



# **Wolkite University**

## **College of Computing and Informatics**

### **Department of Software Engineering**

#### **Ethiopia sign language (ESL) learning app**

##### **Team members**

| <b><u>NO</u></b> | <b><u>NAME</u></b>  | <b><u>ID</u></b> |
|------------------|---------------------|------------------|
| 1                | Fekedew Hailemariam | CIR/133/09       |
| 2                | Dawit Alemayehu     | CIR/092/09       |
| 3                | Getaneh Aychew      | CIR/147/09       |
| 4                | Megersa Wakgari     | CIR/194/08       |

Advisor Mr. Frezer G.

Monday, January 27, 2020



# Wolkite University

## College of Computing and Informatics

### Department of Software Engineering

#### ***Project on Ethiopia sign language (ESL) learning app***

SUBMITTED TO DEPARTMENT OF SOFTWARE ENGINEERING IN  
PARTIAL FULFILMENT OF THE REQUIREMENT FOR THE DEGREE OF  
BACHLER SCIENCE IN SOFTWARE ENGINEERING

**BY: -**

**NO**

**NAME**

**ID**

|   |                     |            |
|---|---------------------|------------|
| 1 | Fekedew Hailemariam | CIR/133/09 |
| 2 | Dawit Alemayehu     | CIR/092/09 |
| 3 | Getaneh Aychew      | CIR/147/09 |
| 4 | Megersa Wakgari     | CIR/194/08 |

Advisor Mr. Frezer G.

Monday, January 27, 2020

## DECLARATION

This is to declare that this project work which is done under the supervision of Mr. Frezer G. and having the title Ethiopian sign language learning application is the sole contribution of Fekedew Hailemariam, Dawit Alemayehu, Getaneh Aychew and Megersa Wakgari.

No part of the project work has been reproduced illegally (copy and paste) which can be considered as plagiarism. All referenced parts have been used to argue the idea and have been cited properly. We will be responsible and liable for any consequence if the violation of this declaration is proven.

**Date:** Monday, January 27, 2020

### Group Members: -

| <u>NO</u> | <u>Full name</u>    | <u>Signature</u> |
|-----------|---------------------|------------------|
| 1         | Fekedew Hailemariam | _____            |
| 2         | Dawit Alemayehu     | _____            |
| 3         | Getaneh Aychew      | _____            |
| 4         | Megersa Wakgari     | _____            |

## APPROVAL FORM

This is to confirm that the project report entitled Ethiopian sign language learning application submitted to Wolkite University, College of Computing and Informatics Department of Software Engineering by Fekedew Hailemariam, Dawit Alemayehu, Getaneh Aychew and Megersa Wakgari is approved for submission.

|                             |                  |             |
|-----------------------------|------------------|-------------|
| <b>Advisor name</b>         | <b>Signature</b> | <b>Date</b> |
| -----                       | -----            | -----       |
| <b>Department head name</b> | <b>Signature</b> | <b>Date</b> |
| -----                       | -----            | -----       |
| <b>Examiner 1 name</b>      | <b>Signature</b> | <b>Date</b> |
| -----                       | -----            | -----       |
| <b>Examiner 2 name</b>      | <b>Signature</b> | <b>Date</b> |
| -----                       | -----            | -----       |
| <b>Examiner 3 name</b>      | <b>Signature</b> | <b>Date</b> |
| -----                       | -----            | -----       |

## **ACKNOWLEDGMENT**

First and for most, we would like to thanks the almighty GOD for help us to be alive and to do all of this. Thanks, our family for everything, and we have very special thanks to our advisor Mr. Frezer Guteta for guiding us, correcting our mistakes, constant supervision, encouraging us to do more and providing necessary information regarding the project. We have also so many individuals, groups, instructors, department deans, faculty deans, and students that we want to thank for supporting while working on this project.

## TABLE OF CONTENTS

|  |     |
|--|-----|
| DECLARATION .....                              | i   |
| APPROVAL FORM .....                            | ii  |
| ACKNOWLEDGMENT .....                           | iii |
| TABLE OF CONTENTS .....                        | iv  |
| LIST OF TABLES.....                            | ix  |
| LIST OF FIGURES.....                           | x   |
| ABBREVIATIONS.....                             | xi  |
| CHAPTER ONE .....                              | 1   |
| 1. INTRODUCTION TO THE PROJECT.....            | 1   |
| 1.1. Introduction .....                        | 1   |
| 1.2. Statement of the problem .....            | 1   |
| 1.3. Objective .....                           | 3   |
| 1.3.1. General objective .....                 | 3   |
| 1.3.2. Specific objective .....                | 3   |
| 1.4. Feasibility Analysis .....                | 3   |
| 1.4.1. Operational feasibility.....            | 3   |
| 1.4.2. Technical feasibility.....              | 4   |
| 1.4.3. Economic feasibility .....              | 4   |
| 1.4.4. Schedule feasibility .....              | 4   |
| 1.5. Scope and limitation of the project ..... | 4   |
| 1.6. Significance of the project .....         | 5   |
| 1.6.1. Deaf person .....                       | 5   |
| 1.6.2. Parent of the deaf child.....           | 5   |

|                     |   |    |
|---------------------|---|----|
| 1.6.3.              | Anyone who wants to learn sign language .....           | 5  |
| 1.6.4.              | Schools, instructors, community and country .....       | 6  |
| 1.7.                | The methodology of the project .....                    | 6  |
| 1.7.1.              | Data Collection .....                                   | 7  |
| 1.7.2.              | System Analysis .....                                   | 7  |
| 1.8.                | Development tool .....                                  | 8  |
| 1.8.1.              | Front end .....   | 8  |
| 1.8.2.              | Back end .....  | 9  |
| 1.9.                | Document organization .....                             | 9  |
| CHAPTER TWO .....   |   | 11 |
| 2.                  | DESCRIPTION OF THE EXISTING SYSTEM .....                | 11 |
| 2.1.                | Introduction of Existing System .....                   | 11 |
| 2.2.                | Users of Existing System .....                          | 11 |
| 2.3.                | Major Functions of the Existing System .....            | 12 |
| 2.4.                | Forms and Other Documents of the Existing Systems ..... | 12 |
| 2.5.                | Drawbacks of the Existing System .....                  | 12 |
| 2.6.                | Business Rules of the Existing System .....             | 14 |
| CHAPTER THREE ..... |   | 18 |
| 3.                  | PROPOSED SYSTEM .....                                   | 18 |
| 3.1.                | Functional requirement .....                            | 18 |
| 3.2.                | Non-functional requirement .....                        | 19 |
| 3.2.1.              | User interface and human factors .....                  | 19 |
| 3.2.2.              | Security issues .....                                   | 20 |
| 3.2.3.              | Performance considerations .....                        | 20 |



|                    |   |    |
|--------------------|---|----|
| 3.2.4.             | Error handling and validation.....            | 21 |
| 3.2.5.             | Quality issues .....                          | 21 |
| 3.2.6.             | Physical environment.....                     | 21 |
| 3.2.7.             | Documentation .....                           | 21 |
| CHAPTER FOUR ..... |   | 22 |
| 4.                 | SYSTEM ANALYSIS.....                          | 22 |
| 4.1.               | System model.....                             | 22 |
| 4.1.1.             | Use case model .....                          | 22 |
| 4.2.               | Object model.....                             | 40 |
| 4.2.1.             | Class diagram .....                           | 40 |
| 4.2.2.             | Data dictionary.....                          | 42 |
| 4.3.               | Dynamic model .....                           | 44 |
| 4.3.1.             | Sequence diagram.....                         | 44 |
| 4.3.2.             | Activity diagram .....                        | 49 |
| 4.3.3.             | Statechart diagram.....                       | 53 |
| CHAPTER FIVE ..... |   | 57 |
| 5.                 | SYSTEM DESIGN .....                           | 57 |
| 5.1.               | Design goal .....                             | 57 |
| 5.2.               | Design Trade-offs .....                       | 58 |
| 5.3.               | Current system architecture .....             | 59 |
| 5.4.               | Proposed system architecture .....            | 59 |
| 5.4.1.             | Subsystem decomposition and description ..... | 60 |
| 5.4.2.             | Hardware/Software mapping .....               | 62 |
| 5.4.3.             | Detailed class diagram .....                  | 63 |

|                    |  |    |
|--------------------|--|----|
| 5.4.3.             | Persistent Data Management .....           | 64 |
| 5.4.4.             | Access Control and Security.....           | 65 |
| 5.5.               | Packages.....                              | 66 |
| 5.6.               | Algorithm Design.....                      | 67 |
| 5.7.               | User Interface Design.....                 | 69 |
| CHAPTER SIX .....  |  | 74 |
| 6.                 | IMPLEMENTATION AND TESTING.....            | 74 |
| 6.1.               | Implementation of the database .....       | 74 |
| 6.2.               | Implementation of class diagram .....      | 74 |
| 6.3.               | Configuration of application security..... | 77 |
| 6.4.               | Implementation user interface .....        | 77 |
| 6.5.               | Testing.....                               | 77 |
| 6.5.1.             | Test case.....                             | 77 |
| 6.5.2.             | Unit testing.....                          | 78 |
| 6.5.3.             | System testing.....                        | 79 |
| 6.5.4.             | Integration testing .....                  | 79 |
| 6.5.5.             | Acceptance testing.....                    | 79 |
| CHAPTER SEVEN..... |  | 80 |
| 7.                 | CONCLUSSION AND RECOMMENDATION .....       | 80 |
| 7.1.               | Conclusion.....                            | 80 |
| 7.2.               | Recommendation.....                        | 80 |
| References .....   |  | 81 |
| APPENDICES.....    |  | 82 |
| Appendix 1 .....   |  | 82 |

|                  |    |
|------------------|----|
| Appendix 2 ..... | 86 |
|------------------|----|

## LIST OF TABLES

|  |    |
|--|----|
| Table 1-1 Front end development tool.....  | 8  |
| Table 1-2 Back end development tool .....  | 9  |
| Table 2-1 BR1 Providing college degree to register .....                                     | 15 |
| Table 2-2 BR2 One instructor must not assign for different rooms at the same time .....      | 15 |
| Table 2-3 BR3 Further learner should pay a monthly fee.....                                  | 15 |
| Table 2-4 BR4 Basic sign learner should learn all alphabet and basic word.....               | 16 |
| Table 2-5 BR5 Intermediate sign learner should learn object name at home and classroom ..... | 16 |
| Table 2-6 BR6 Advance sign learner should learn more word and sentence making .....          | 17 |
| Table 4-1 Use case description - Login to the system .....                                   | 27 |
| Table 4-2 Use case description Register .....  | 29 |
| Table 4-3 Use case description Learn sign language .....                                     | 30 |
| Table 4-4 Use case description Exercise sign language .....                                  | 31 |
| Table 4-5 Use case description Make a video call.....  | 32 |
| Table 4-6 Use case description - Sign translation.....                                       | 34 |
| Table 4-7 Use case description - Add new sign.....   | 35 |
| Table 4-8 User data dictionary.....  | 42 |
| Table 4-9 Instructor data dictionary.....  | 43 |
| Table 4-10 Dictionary data dictionary .....  | 43 |
| Table 4-11 Intermediate data dictionary .....  | 44 |
| Table 5-1 Access control and security .....  | 65 |
| Table 6-1 Login test case .....  | 77 |
| Table 6-2 Add new sign .....   | 78 |

## LIST OF FIGURES

|  |    |
|--|----|
| Figure 4-1 ESL General use case diagram .....          | 25 |
| Figure 4-2 Learner detail use case diagram .....       | 26 |
| Figure 4-3 Instructor detail use case diagram .....    | 26 |
| Figure 4-4 Analysis level class diagram .....          | 41 |
| Figure 4-5 Sequence diagram for login.....             | 45 |
| Figure 4-6 Sequence diagram for register.....          | 46 |
| Figure 4-7 Sequence diagram for learn.....             | 47 |
| Figure 4-8 Sequence diagram for exercise.....          | 48 |
| Figure 4-9 Sequence diagram for Sign translation.....  | 49 |
| Figure 4-10 Administrator activity diagram .....       | 50 |
| Figure 4-11 Instructor activity diagram .....          | 51 |
| Figure 4-12 Learner activity diagram.....              | 52 |
| Figure 4-13 Learn state chart diagram .....            | 53 |
| Figure 4-14 Exercise state chart diagram.....          | 54 |
| Figure 4-15 Login state chart diagram.....             | 55 |
| Figure 4-16 Register state chart diagram .....         | 56 |
| Figure 5-1 Proposed system architecture .....          | 60 |
| Figure 5-2 Subsystem decomposition .....               | 61 |
| Figure 5-3 Deployment diagram .....                    | 62 |
| Figure 5-4 Detailed class diagram .....                | 63 |
| Figure 5-5 Persistence data diagram .....              | 64 |
| Figure 5-6 Package diagram.....                        | 67 |
| Figure 5-7 Login interface desktop.....                | 70 |
| Figure 5-8 Super admin page .....                      | 71 |
| Figure 5-9 Mobile app navigation in Amharic local..... | 72 |
| Figure 5-10 Mobile app English navigation .....        | 72 |
| Figure 5-11 Learn day page .....                       | 73 |
| Figure 5-12 Mobile learner home page .....             | 73 |

## ABBREVIATIONS

| <b><u>No</u></b> | <b><u>Acronyms</u></b> | <b><u>Meaning</u></b>              |
|------------------|------------------------|------------------------------------|
| 1                | <b>ESL</b>             | Ethiopia Sign Language             |
| 2                | <b>SL</b>              | Sign Language                      |
| 3                | <b>3D</b>              | 3 Dimensional                      |
| 4                | <b>OO</b>              | Object-Oriented                    |
| 5                | <b>OOA</b>             | Object-Oriented Analysis           |
| 6                | <b>OOD</b>             | Object-Oriented Design             |
| 7                | <b>UML</b>             | Unified Modeling Language          |
| 8                | <b>IDE</b>             | Integrated Development Environment |
| 9                | <b>BR</b>              | Business Rule                      |
| 10               | <b>ASL</b>             | American Sign Language             |
| 11               | <b>SQL</b>             | Structured Query Language          |
| 12               | <b>HCI</b>             | Human-Computer Interface           |
| 13               | <b>UI</b>              | User Interface                     |

## **CHAPTER ONE**

### **1. INTRODUCTION TO THE PROJECT**

#### **1.1. Introduction**

Ethiopian Sign Language (ESL) is the natural language used primarily by about a million Ethiopian deaf community. ESL, as others know sign language (SLs), is accepted as a minority language, which coexists with majority languages [1] and is the native language for many deaf people. The language's fundamental units are base signs that correspond to the traditional nation of the morpheme. SLs are composed of three-dimensional (3D) manual and non-manual features. Hand-shape, hand orientation, hand position, and hand movement are the manual component and non-manual features include the position of the upper torso, mouth pattern, head and shoulder movement, facial expression and eye-gaze.

