Documentation](#)
- [Wikipedia](#)
- [Training and Dictionary set of the two language - Dataset](#)

Appendices

- **Activity Diagram(AD)** – is graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.

- **Android** – is a mobile operating system based on a modified version of the Linux kernel and other open-source software, designed primarily for touchscreen mobile devices such as smartphones and tablets.

- **Artificial Intelligence(AI)** – is intelligence → perceiving, synthesizing, and inferring information → demonstrated by machines, as opposed to intelligence displayed by non-human animals and humans.

- **Class Responsibility Collaborator(CRC)** – is cards that are a brainstorming tool used in the design of object-oriented software. They originally proposed by Ward Cunningham and Kent Beck as a teaching tool but are also popular among expert designers and recommended by extreme programming.

- **iOS** – is a mobile operating system created and developed by Apple Inc. exclusively for its hardware. It is the operating system that powers many of the company's mobile devices, including the iPhone.

- **Linux** – is a family of open-source Unix-like operating systems based on the Linux kernel, an operating system kernel first released on September 17, 1991, by Linux Torvalds.

- **MacOS** – is a Unix operating system developed and marketed by Apple Inc. since 2001. It is the primary operating system for Apple's Mac computer.

- **Non-Functional Requirement(NFR)** – is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviors. They are contrasted with functional requirements that define specific behavior or function.

- **Object Oriented(OO)** – is a programming paradigm based on the concept of "objects", which can contain data and code. The data is in the form of fields, and the code is in the form of procedures. A common feature of objects is that procedures are attached to them and can access and modify the object's data fields.

- **Object Oriented Analysis and Design (OOA & OOD)** – is a technical approach for analyzing and designing an application, system, or business by applying object-oriented programming, as well as using visual modeling throughout the software development process to guide stakeholder communication and product quality.

- **Operating System(OS)** – is system software that manages computer hardware, software resources, and provides common services for computer programs.

- **Personal Computer(PC)** - is a multi-purpose microcomputer whose size, capabilities, and price make it feasible for individual use.

- **Sequence Diagram(SD)** - shows process interactions arranged in time sequence in the field of software engineering. It depicts the processes involved and the sequence of messages exchanged between the processes needed to carry out the functionality.

- **User Interface(UI)** - In the industrial design field of human–computer interaction, a user interface is the space where interactions between humans and machines occur.

- **Unified Modeling Language(UML)** - is a general-purpose, developmental modeling language in the field of software engineering that is intended to provide a standard way to visualize the design of a system.

- **Web –** is World Wide Web, commonly known as the Web, is an information system enabling documents and other web resources to be accessed over the Internet. Documents and downloadable media are made available to the network through web servers and can be accessed by programs such as web browsers.

- **Windows -** is a group of several proprietary graphical operating system families developed and marketed by Microsoft. Each family caters to a certain sector of the computing industry. For example, Windows NT for consumers, Windows Server for servers, and Windows IoT for embedded systems.