Ethiopian deaf education is constrained by many factors. Among others is the lack of the development of sign language. Lack of proficiency (or nearly the total absence) sign language teachers is another constraint. There are only a few deaf people employed at the deaf schools in Addis Ababa without any training in the teaching profession. Only hearing people were trained as teachers of the deaf. These hearing teachers lack proficiency in Sign Language and tend to promote Orals or their spoken languages. Due to these facts, the Ethiopian deaf students are denied strong role models in their education. The deaf students were, therefore, subject to the wishes of their hearing teachers and professionals.

#### **1.2. Statement of the problem**

Hearing parents of deaf children have the same needs as all parents to communicate with their child to fulfill their role as parents. However, parent-child interaction is seriously threatened when parents cannot communicate in their natural spoken language, and have to learn to communicate through a visual modality using sign language. More than 90% of deaf children are born from hearing parents who have had no prior experience with deafness and the use of signing. In Ethiopia, many parents are severely disadvantaged as they are unable to access sign language. [2]

In the age of globalization, information is key for facilitating everyone's activities and it fulfills individual or organizational interests. Information is exchanged between different individuals, organizations, and countries. Individuals may have different personalities, behavior, and culture; also, some of them may have a disability such as hearing impairment. Those differences can be a barrier to communication and may lead to misunderstanding. Besides these barriers, language differences have a negative impact on communication such as the communication gap between the deaf and non-deaf people.

In Ethiopia using a translator person (maybe family or close friend) is a solution for filling the communication gap between deaf people with non-deaf people. Even though the translators play a great role in the translation process, sometimes it's difficult to exchange confidential information between these two parties, for example in closed court, medical area, and different social issues. Also, using a translator person is need more economical cost, time and effort. Also, some of the main problems are listed below: -

- No software or site available for sign language learning
- National work language is Amharic (unable to get work with lack of communication)
- Written representation is in syllables
- Many deaf people in Ethiopia
- Little access to hearing aids or cochlear implantation
- Misunderstanding between deaf and no deaf person
- Lack of adapted curriculum and adapted technology
- Insufficient materials and equipment provision
- A limited understanding of the concept of disability
- Lack of a written form of sign language rules and gestures.



### **1.3. Objective**

#### **1.3.1. General objective**

The main objective of this project is to design and develop a new web and mobile application for sign language learning.

#### **1.3.2. Specific objective**

- Identify all problem related to Ethiopian sign language
- Study and analysis of the current learning system of ESL
- Review related work with ESL
- Gather the required information from the sign language school
- Propose a system that considers the current ESL system
- Designing and develop Amharic-word to ESL mapping system
- Designing and develop Amharic-alphabet to ESL mapping system
- Designing and develop Amharic-number to ESL mapping system
- Designing and develop Amharic-word to ESL mapping system
- Implement the designed (proposed) system
- Test and evaluate the implemented system
- Deployment the tested system

### **1.4. Feasibility Analysis**

The feasibility study is a test of system proposal according to its workability, impact of the organization, ability to meet needs and effective use of the resources. The objective is to determine whether the proposed system is feasible or not through the following feasibility study mechanisms.

#### **1.4.1. Operational feasibility**

Our system which has been developed will provide accurate, active, secured service and easy to use. Since the system is mobile and web application, the mobile app is on android (most of the

users use it) and the web application is platform-independent which indicates that the system will be operationally feasible.

### **1.4.2. Technical feasibility**

In a technical feasibility study, one has to test whether the proposed system can be developed using existing technology or not. It is planned to implement the proposed system using android studio IDE in windows operating system, mobile applications. The necessary hardware and software are available for the development and implementation of the proposed system. So, the system will technically feasible. The assessment is based on an outline design of system requirements, to determine whether the user has the technical expertise to handle the new app. Most of Ethiopian use android based mobile. Hence the solution is technically feasible.

### **1.4.3. Economic feasibility**

Our system will not need special requirements like standalone server, additional cost incurred for tools and large cost for system development process the system will be economically feasible.

Currently, the third-party translator is the solution to communicate between the deaf and normal person and it's costly and needs much more effort, so our system reduces the cost and effort during the communication between deaf and normal people.

### **1.4.4. Schedule feasibility**

Schedule feasibility means that the project can be implemented in an acceptable time frame or the probability of a project to be completed within its scheduled time limits, by a planned due date. So, the project accomplished and delivered within the given time: also, the technology we used to develop the system is known and there is a work division in our group. Due to this reason the team will deliver based on the given time scope.

## **1.5. Scope and limitation of the project**

There are numbers of language to be translated in to sign, but this project is particularly focused on Amharic and English to sign language mapping. That is a number, letter, word or sentence to image or animation mapping. The main activities are teaching language mapping from basic (like alphabet and number), testing the user knowledge and passing to the next level of learning (like word and sentence). We provide everyone sign language teachers to add some words to improve this application from everywhere.

The web application has the same functionality as the android, in addition, it has the features that provide for a valid user to add sign language with its associated word or sentence representation.

As a limitation our system currently doesn't include advance machine learning concept it's only mobile and web-based sign language mapping system with basic sign translation.

### **1.6. Significance of the project**

The major beneficiaries of this project will be the whole national and international community which includes deaf person, parent of the deaf child, anyone who wants to learn sign language, deaf person instructor, community, and country in the following ways: -

#### **1.6.1. Deaf person**

- To encourage deaf children or students to join the school, science, and technology
- To learn sign language easily
- To enable communication with the normal (nondeaf) person
- To increase the attitude (when communicating with anyone there is no tension)
- To understand thing easy (communication with everyone)
- Learn sign language from everywhere at every time.

#### **1.6.2. Parent of the deaf child**

- Reduce the conflict with a person, it's communicating by sign language
- Silent communication
- Enable to treat his/her deaf children without misunderstanding and fulfill their needs child as parent
- Learn sign language at home or everywhere at any time

#### **1.6.3. Anyone who wants to learn sign language**

- Increase communication with deaf person and reduce conflict with them
- Introduce you to the deaf culture and community
- Bilingualism boost brain

- Increase silence by communicating with sign language only
- Improves your peripheral vision and reaction time
- You can communicate with babies by you have some basic communications skill in sign language
- Silent communication
- Meet people and make new friends by learning sign language you can also build relationships with new people
- Learn sign language everywhere

#### **1.6.4. Schools, instructors, community and country**

- It minimizes the communication gap and/or misunderstanding between the deaf and normal person
- Introduce you to the deaf culture and community
- Meet people and make new friends by learning sign language you can also build relationships with new people
- Improves your peripheral vision, reaction time, according to a study by the University of Sheffield, sign language users have better peripheral vision and reaction time.
- You can communicate with babies by you have some basic communications skill in sign language
- Bilingualism boost brain

#### **1.7. The methodology of the project**

In the proposed system, we used different methods to develop the system, to gather information or data of the existing system or from the stakeholder. As we are going to use an object-oriented system for developing proposed we used iterative software development models because testing and debugging during smaller iteration is easy and we are built and improve the product step by step.

### 1.7.1. Data Collection

Data collection is the most important part of the project to find the main requirement of the system and to understand how the system works. We used different methods to collect data. Among the methods, we use the following: -

- **Interviews:** We used interviews for gathering direct information from a stockholder. Because interviewing is a popular way of gathering qualitative data because it is perceived as “talking”, and talking is natural.
- **Document analysis:** We also used document analysis as a data collection method by reviewing different books, websites, recorded human communications, laws, and regulations (if any) to get basic information regarding sign language and make certain deductions based on the study parameter.

### 1.7.2. System Analysis

We used the object-oriented software engineering methodology (approaches) to develop the system. because it is a popular technical approach for analyzing, designing an application system, or business by applying the object-oriented paradigm and visual modeling throughout the development life cycles to foster better stakeholder communication and software product quality. Object-Oriented Analysis is selected since it has so many advantages and which can make the system more effective.

The advantages of Object-Oriented Analysis are:

- It allows effective management of software complexity by the virtue of modularity.
- It can be upgraded from small to large systems at greater ease than in systems following structured analysis.
- The principles of encapsulation and data hiding help us to develop systems that cannot be tampered by other parts of the system.
- Focuses on data rather than the procedures as in Structured Analysis.

- Increased consistency among analysis, design and programming activities.
- Common design and coding methodologies, including Object-Oriented Design (OOD), Design Patterns, Refactoring, and the Unified Modeling Language (UML)
- Increased consistency among all the models developed during Object-Oriented Analysis, design and implementation.

## 1.8. Development tool

### 1.8.1. Front end

Table 1-1 Front end development tool

| No. | Name            | Version | Reasons  |
|-----|-----------------|---------|--|
| 1   | Java            | 8       | It is the latest release of java and it's more popular for android application development everyone on stack overflow or Quora. help us if we have any error, so we choose java for android application development  |
| 2   | Bootstrap       | 4       | It makes us build interactive, easy to use web application, and bootstrap 4 is the latest version of bootstrap that include much more thing than the previous bootstrap version to build a userfriendly application. |
| 4   | Edraw Max       | 9.4     | The current version of Edraw Max is 9.4 An all-inclusive diagramming software that is capable of 260+ drawing types including flowcharts, mind maps, org charts, ...   |
| 5   | Adobe Photoshop | 20.0    | The October 2018 release of Photoshop (version 20.0) rolls out exciting new features for designers, and illustrators and it support us to create animation or to edit image easy for sign language representation.   |

|   |                              |  |  |
|---|------------------------------|--|--|
| 6 | Different online application |  | For image size decreasing, gif maker and background remover. |
|---|------------------------------|--|--|

### 1.8.2. Back end

Table 1-2 Back end development tool

| No. | Name     | Version | Reasons   |
|-----|----------|---------|---|
| 1   | Firebase | 6.12    | Hosting: Firebase provides fast, secure, static, and productiongrade hosting for developers. It allows us to efficiently deploy web apps and static content. It gives us free storage: It is another best advantage of Firebase   |
| 2   | Django   | 2.2.7   | Django allows us to build secure web-application in a few days, it doesn't expect to matter about the session and another thing most of the activity is done in Django by default. Python 3.8.0 is the newest major release of the Python support Django 2.27 and it's an easy and fast way to develop a good website that is why we choose Django for backend web development. |
| 3   | MySQL    | 7.2     | Because it's a simple relational database for us, and it simplifies the creation of number, alphabet, word or sentence to sign language mapping.  |

## 1.9. Document organization

The proposed system document contains five-chapter each chapter describe it in the following manner:

- **Chapter one:** - In chapter one we describe the introduction of the existing system, the problem of the existing system, objective (general and specific objective) and methodology we followed during data collection and analysis.

- **Chapter two:** - This chapter describes what the existing system looks like in detail (who uses the existing system with major function and drawback of the existing system described in this chapter in detail).
- **Chapter three:** - This chapter is about the proposed system which includes the functional and nonfunctional requirements of the proposed system.
- **Chapter four:** - Describe the analysis of the proposed system consists of the flow of events in the scenario, use case model with its description of major use case, detail model and dynamic model of the proposed system.
- **Chapter five:** - Deals with system design. Which includes the overview of the system, design consideration, design goal, design tradeoffs, the architecture of the System, subsystem decomposition, persistent data management, and user interface design.



## CHAPTER TWO

### 2. DESCRIPTION OF THE EXISTING SYSTEM

#### 2.1. Introduction of Existing System

Sign language is the primary language for almost all deaf persons. Sign language mapping from Amharic or English helps the deaf person to communicate with anyone and encourage them to join school, science, and technology, and contribute to the development of our country.

In Ethiopia, to learn sign language deaf person or anyone's person who want to learn sign language should have to go to school. Then learn from sign language instructor by signing and writing the association word. For this reason, the learner doesn't want to learn sign language because it is tedious, time-consuming, a long-distance from their town or no access at all (there is a limited sign language school in Ethiopia). This project is to create a sign language learning application from anywhere anytime.

Currently, there is one sign language application that has only sign for Amharic and English alphabet and that is not full even if for all alphabet. Of course, we can say there is no online or mobile application for sign language learning the only thing is from the instructor by going to school if exists.

#### 2.2. Users of Existing System

Currently, sign language learning system is executed traditionally or manually for the learner it's learning system is by the lecturer (deaf instructor). The existing system has the following users: -

- **Deaf person:** - In the existing system, deaf persons are a learner who goes to school and appear in class according to the school schedule, and learn from his/her instructor by watching his/her sign, word written and image drawn (if any).
- **Instructor:** - In the existing system instructors are the ones who teaches a deaf person. Instructor prepares themselves on the part of the class to teach, he/ she can use the teacher guide book as a reference.

- **Parents:** - Deaf person parents also use the existing system in order to communicate with their deaf child. parents learn sign language in order to tell sign language representation of alphabet, object name or conversation to his/her deaf child.
- **Community:** - Community uses the existing system to share information with a deaf person and facilitate communication with them. There is a communication gap between deaf and normal people because the deaf person can't hear and the normal person neither understands the sign of the deaf person. In the existing system third person that understands some sign language used as a translator to overcome this problem.

### **2.3. Major Functions of the Existing System**

The existing system for sign language learning is manual and the functionality and learning process is as follows: -

- Learner (E.g. Deaf person) must go to school
- The instructor prepares itself for the teaching schedule and level of class he/she teaches.
- The learner sits on the chair and looks for the sign of the instructor.
- Instructor start teaching by signing and writing the representation of the sign
- It's almost similar with all teaching-learning process but it uses sign language instead of oral

### **2.4. Forms and Other Documents of the Existing Systems**

There is a book used by an instructor as a form to teach a deaf person. The book is contained sign language representation of alphabet, number, word, and sentence with associated image description for almost all signs. The book is also used by the sign language learner for further understanding. Look at Appendix 1.

### **2.5. Drawbacks of the Existing System**

The existing system has may problem some of them are listed below: -

- Negative attitudes towards persons with disabilities and a hardened resistance
- Shortage of teachers training programs

- Lack of adapted curriculum
- Lack of adapted technology
- Insufficient materials and equipment provision
- Poor Financial sources are among the frontiers of challenge in the implementation of inclusive education.
- Less integrated with hearing students or absence of communication with other hearing peers
- Less motivation among teachers who teach hearing impaired /Many teachers did not show interest to update their teaching method to use the flexible curriculum to ease the learning of students with hearing impairment.
- Not everyone knows sign language,
- The learner must have to go to school.
- There is limited access to school or long-distance from home.
- Afraid of going to school for a deaf person.
- Afraid of asking a question for an instructor.
- Lack of transferring knowledge (for the instructor) because most of the time instructors are trained normal persons who prior oral instead of a sign, so the deaf person may get confused and not even understand.
- Time-consuming when going and back from home to school.
- Limited instructor, everyone who want to learn sign language cannot get the opportunity to learn.
- No means for the parent or everyone who wants to learn sign language at home or everywhere.
- No book on the market to buy and use it's only in deaf schools. ● Majority of the deaf struggle with reading.
- Deaf persons can't have a conversation with sign language when a person is in another room.

## **2.6. Business Rules of the Existing System**

A business rule is effectively an operating principle or policy the software must satisfy. The ESL governs and controls the workflow through the following business rules. These are rules and policy which are used to govern all actives in the specified workflow, control the workflow, and performed in the work environment. The proposed system (ESL) provides the following business rules for the user of the system.

Table

Table 2-1 BR1 Providing college degree to register

|                     |  |
|---------------------|--|
| <b>Name:</b>        | Providing college degree to register   |
| <b>Identifier:</b>  | BR1  |
| <b>Description:</b> | Only registered instructors can teach sign language and to register as an instructor of sign language, he/she must provide the qualified documents in sign language at least at the college level. |
| <b>Example:</b>     | Mr. Kefelegn instructor of sign language at Wolkite town provides a college degree to register and become an instructor of sign language.  |
| <b>Source:</b>      | School policies and procedures   |

Table 2-2 BR2 One instructor must not assign for different rooms at the same time

|                     |   |
|---------------------|---|
| <b>Name:</b>        | One instructor must not assign for different rooms at the same time   |
| <b>Identifier:</b>  | BR2   |
| <b>Description:</b> | One instructor cannot teach in more than one class at the same time he/she can only teach one class. This is to avoid conflict. |
| <b>Source:</b>      | School policies and procedures  |

Table 2-3 BR3 Further learner should pay a monthly fee

|                     |   |
|---------------------|---|
| <b>Name:</b>        | The further learner should pay a monthly fee  |
| <b>Identifier:</b>  | BR3   |
| <b>Description:</b> | Anyone who wants to learn sign language (includes deaf if he/she want to learn further) should be registered and pay a monthly payment. |

Table

|                 |  |
|-----------------|--|
| <b>Example:</b> | If I want to learn sign language I must first be registered as a sign language student and pay monthly payment according to the rule of the organization or group of people who teach sign language. |
| <b>Source:</b>  | Mrs. Aynie Belete, a sign language instructor at Debre Birhan college  |

Table 2-4 BR4 Basic sign learner should learn all alphabet and basic word

|                       |   |
|-----------------------|---|
| <b>Name:</b>          | Basic sign learner should learn all alphabet and basic word   |
| <b>Identifier:</b>    | BR4   |
| <b>Description:</b>   | Basic sign language learners should learn the sign for all numbers and alphabets for both English and Amharic, and this level of learner also learns basic of greeting word, human body, and animal name. |
| <b>Source:</b>        | School policies and procedures  |
| <b>Related rules:</b> | BR5 Intermediate sign learner should learn object name exists at home and classroom.<br>BR6 Advance sign learner should learn sentence making   |

Table 2-5 BR5 Intermediate sign learner should learn object name at home and classroom

|                     |  |
|---------------------|--|
| <b>Name:</b>        | Intermediate sign learner should learn object name exists at home and classroom.   |
| <b>Identifier:</b>  | BR5  |
| <b>Description:</b> | Intermediate level sign language learner should learn sign representation of an object that exists in the classroom and at home, and making basic sentence for greeting and self-introduction. |
| <b>Source:</b>      | School policies and procedures   |

Table

|                       |  |
|-----------------------|--|
| <b>Related rules:</b> | BR4 basic sign learner should learn all alphabet and basic word<br>BR6 Advance sign learner should learn more word and sentence making |
|-----------------------|--|

Table 2-6 BR6 Advance sign learner should learn more word and sentence making

|                       |  |
|-----------------------|--|
| <b>Name:</b>          | Advance sign learner should learn more word and sentence making  |
| <b>Identifier:</b>    | BR6  |
| <b>Description:</b>   | In advance sign language learners should learn more words on everything in the environment, on sentence making to communicate with anyone in sign language and understand sign language grammar. |
| <b>Source:</b>        | School policies and procedures   |
| <b>Related rules:</b> | BR4 Intermediate sign learner should learn all alphabet and basic word BR5 Intermediate sign learner should learn object names at home and classroom.  |

## CHAPTER THREE

### 3. PROPOSED SYSTEM

#### 3.1. Functional requirement

The proposed system is expected to provide all sign language-related services and functionalities, like sign language learning, exercise sign language, video call, and Sign translation. For a better understanding of the proposed system, we divided the functionalities of the proposed system by modules. The new system is expected to provide the following functionalities:

##### **Learn**

- Choose the level of learning (basic, intermediate, advanced)
- Search sign representation using keyword
- Download sign (if not previously downloaded)
- Play the animation for the selected sign (if any)
- Favorite selected sign
- Trash unwanted sign

##### **Exercise**

- Choose the level of learning (basic, intermediate, advanced)
- Practice selected level sign language
- Choose an answer for displayed sign representation
- Go to the previous and next question
- See the correct answer for each question
- See result for overall test
- See test progress
- Reset test progress



### **Video call and chatting**

- Search for a friend by username or phone number
- Add new friend using phone number or username
- Selected friend and make a call

### **User module**

- Register the user to the system
- Login the user and redirect him/her to their respective landing page based on their role (instructor, learner or admin)
- Rate the mobile application
- Update user profiles (edit name, change photo)
- An instructor can add new word train them in and upload for another user to be accessible
- Admin can add new instructors

### **Sign translation**

- Use a camera to capture video
- Learner start signing (hand gesture)
- The system automatically recognizes and display text representation or
- Unknown sign for an untrained sign.

## **3.2. Non-functional requirement**

The following are the nonfunctional requirements that will be applied to the proposed system.

### **3.2.1. User interface and human factors**

The interface of the proposed system provides easy to understand, easy to use and user-friendly interface and users of the system easily use and perform their task. Everything stems from knowing our users, including understanding their goals, skills, preferences, and tendencies. Once we know about our users, we can develop a system that satisfies our user needs.

We consider the following list while designing our interface:

- **Keep the interface simple:** We avoid unnecessary elements and keep the user interface clear in the language they use on labels and in message response to the user from the application. We also support local language for users to select and make any label named locally (easy to understand).
- **Create consistency and use common UI elements:** By using common elements in our UI, users feel more comfortable and can get things done more quickly, layout and design throughout the site and the application to help facilitate efficiency.
- **Be purposeful in page layout:** Consider the spatial relationships between items on the page and structure the page based on importance. Careful placement of items can help draw attention to the most important pieces of information and can aid scanning and readability.
- **Strategically use color and texture:** You can direct attention toward or redirect attention away from items using color, light, contrast, and texture to your advantage, for example, make text or button red color for danger (maybe like for delete button or error message).
- By carefully thinking about and anticipating the goals people bring to our site or application, we create defaults to reduce the burden on the user.

### 3.2.2. Security issues

The system is much have secured based on the username and password for all user activity. Nobody can access the system without an authorized person. Passwords are not visible cannot be accessed by anyone because the passwords are encrypted. We use the Django python framework for backend web development it allows us to develop highly secured applications even without writing difficult security algorithms.

### 3.2.3. Performance considerations

The web-based system should have easy and efficient code manipulation and have a clear database. Thus, the response time should be very small i.e. not more than 0.2 seconds. We are confident that the android system has much faster than our website. Because native android applications are usually 1.5 times faster than websites and they perform actions much faster too and we used the native.

#### **3.2.4. Error handling and validation**

Django gives us good URL management (even we can use regular expression for URL) so the URL entered by the user checked and return for a specific page or return page doesn't exist if the user enters the wrong URL and mentioned to enter correct path. The system checks user inputs to the system to handle the error. It handles and shows error in a user-friendly manner, without exaggerating the user. We handle this error on the client-side by using javascript validator. We will use front end validation to reduce the loading time of pages. If there is any server-side error, it is only visible on the server log, they are not visible to the user.

#### **3.2.5. Quality issues**

The quality issue is about how reliable the system should, we can say our system is reliable as we mentioned before the system is consistently perform its intended function. Even if there is no internet connection mobile application user can use the application and access the downloaded sign offline, so the proposed system is available at any time 24/7. In the quality assessing the users will be involved by a feedback mechanism in which they can give comments and rate on the application.

#### **3.2.6. Physical environment**

The web application of the proposed system is deployed on a server that is accessible over the internet in a specific address, and the mobile application of the proposed system would install on an android (min 5.0) based mobile device. For both, there is no need for special hardware device to work only your android based phone to access the mobile application or computer to use the web application.

#### **3.2.7. Documentation**

For a system user-level document is provided that the user can read the document to know how to use the system and what functionality should the system support to the user. Also, the development process is provided that the user can read to know about the process and what type of model the developer uses to develop the system. We prepared a short and precise help file on how to use the application for the application user.

## **CHAPTER FOUR**

### **4. SYSTEM ANALYSIS**

In this chapter, we describe the system model, object model, and dynamic model. System analysis is the analysis of the role of a proposed system and the identification of a set of requirements that the system should meet, and thus the starting point for system design. The design is then passed to the programmers, who are responsible for the actual implementation of the system. Analysis model contains three models: Functional, object and dynamic models. The functional model can be described by use case diagrams. Class diagrams and object diagram describes the object model. The dynamic model can also be described in terms of sequence, activity and state chart diagrams.

#### **4.1. System model**

The system model is the conceptual model as a result of system modeling that describes and represents a system. It is the process of developing abstract models of a system, with each model presenting a different view or perspective of that system. System modeling has now come to mean representing a system using some kind of graphical notation, which is now almost always based on notations in the Unified Modeling Language (UML), and we also use this modeling language. It helps us to describe the functionality of the proposed system (ESL) application and provides an essential representation of software that can be mapped into implementation.

##### **4.1.1. Use case model**

A use case defines a goal-oriented set of interactions between external actors and the system under consideration. Actors are parties outside the system that interact with the system. An actor may be a class of users, roles users can play, or other systems. A use case is initiated by a user with a particular goal in mind and completes successfully when that goal is satisfied. It describes the sequence of interactions between actors and the system necessary to deliver the service that satisfies the goal. It also includes possible variants of this sequence, e.g., alternative sequences that may also satisfy the goal, as well as sequences that may lead to failure to complete the service because of exceptional behavior, error handling, etc. Use cases are an effective technique for narrating the ways a system works in fulfilling a user story/requirement. Use cases explain what happens when someone uses the system to (try to) achieve a goal, this will help us understand how

it should work in detail. Each use case explains one or more **scenarios** that describe how the system should interact with an **actor** to achieve a specific **goal**.

- A **Scenario** is a narrative that describes what happens within a specific use case.
- An **Actor** is an external agent that interacts with the system.
- A **Goal** is a specific thing accomplished as a result of executing a use case

The use case model (use case diagram, use case description, use case scenario) is used to define the core elements and processes that make up the proposed system. This use case model captures the functional system components. Because use case models are simple in nature, this use case models are a great way to storyboard flows with users and define the system requirements being modeled and help write the scenarios later used in testing.

#### 4.1.1.1. Use case diagram

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved.

In the proposed system we identified both actor and use case (service, task or functionality) as the following.

- Administrator
- Instructor
- Learner and each of those actors can perform actions that are specific to a particular role as follows: -
  - ✓ Administrator
    - ✦ An administrator can approve registered instructors.
    - ✦ An administrator can login and logout to the system.
    - ✦ An administrator can manage their profiles.
    - ✦ An administrator can check the validity of the new sign added by the instructor.
  - ✓ Instructor

- ✦ An instructor can manage their profile.
  - ✦ An instructor can login and logout to the system.
  - ✦ An instructor can add a new word.
  - ✦ An instructor can register.
  - ✦ An instructor can learn sign language.
- ✓ Learner
- ✦ The learner can learn selected levels of sign language.
  - ✦ The learner can exercise selected levels of sign language.
  - ✦ The learner can register to the system.
  - ✦ The learner can login and logout to the system.
  - ✦ The learner can manage his/her profiles.
  - ✦ The learner can make a video call to his/her friends.
  - ✦ The learner can Sign translation that can be automatically recognized and converted to associated text by the system.
  - ✦ The learner can favorite signs.
  - ✦ The learner can trash unwanted sign from his/her device.
  - ✦ The learner can rate the mobile application

Figure 4.1 show the general use case of the proposed system.

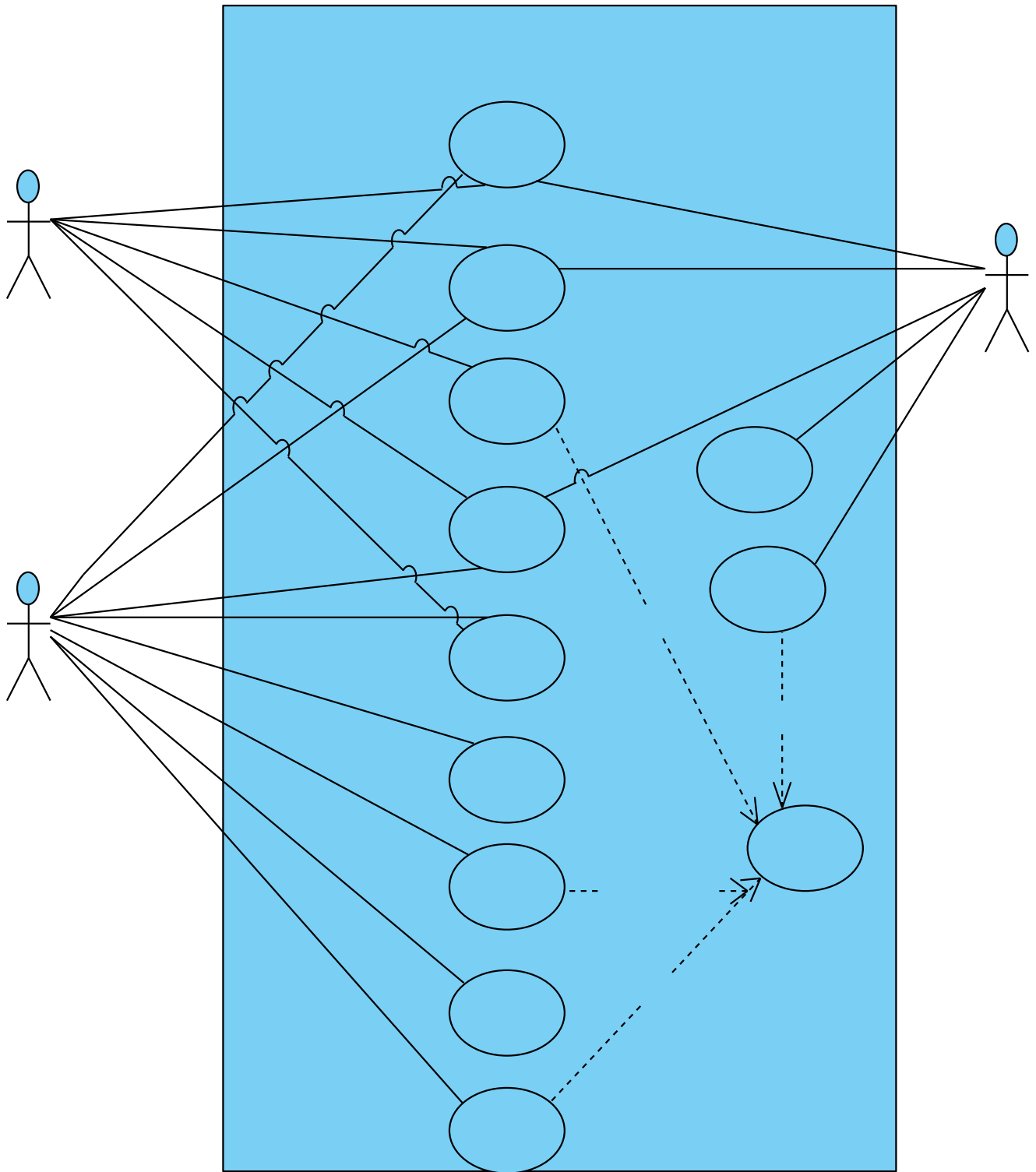


Figure 4-1 ESL General use case diagram

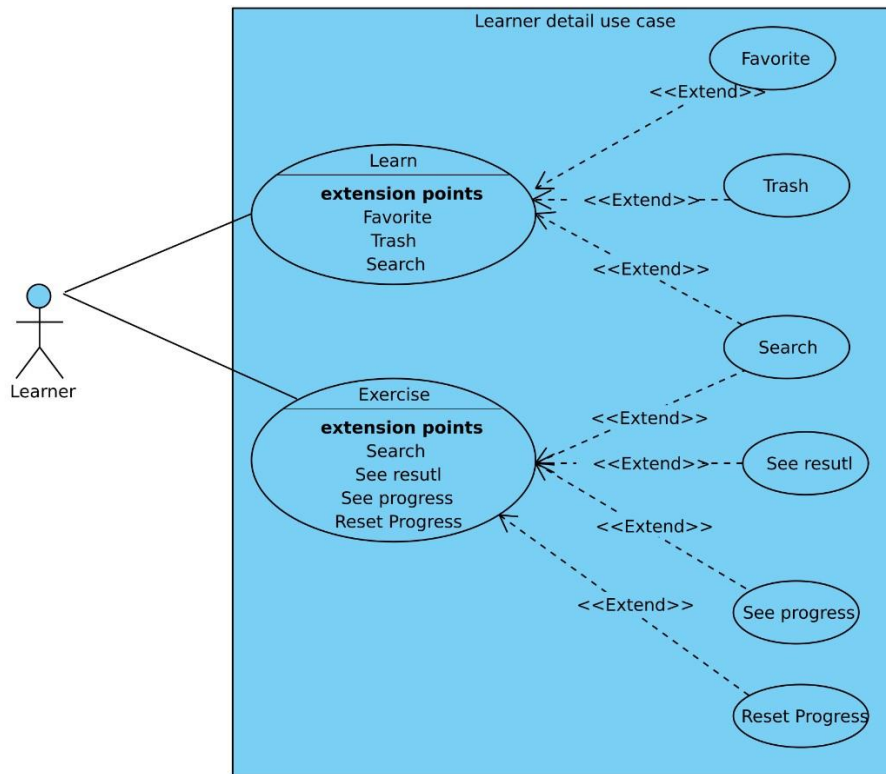


Figure 4-2 Learner detail use case diagram

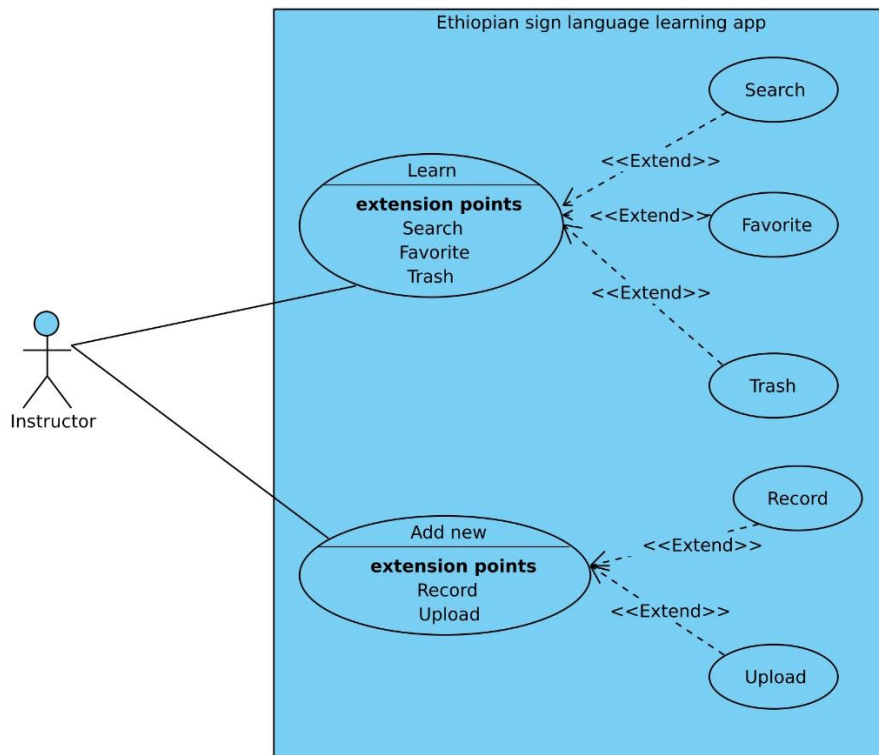


Figure 4-3 Instructor detail use case diagram



## 4.1.1.2. Use case description

The use case description is a written account of the sequence of steps performed by an actor to accomplish a complete business transaction. It's initiated by an actor, provides value to that actor, and is a goal of the actor working in that system. Below, you see a use case description that clearly documents how the actor identified above (in 4.1.1.1.) interacts with the proposed system in order to perform user and system functionality.

*Table 4-1 Use case description - Login to the system*

|                      |   |
|----------------------|---|
| Use case id          | UC_01   |
| Use case title       | Login   |
| Use case description | This use case describes how the user uses the proposed system in order to login to the system.  |
| Actors               | <ul style="list-style-type: none"> <li>• Administrator</li> <li>• Instructor</li> <li>• Learner</li> </ul>  |
| Pre-condition        | <ol style="list-style-type: none"> <li>1. Users should have a valid username and password (valid account).</li> <li>2. Computer to surf the web application or mobile phone that has this Ethiopian sign language application installed on it.</li> </ol>   |
| Basic flow of events | <ol style="list-style-type: none"> <li>1. The use case begins when the user opens the application</li> <li>2. The user clicks “login”</li> <li>3. The application show form to enter user name and password.</li> <li>4. The user fill username and password</li> <li>5. Username and password sent to the controller; the controller replies with a go/no go reply telling if the username and password are ok.</li> <li>6. The application redirects to the user's homepage according to their privilege.</li> <li>7. The use case ends successfully</li> </ol> |

|                  |   |
|------------------|---|
| Alternative flow | <p>Step [5]: If the entered username and/or password are not correct.</p> <ol style="list-style-type: none"><li>1. The system shows the wrong username or password error.</li><li>2. The system doesn't grant permission to access.</li><li>3. The system informs the user to reenter username and/or password.</li><li>4. The use case resumes at step 4 of the main sequences.</li></ol>  |
| Postcondition    | <ol style="list-style-type: none"><li>1. Successful completion<br/>The application shows user homepage according to their privilege</li><li>2. Failure condition<ol style="list-style-type: none"><li>a. The user doesn't have access to the application if there is no internet connection.</li><li>b. The application shows an error message for an invalid username or password and stays in the login page.</li></ol></li></ol> |

Table 4-2 Use case description Register

|                      |  |
|----------------------|--|
| Use case id          | UC_02  |
| Use case title       | Register to Ethiopian sign language learning application   |
| Use case description | This use case describes how user uses this application in order to register and get a valid account to login.  |
| Actors               | <ul style="list-style-type: none"> <li>• Learner •</li> <li>Instructor</li> </ul>  |
| Precondition         | Install the application on android device or open in the web browser   |
| Basic flow of events | <ol style="list-style-type: none"> <li>1. This use case begins when the user opens the application</li> <li>2. The user clicks “register”</li> <li>3. The application show form to enter full information.</li> <li>4. The user fills all required information</li> <li>5. User information sent to the controller; the controller replies with a go/no go reply telling if the user information is ok.</li> <li>6. The application redirects to the homepage.</li> <li>7. The use case ends successfully</li> </ol> |
| Alternative flow     | <p>Step [5]: If the entered information is validation error</p> <ol style="list-style-type: none"> <li>1. The system doesn’t create an account.</li> <li>2. The system informs the user to fills all required fields correctly.</li> <li>3. The use case resumes at step 4 of the main sequences</li> </ol>  |
| Postcondition        | <ol style="list-style-type: none"> <li>1. Successful completion<br/>The application creates and adds user account to the database.</li> <li>2. Failure condition <ol style="list-style-type: none"> <li>a. The user doesn’t have access to the application if there is no internet connection.</li> </ol> </li> </ol>  |

|  |   |
|--|---|
|  | b. The application shows an error message for invalid personal information. |
|--|---|

Table 4-3 Use case description Learn sign language

|                      |   |
|----------------------|---|
| Use case id          | UC_03   |
| Use case title       | Learn sign language   |
| Use case description | This use case describes how a deaf person or anyone who wants to learn sign language use this application to learn and know basic, intermediate or advance sign language.   |
| Actors               | <ul style="list-style-type: none"> <li>• Learner •</li> <li>Instructor</li> </ul>   |
| Pre-conditions       | Install Ethiopian sign language learning application or open in browser   |
| Basic flow of events | <ol style="list-style-type: none"> <li>1. The use case begins when learner open the application</li> <li>2. The application shows an alternative to learn, exercise, make a video call and Sign translation. But in this case, learners always select learn.</li> <li>3. The system shows a level of sign language (basic, intermediate, advance) to be select.</li> <li>4. Learners select level of sign language to learn.</li> <li>5. System display list of sign language based on the selected level of sign language.</li> <li>6. Learners practice sign language by watching and playing animation (if any).</li> <li>7. The use case end successfully.</li> </ol> |
| Alternative flow     | <ol style="list-style-type: none"> <li>1. No response from the application               <ol style="list-style-type: none"> <li>a. Application retries to connect with the internet.</li> <li>b. The application shows no internet connection.</li> </ol> </li> </ol>   |

|               |   |
|---------------|---|
| Postcondition | <ol style="list-style-type: none"> <li>1. Successful completion <ol style="list-style-type: none"> <li>a. The user mentioned to take a quiz for the selected level of sign language learning.</li> </ol> </li> <li>2. Failure condition <ol style="list-style-type: none"> <li>a. The user doesn't have access to the application.</li> </ol> </li> </ol> |
|---------------|---|

Table 4-4 Use case description Exercise sign language

|                      |   |
|----------------------|---|
| Use case id          | UC_04   |
| Use case title       | Exercise sign language.   |
| Use case description | This use case describes how learners use this application to exercise (take the quiz) sign language.  |
| Actors               | Learner   |
| Pre-conditions       | Install Ethiopian sign language learning application or open in the browser.  |
| Basic flow of events | <ol style="list-style-type: none"> <li>1. The use case begins when the learner opens the application.</li> <li>2. The application shows an alternative to learn, exercise, make a video call and Sign translation. But in this case, learners always select exercise.</li> <li>3. The system shows level of sign language (basic, intermediate, advance) to be select.</li> <li>4. Learners select level of sign language to exercise.</li> <li>5. Application display list of questions for the learner to exercise based on the selected level of sign language.</li> <li>6. The learner chooses the answer and goes to the next question.</li> <li>7. Use case end successfully</li> </ol> |
| Alternative flow     | <ol style="list-style-type: none"> <li>1. No response from the application <ol style="list-style-type: none"> <li>a. Application retries to connect to the internet.</li> <li>b. The application shows no internet connection.</li> </ol> </li> <li>2. Wrong answer</li> </ol>  |

|               |  |
|---------------|--|
|               | a. Learner select another choice   |
| Postcondition | <ol style="list-style-type: none"> <li>1. Successful completion                             <ol style="list-style-type: none"> <li>a. The learner can see the total result of their quiz.</li> </ol> </li> <li>2. Failure condition                             <p style="margin-left: 40px;">The user doesn't have access to the application.</p> </li> </ol> |

Table 4-5 Use case description Make a video call

|                      |  |
|----------------------|--|
| Use case id          | UC_05  |
| Use case title       | Make a video call  |
| Use case description | This use case describes how the user uses this application in order to make a video call and chat with her/his friends.  |
| Actors               | <ul style="list-style-type: none"> <li>• Learner</li> <li>• Administrator</li> <li>• Instructor</li> </ul>   |
| Precondition         | <ol style="list-style-type: none"> <li>1. User must have login to the system</li> <li>2. There is an active internet connection to use this application</li> <li>3. Computer to surf the web application or mobile phone that Ethiopian sign language learning application installed on it.</li> </ol> |

|                      |  |
|----------------------|--|
| Basic flow of events | <ol style="list-style-type: none"> <li>1. The use case begins when the user opens the application.</li> <li>2. The application shows an alternative to learn, exercise, make a video call and Sign translation. But in this case, the user always selects make a video call.</li> <li>3. The application shows a list of friends to make a call.</li> <li>4. The user selects one friend at a time to make a video call.</li> <li>5. The application opens a camera and starts recording and show video from his/her friends.</li> <li>6. The use case ends successfully.</li> </ol> |
| Alternative flow     | <ol style="list-style-type: none"> <li>1. No response from the application <ol style="list-style-type: none"> <li>a. Application retries to connect with the internet or show no internet access message.</li> </ol> </li> <li>2. The selected friend is not active <ol style="list-style-type: none"> <li>a. The application show friend is not active now message to use and return on the friend list page.</li> </ol> </li> </ol>  |
| Postcondition        | <ol style="list-style-type: none"> <li>1. Successful completion<br/>The user account shares information with his/her fiend.</li> <li>2. Failure condition: - The device has not video recording capability.</li> </ol>   |

Table 4-6 Use case description - Sign translation

|                      |  |
|----------------------|--|
| Use case id          | UC_06  |
| Use case title       | Sign translation   |
| Use case description | This use case describes how the user uses this application to translate its hand gesture to text.  |
| Actors               | ○ Learner  |
| Precondition         | 1. The application opens in a web browser  |
| Basic flow of events | <ol style="list-style-type: none"> <li>1. This use case begins when the learner opens the application.</li> <li>2. The application shows an alternative to learn, exercise, make a video call and Sign translation. But in this case, the user always selects Sign translation.</li> <li>3. The application starts the camera.</li> <li>4. Users start signing (signal it sings).</li> <li>5. The application translates its hand gesture to text automatically.</li> <li>6. Application display text equivalent of user sign.</li> <li>7. This use case ends successfully.</li> </ol> |
| Alternative flow     | <p>Step [3]: If the device hasn't camera</p> <ol style="list-style-type: none"> <li>1. The application displays error message (unable to open camera)</li> </ol> <p>Step [5] if the application doesn't get a match of hand gesture equivalent</p> <ol style="list-style-type: none"> <li>1. The application shows an unknown sign</li> </ol>  |
| Postcondition        | 1. Users can translate hand gestures to text automatically.  |



Table 4-7 Use case description - Add new sign

|                      |   |
|----------------------|---|
| Use case id          | UC_07   |
| Use case title       | Add new sign  |
| Use case description | This use case describes how instructor uses this application to add a new sign to improve Ethiopian sign language learning application.   |
| Actors               | ● Instructor  |
| Precondition         | <ol style="list-style-type: none"> <li>1. The instructor should have a valid account and login to the application.</li> <li>2. There must be an active internet connection.</li> <li>3. Computer to use web application.</li> </ol>   |
| Basic flow of events | <ol style="list-style-type: none"> <li>1. This use case begins when instructor open and login to the application.</li> <li>2. Instructor select add new sign.</li> <li>3. The application show form to enter text representation, description [optional] and its associated sign.</li> <li>4. Instructor fills all the required fields.</li> <li>5. Instructor click add button.</li> <li>6. Sign language representation of the word (the image or gif) checked for validity and maximum file size, and word checked for uniqueness to avoid duplication (already added word can't be overwritten).</li> <li>7. The application shows a success message.</li> <li>8. This use case ends successfully.</li> </ol> |
| Alternative flow     | <p>Step [6]: If the entered information is a validation error</p> <ol style="list-style-type: none"> <li>1. The system doesn't add a new sign.</li> <li>2. The system informs the user to fills all required fields correctly.</li> <li>3. The use case resumes at step 4 of the main sequences.</li> </ol>   |

|               |   |
|---------------|---|
| Postcondition | <ol style="list-style-type: none"> <li>1. New sign successfully added to Ethiopian sign learning application database.</li> <li>2. Error message display for any validation error.</li> </ol> |
|---------------|---|

#### 4.1.1.3. Use case scenario

A use case scenario is a single path through the use case. A scenario is an instance of a use case describing a concrete set of actions. Scenarios are used as examples for illustrating common cases or describes a real-world example of how one or more people or organizations interact with a system. They describe the steps, events, and/or actions that occur during the interaction. Their focus is on understandability. The following Scenarios describe the interaction between the user and the Ethiopian sign language learning application.

##### 1. Login Scenario

Scenario name: Login

Participant actor: System admin, instructor, and learner

When all user wants to login, first he/she must install the Ethiopian sign language learning application on his/her mobile device or he/she open the web application in the browser and have a valid username and password. Then user click login button the system display login form to enter his/her username and password after that user fill all required field the system validates for validity and existence if pass (ok) user redirected to his/her personal homepage and access all private information from Ethiopian sign language learning application (like make video call, ability to add new sign if he/she is instructor).

##### 2. Register Scenario

Scenario name: Register

Participant actor instances: Instructor and learner

If sign language instructor or learner as a user of the proposed system wants to register, first he/she must install the Ethiopian sign language learning application on his/her mobile device or he/she

open the web application in the browser. Then sign language instructor or learner click register button the system display register form to enter his/her personal information after that he/she fill all required field and press register button then the system validates for validity and if it's ok to sign language instructor or learner successfully create an account to Ethiopian sign language learning application to login and make video call with his fiend or to add new sign to the application if he/she is an instructor.

### **3. Learn Scenario**

Scenario name: Learn

Participant actor instances: Learner

If sign language learner as a user of the proposed system wants to learn Ethiopian sign language, first he/she must install the Ethiopian sign language learning application on his/her mobile device or open the web application in the browser. The system displays options (menu) and learner select learn option from a menu, then the system display choice to select a level of learning (basic, intermediate, advance) learner select his level and after that the system display sign language related to the selected level of learning. Then learners practice sign language by scrolling to get all signs in the selected level and watching the image associated with Amharic or English text and playing the gif for the sign (if any).

### **4. Exercise Scenario**

Scenario name: Exercise

Participant actor instances: Learner

Sign language learner as a user of the proposed system wants to exercise Ethiopian sign language, first, he/she must install the Ethiopian sign language learning application on his/her mobile device or open the web application in the browser. The system displays options (menu) and learner select exercise option from a menu, then the system display choice to select level of exercise (basic, intermediate, advance) learner select his/her level and after that the system display sign language exercise related to the selected level of sign. Then sign language learner practice sign language by

choosing answer Amharic or English text for given sign image and playing the gif for the sign (if any). After choosing the answer learner click next button to get the next question. If the answer for the given questions is not correct the system displays the wrong answer message and stays that questions. At the end of the quiz, learner can see the total mark for the quiz can reset and restart the quiz again and again over.

### **5. Video call Scenario**

Scenario name: Make a video call

Participant actor instances: Learner

Sign language learner as a user of the proposed system wants to make a video with his/her friends using Ethiopian sign language learning application, first, he/she must install the Ethiopian sign language learning application on mobile device or open the web application in the browser. The system display options (menu) and learner first must login to the system using his/her username and password (if he hadn't he can sign up and create new account) after login select make video call option from menu, then the system display list of friends to make video call with their online status (active or not) then learner select one friend at a time to make video calls and the system open camera to record his hand gesture and the system also display his friends video if he/she is online start communicating. If learner select inactive friends the system show can't make a video call with an offline user and stay in the list of friend's page and also in case of learner uses a device that hasn't camera the system also shows an error message and stay in the list of friend's page.

### **6. Sign translation Scenario**

Scenario name: Sign translation

Participant actor instances: Learner

Learner as a user of the proposed system wants to sign translation to automatically translate into Amharic or English text using Ethiopian sign language learning application, first, he/she must install the Ethiopian sign language learning application on his/her mobile device or open the web application in the browser. The system displays options (menu) and the learner selects Sign

translation option from the menu, then the system opens camera and starts recording his/her hand gesture. Then the system automatically translates learner hand gesture to Amharic or English text and displays the associated text if it gets match or unknown sign for complex (untrained) hand gesture (sign).

### **7. Add new sign Scenario**

Scenario name: Add new sign

Participant actor instances: Instructor

Instructor as a user of the proposed system wants to add a new sign to Ethiopian sign language learning application, first, he/she must install the Ethiopian sign language learning application on his mobile device or open the web application in the browser. Sign language instructors must first register as an instructor and have a valid account to login. After the sign language instructor logged to the system the system displays options (menu) and sign language instructor select add new sign from menu, then the system display add new sign form he/she fills all the required fields, the system validates the field (like maximum size for image or gif) and add to Ethiopian sign language learning application database for approval by administrator. Sign only approved (checked for validity) by an administrator is visible for the learner.

### **8. Update profiles Scenario**

Scenario name: Update profile

Participant actor instances: Learner, instructor and system administrator

When all user of the proposed system wants to update his/her profiles on Ethiopian sign language learning application, first he/she must install the Ethiopian sign language learning application on his mobile device or open the web application in the browser. After opening the application user must login to the system. Then the system displays options (menu) and user selects an update profile option from the menu, the system display profile form filled with his current profile information's and user changes any fields as he/she wants and click save button, the system updates the user profiles in databases.

## 4.2. Object model

An object model is a logical interface, software or system that is modeled through the use of object-oriented techniques. It enables the creation of an architectural software or system model prior to development or programming.

An object model helps describe or define a software/system in terms of objects and classes. It defines the interfaces or interactions between different models, inheritance, encapsulation and other object-oriented interfaces and features. In the proposed system we use object models like class diagram and data dictionary.

### 4.2.1. Class diagram

A class consists of a collection of attributes and methods that determine the state and the behavior of its instances (objects). Classes are connected to each other by associations and inheritance relations. The class name identifies the class.

Class diagram is basically a graphical representation of the static view of the system and represents different aspects of the application. A collection of class diagrams represents the whole system.

The following points should be remembered while drawing a class diagram:

- The name of the class diagram should be meaningful to describe the aspect of the system.
- Each element and their relationships should be identified in advance.
- Responsibility (attributes and methods) of each class should be clearly identified
- For each class, minimum number of properties should be specified, as unnecessary properties will make the diagram complicated.
- Use notes whenever required to describe some aspect of the diagram. At the end of the drawing, it should be understandable to the developer/coder.
- Finally, before making the final version, the diagram should be drawn on plain paper and reworked as many times as possible to make it correct.

Therefore, following this, we identify classes such as user-class (learner, instructor, administrator), dictionary-class (basic, intermediate, advance) and video call-class for the proposed system.

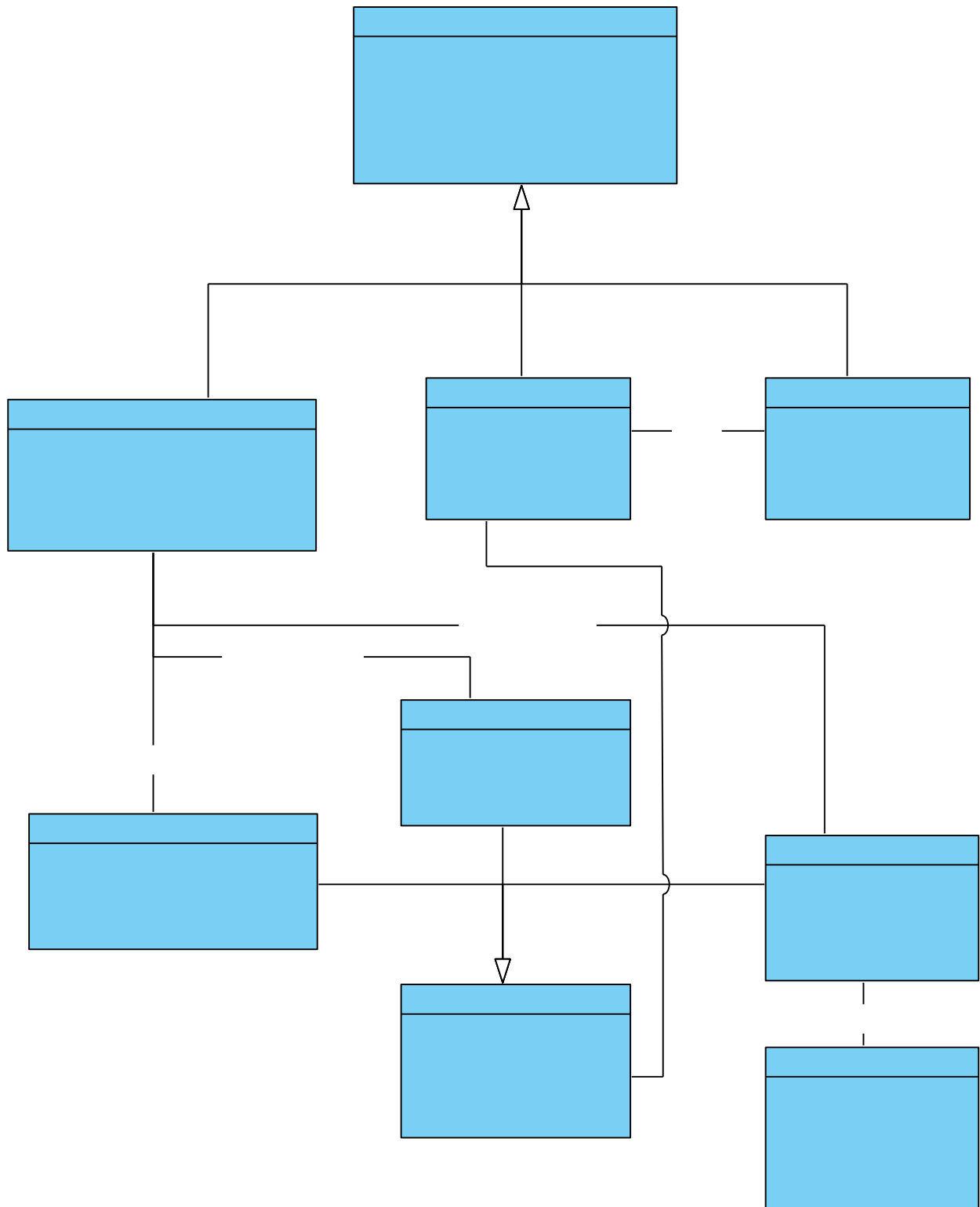


Figure 4-4 Analysis level class diagram

#### 4.2.2. Data dictionary

A data dictionary is a collection of descriptions of the data objects or items in a data model for the benefit of programmers and others who need to refer to them. A first step in analyzing a system of objects with which users interact is to identify each object and its relationship to other objects. This process is called data modeling and results in a picture of object relationships. After each data object or item is given a descriptive name, its relationship is described (or it becomes part of some structure that implicitly describes relationship), the type of data (such as text or image or binary value) is described, possible predefined values are listed, and a brief textual description is provided. This collection can be organized for reference in a book called a data dictionary. The data dictionary is a description of the database table of the system such as functionality, data types, data size, contents item and table relations. In the proposed system we describe the data dictionary in the following tables.

**Table name:** User

**Primary key:** User id

**Description:** Used to store account information for the user

*Table 4-8 User data dictionary*

| Field name    | Data type    | Constraints | description  |
|---------------|--------------|-------------|--|
| User id       | Int          | Primary key | Unique identification of the user for the developer (auto increments)                                    |
| User name     | Varchar (50) | Not null    | Unique user name for each user used to login to make a video call with his/her friend or add a new sign. |
| User password | Varchar (50) | Not null    | Associated password for each user  |
| Register date | Data time    | Not null    | Date of registration for each user   |
| Login status  | Boolean      | Not null    | The status of the user (active-True or in active-False)  |



**Table name:** Instructor

**Primary key:** Instructor id

**Description:** Used to store instructor information

*Table 4-9 Instructor data dictionary*

| Field name    | Data type     | Constraints | description   |
|---------------|---------------|-------------|---|
| Instructor id | Int           | Primary key | Unique identification of instructor   |
| Description   | Varchar (500) | Not null    | Instruction self-description maximum of 150 words required in order to register as an instructor of sign language to add new signs. |
| Phone number  | Int           | Not null    | Used to store the phone number of instructors.  |
| User Id       | Int           | Foreign key | Unique identification of instructor as a user.  |

**Table name:** Dictionary

**Primary key:** Sign id

**Description:** Used to store sign language image location and associated text for each sign

*Table 4-10 Dictionary data dictionary*

| Field name    | Data type    | Constraints | description                   |
|---------------|--------------|-------------|-------------------------------|
| Sign id       | Int          | Primary key | Unique identification of sign |
| Sign location | Varchar (50) | Not null    | The image location for sign   |
| Text          | Varchar (50) | Not null    | The associated text for sign  |

**Table name:** Intermediate

**Description:** Used to store intermediate sign language image location and associated text for each sign.

*Table 4-11 Intermediate data dictionary*

| Field name  | Data type     | Constraints | description  |
|-------------|---------------|-------------|--|
| Sign id     | Int           | Foreign key | Unique identification of signs in the dictionary.          |
| Description | Varchar (250) | Not null    | The brief description of the sign for extra understanding. |

### 4.3. Dynamic model

The dynamic model represents the time-dependent aspects of a system. It is concerned with the temporal changes in the states of the objects in a system. In this section, we document the behavior of the object model, in terms of sequence, activity and state chart diagrams.

#### 4.3.1. Sequence diagram

In this section, we illustrate (diagrammatically) a sequential logic, in effect, and the time ordering of messages. From a business process perspective. How the business model is executed? A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. The proposed system sequence diagram for each use cases are as follows: -

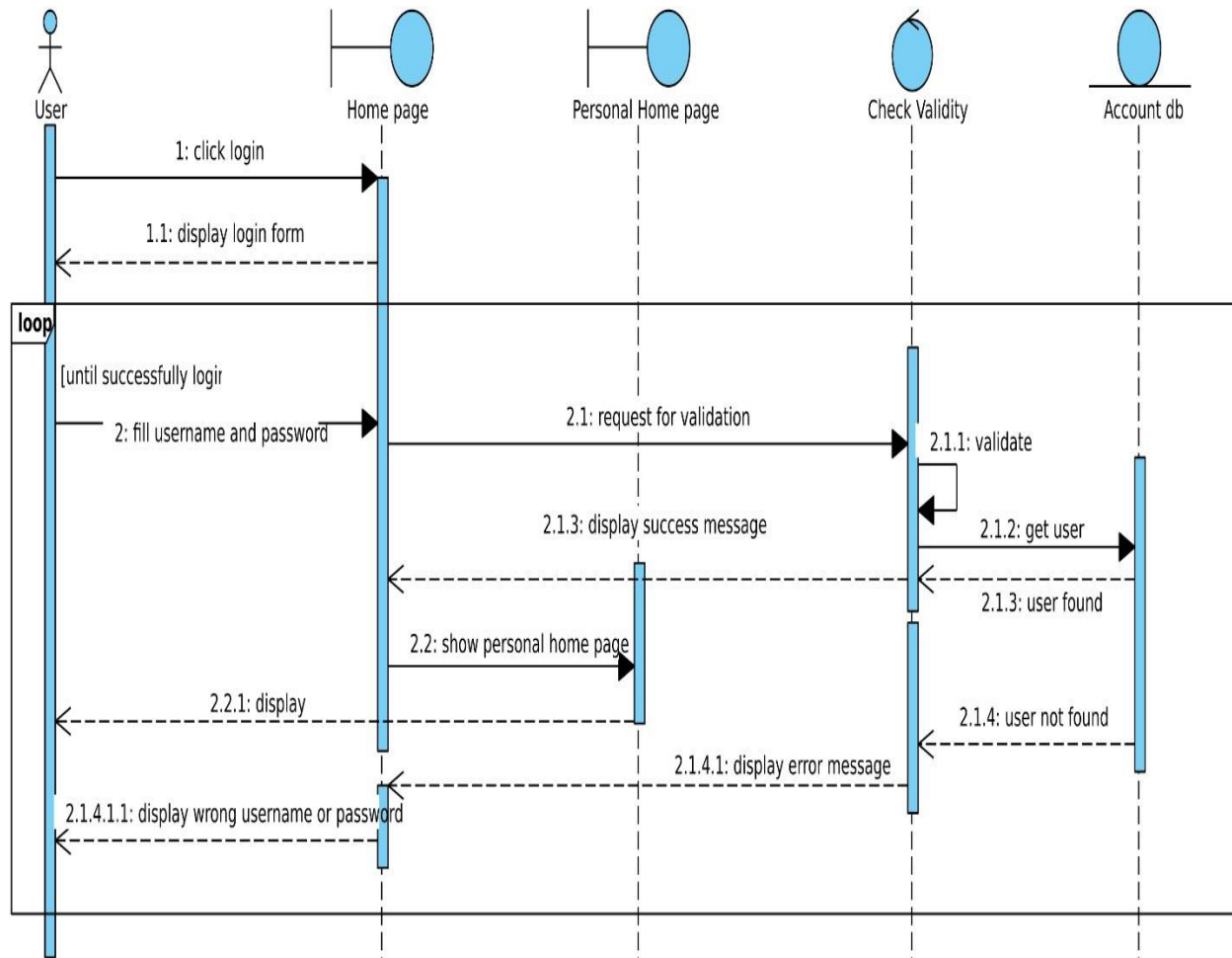


Figure 4-5 Sequence diagram for login

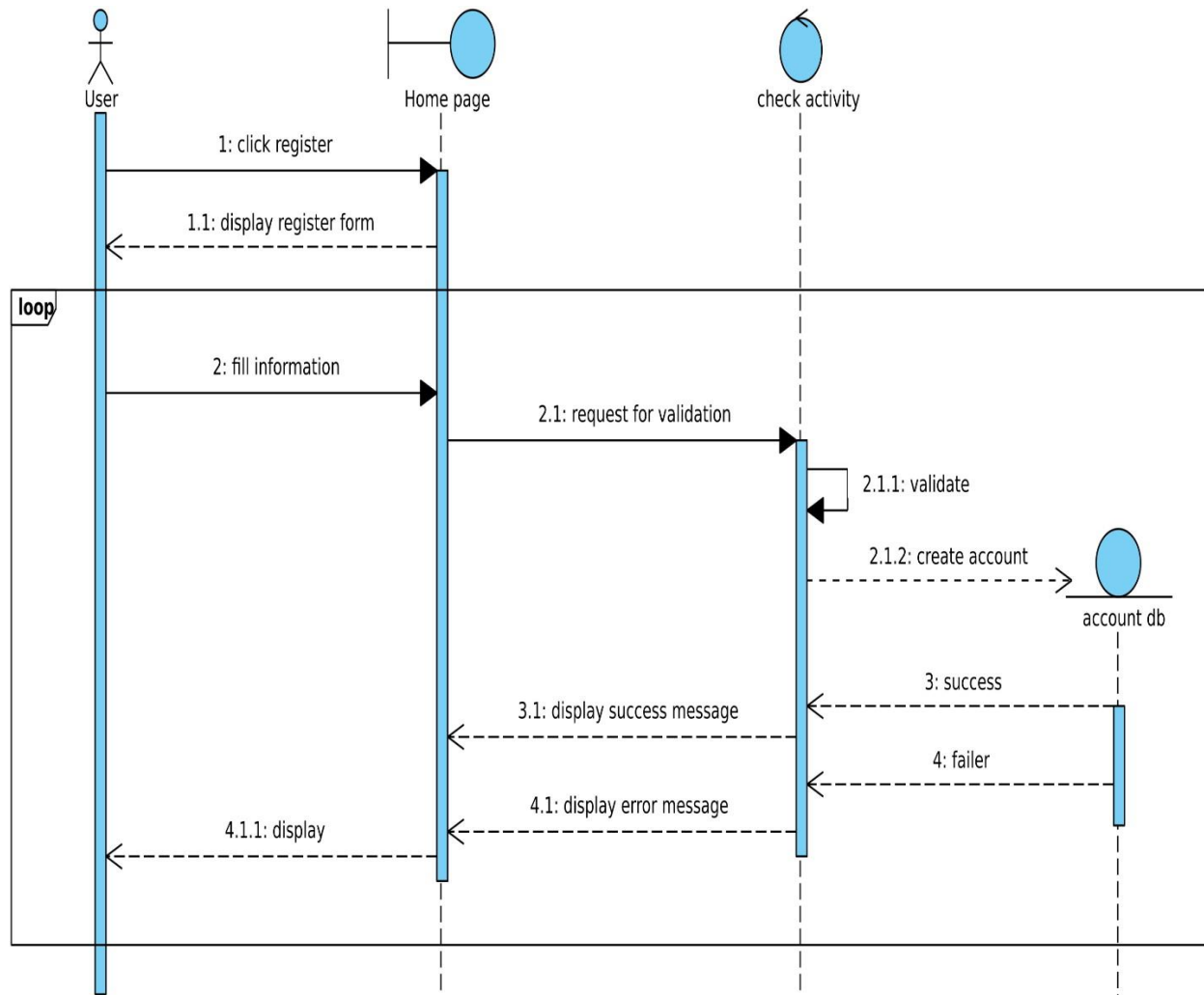


Figure 4-6 Sequence diagram for register

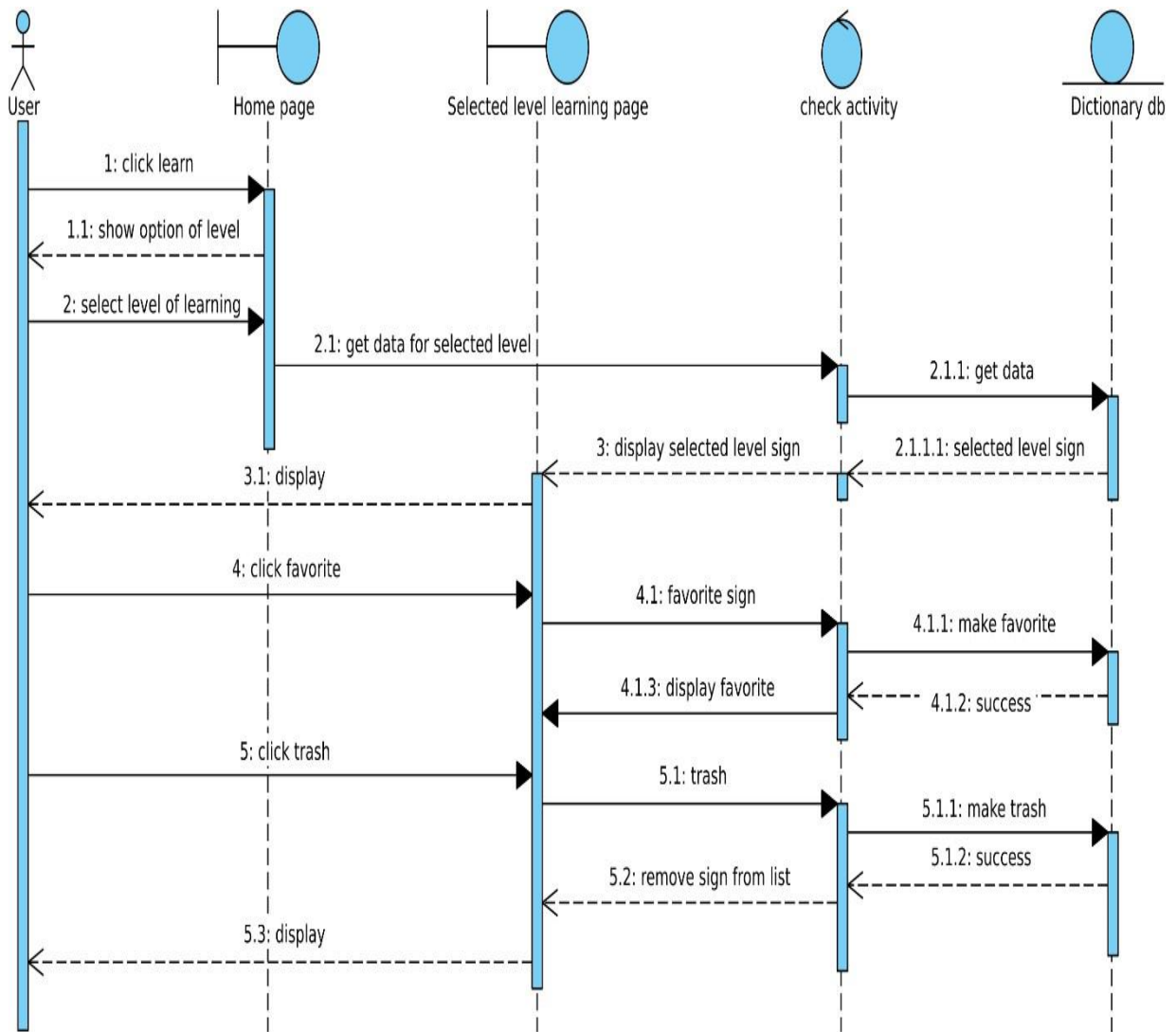


Figure 4-7 Sequence diagram for learn

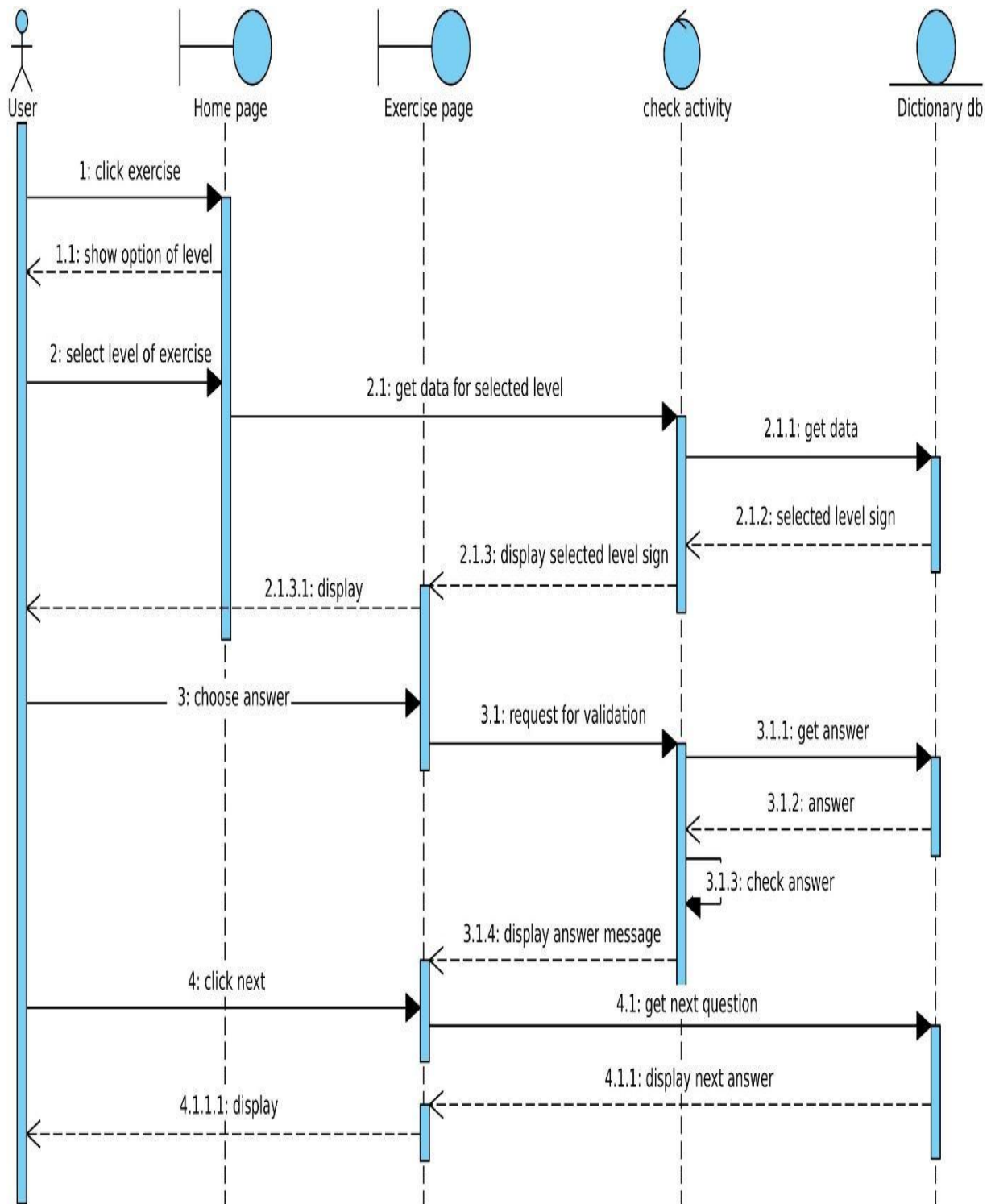


Figure 4-8 Sequence diagram for exercise

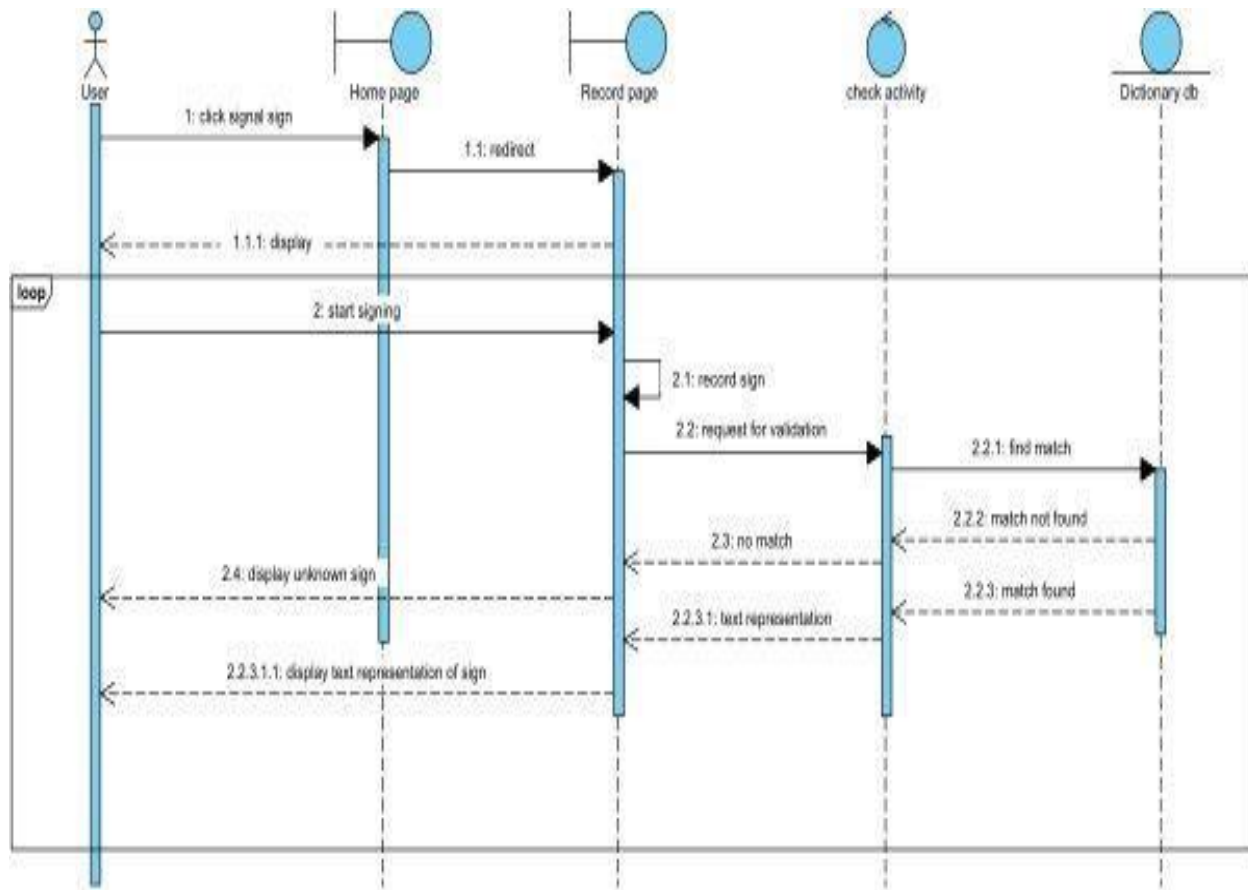


Figure 4-9 Sequence diagram for Sign translation

#### 4.3.2. Activity diagram

The activity diagram is a flow chart to represent the flow from one activity to another activity. It describes how activities are coordinated to provide a service. UML activity diagrams are used to document the logic of a single operation/method, a single-use case, or the flow of logic of a business process. The activity diagram used to describe the action performed by the actor and what the system responds to the actor.

In this section, we illustrate graphical representations of workflows of stepwise activities and actions with support for choice, iteration, and concurrency in the following figures.

Figure 4.10 below activity diagram describes how administrator login and to approve instructor, check the validity of new sign, or update his/her profiles.

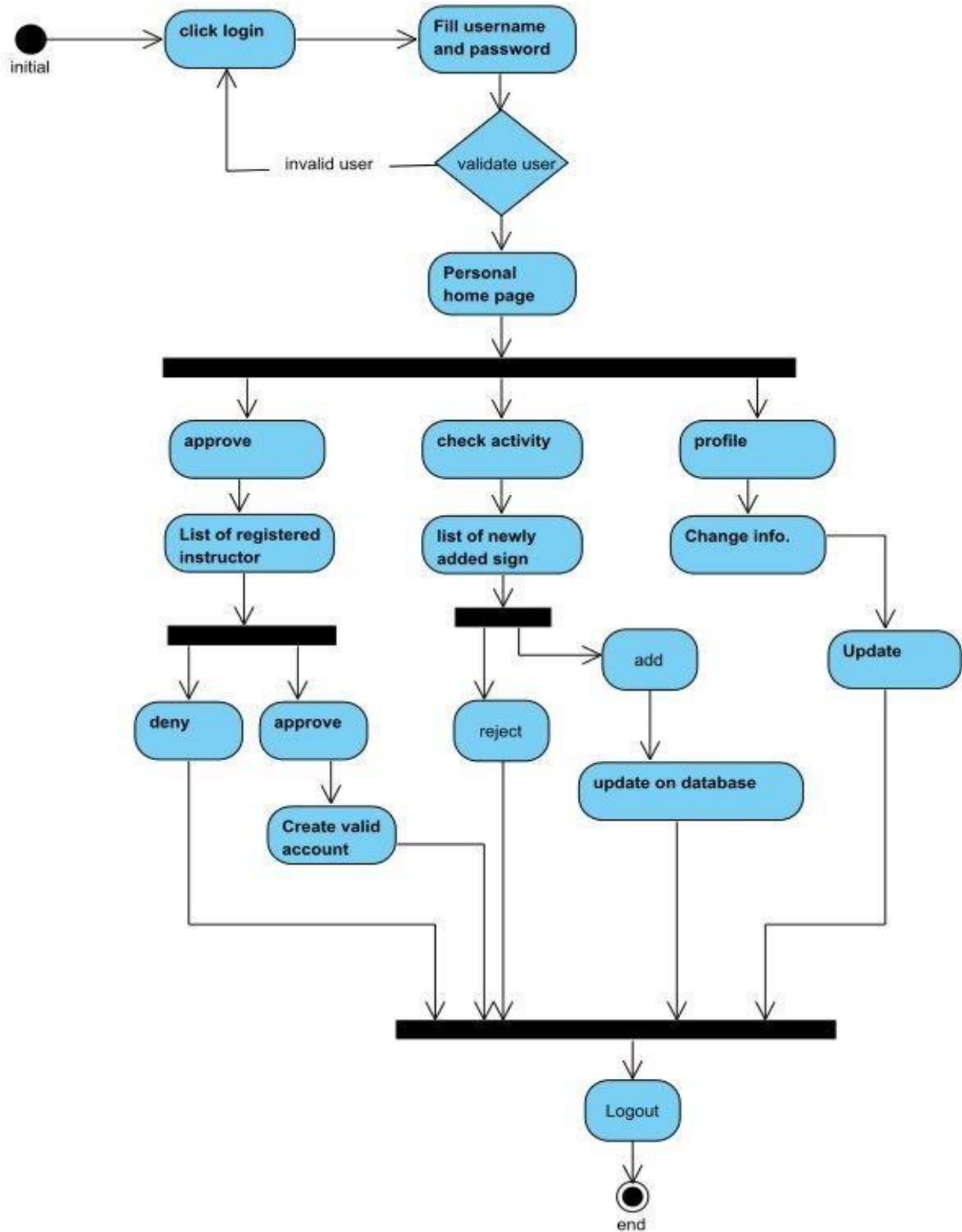


Figure 4-10 Administrator activity diagram



Figure 4.11 below activity diagram describes how an instructor learn (trash, favorite, search for a specific sign using its text representation) or login to the application in order to add a new sign.

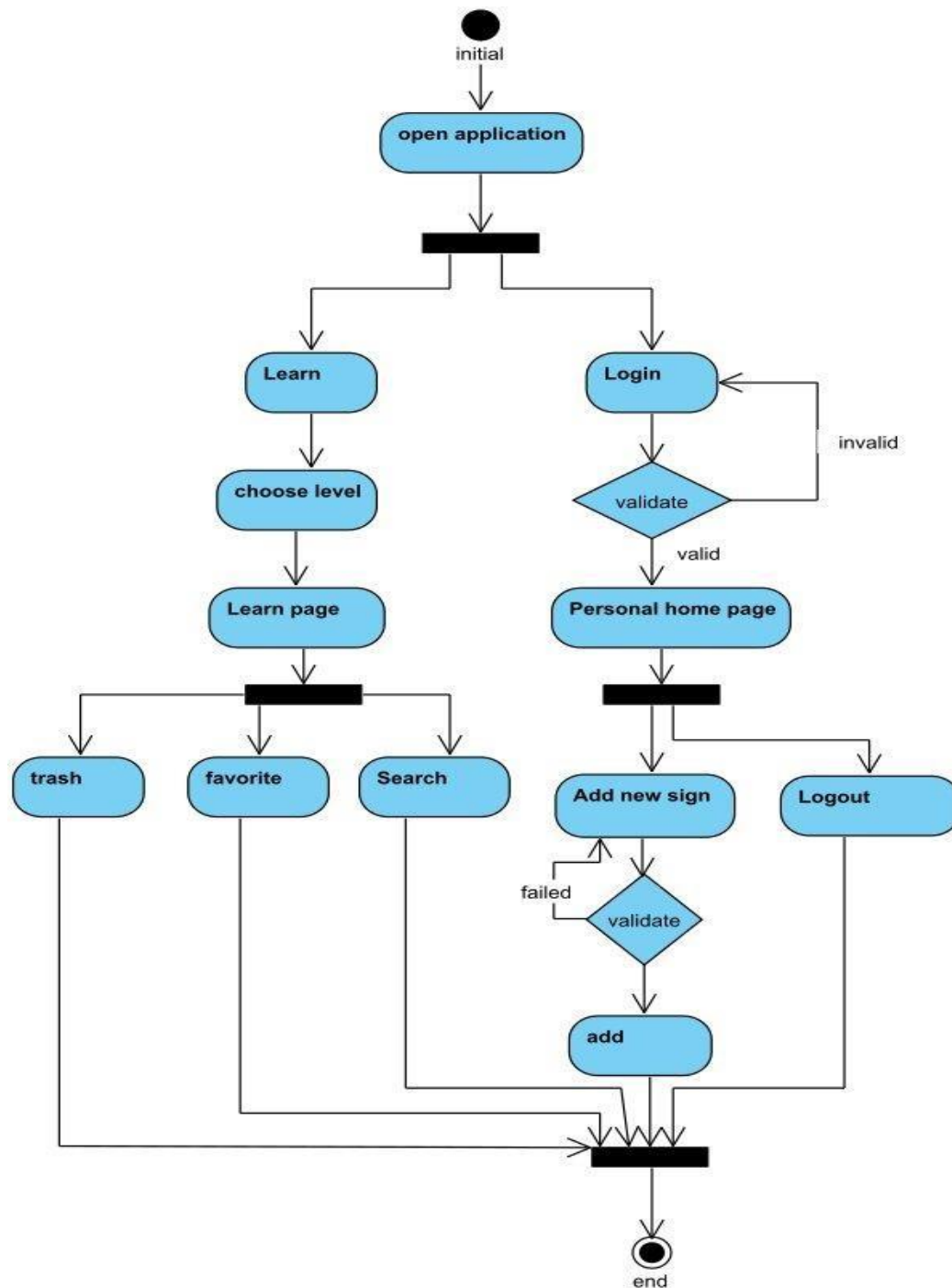


Figure 4-11 Instructor activity diagram

Figure 4.12 below the activity diagram describes how learner registers (to make a video call with his/her friends), login, learn (trash, favorite, search for specific sign), exercise and Sign translation to automatically translate with text.

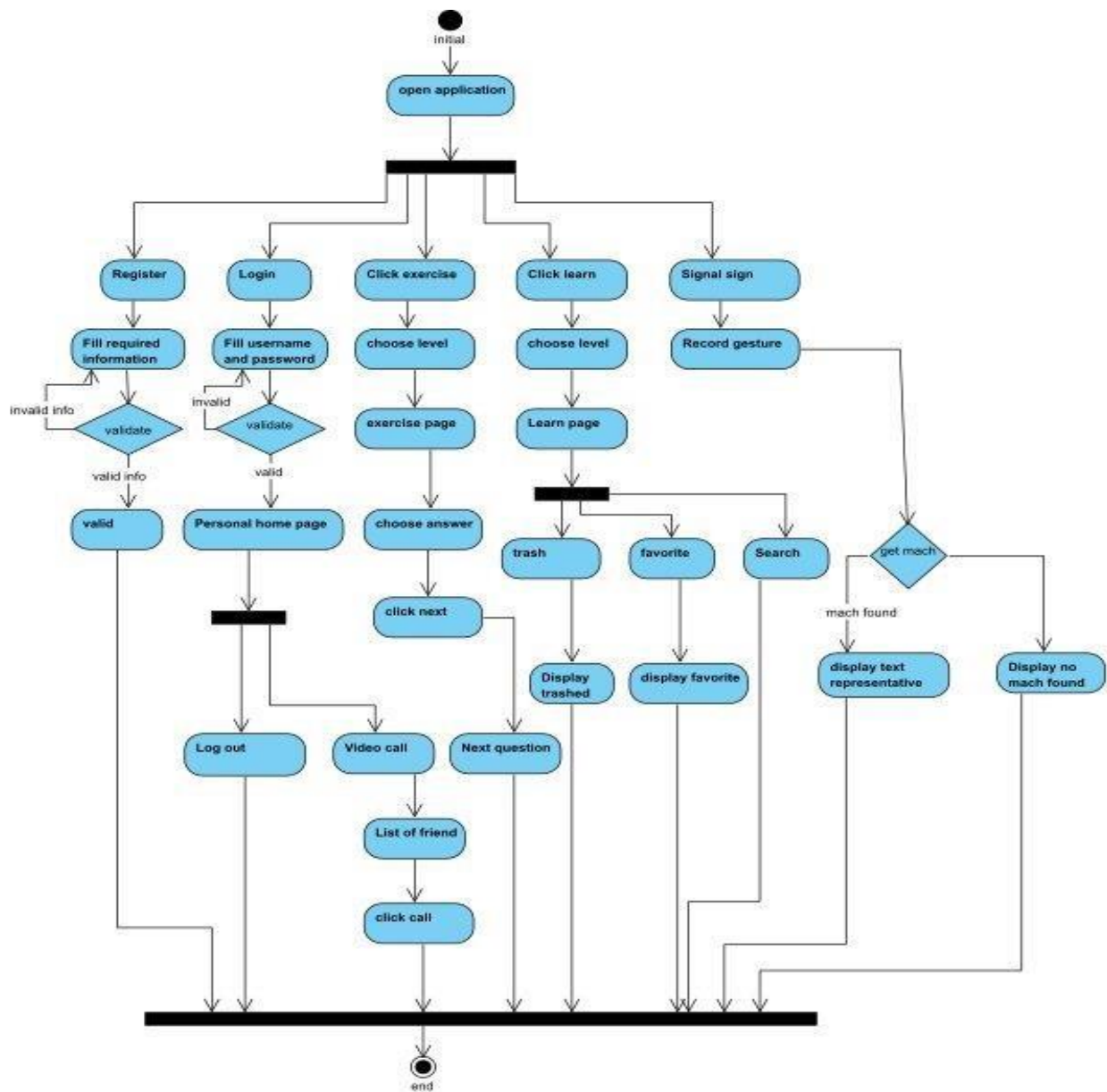


Figure 4-12 Learner activity diagram

### 4.3.3. Statechart diagram

A state diagram is a type of UML diagram used to represent the behavior of nontrivial objects. State diagrams require that the system described is composed of a finite number of states. They define different states of an object during its lifetime and these states are changed by events. Statechart diagrams are useful to model reactive systems. Reactive systems can be defined as a system that responds to external or internal events. Statechart diagram describes the flow of control from one state to another state. The common model elements that state chart diagrams contain are:

- States
- Start and end states
- Transitions
- Entry, do, and exit actions
- Registration state machine diagram

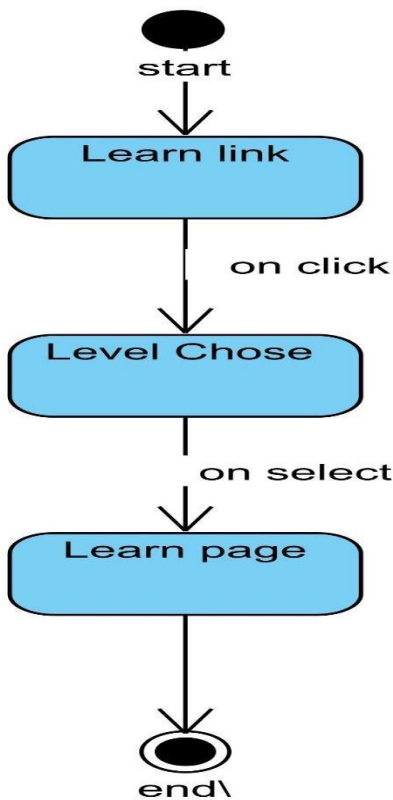


Figure 4-13 Learn state chart diagram

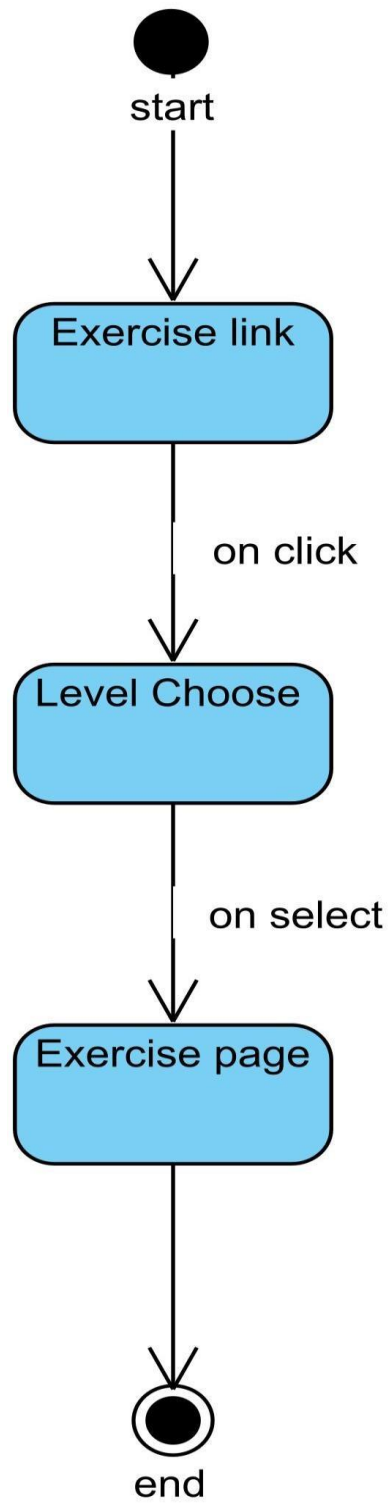


Figure 4-14 Exercise state chart diagram

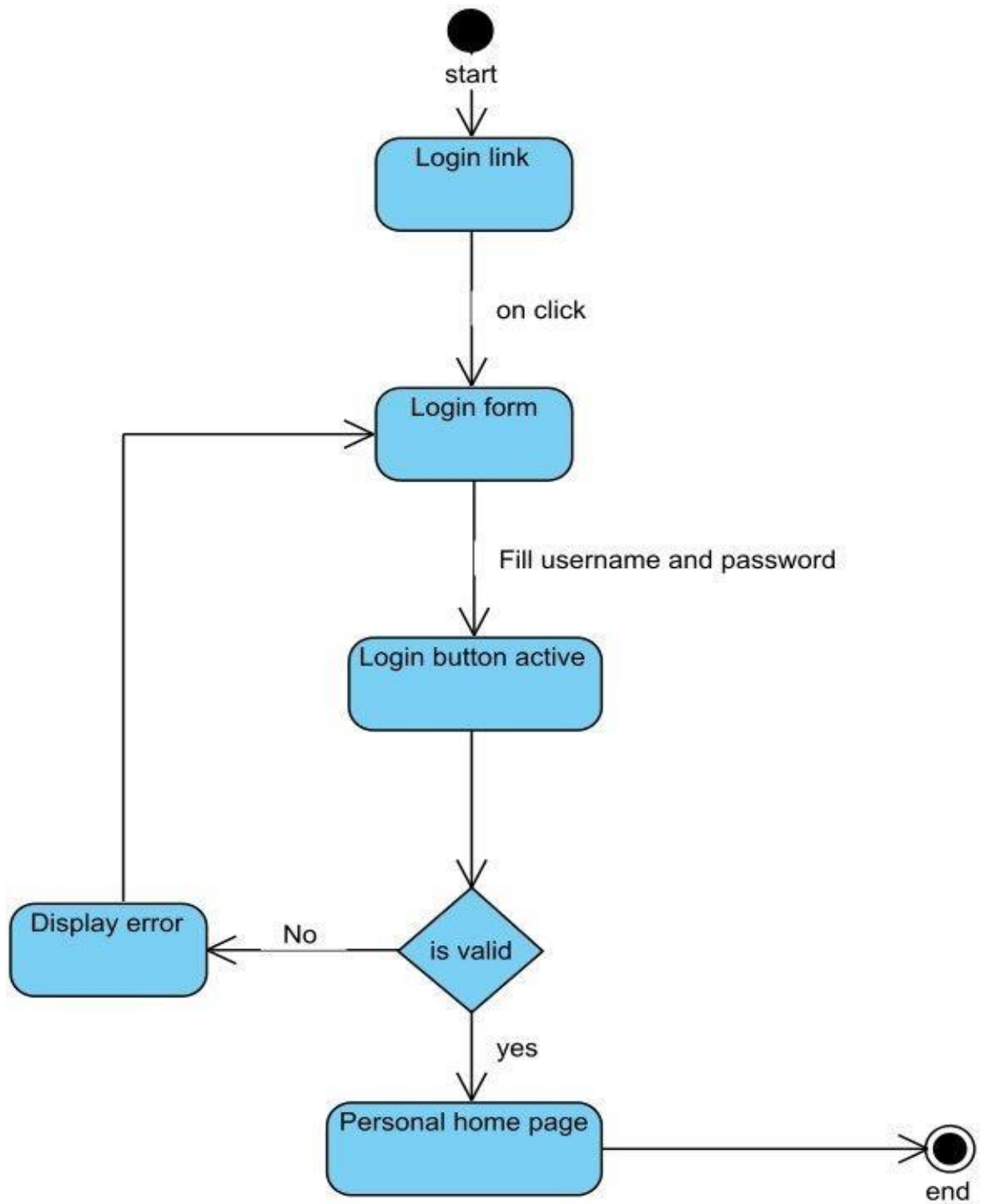


Figure 4-15 Login state chart diagram

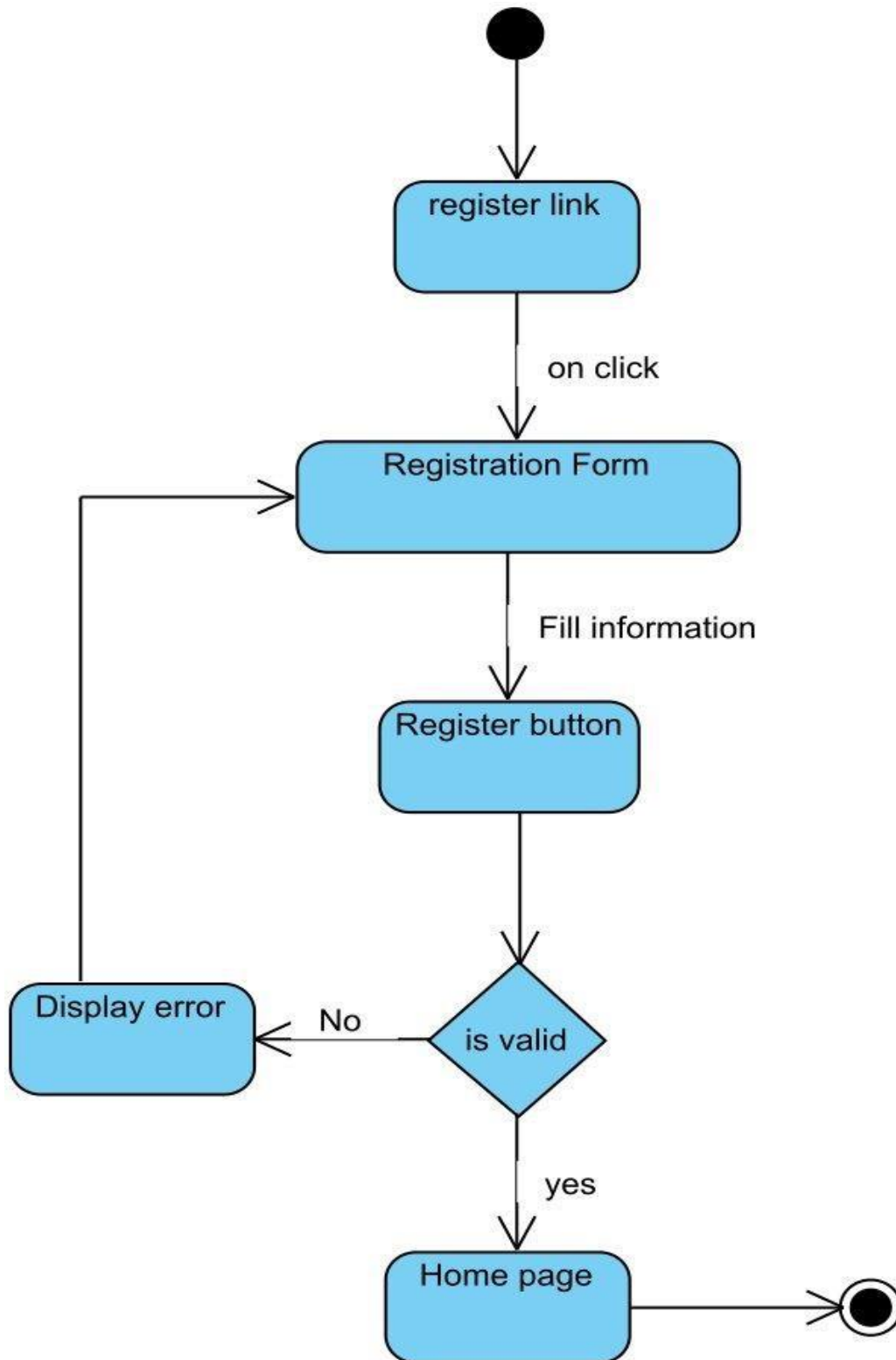


Figure 4-16 Register state chart diagram

## CHAPTER FIVE

### 5. SYSTEM DESIGN

In this chapter, we provide a brief overview of the design goals, design trade-off, current and proposed software architecture, Hardware/software mapping, Persistent data management, and Access control and security to the proposed system.

#### 5.1. Design goal

The Design Goals specify the qualities of the system that are derived from the nonfunctional requirement and should be achieved and addressed during the design of the system. Design goals are targets for design work. These are typically agreed upon by stakeholders as the criteria for comparing design alternatives and evaluating design outcomes. [3]. The major design goal of the system includes: -

- **Performance:** - Users are becoming increasingly unwilling to wait for the page to load; therefore, the system has good performance as much as possible this is attained via easily loadable interface components and fast response time in make searching, updating, deleting, inserting and viewing information from the database. Furthermore, the system should not be taking up too much space in memory only some sign language is loaded at first and other sign language data is downloaded as learner need.
- **User interface and human factors:** - The interface of the proposed system provides easy to understand, easy to use and user-friendly interface and users of the system easily use and perform their task. Everything stems from knowing our users, including understanding their goals, skills, preferences, and tendencies. Once we know about our users, we can develop a system that satisfies our user needs that is done.
- **Maintenance:** - In the time of failure or need modification the system needs to be maintained. To be maintainable the system will meet the following maintenance criteria.
  - ✓ **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.

- ✓ **Modifiability:** - If in the system, some functionality requires to be modified, this modification must be done specifically to that function or page without affecting the overall system organization.
- **Dependability:** - It can be defined as the probability that a component either does not fail or fails and will be repaired within an allowable time interval. The proposed system component can work independently mean for example if the learning component fails the exercise, video call or sign translation can work effectively.
- **Hardware consideration:** - The proposed system will not require any special hardware.  
  
The mobile application will install on any android based mobile device and the web application is accessible in any machine having a browser.

## 5.2. Design Trade-offs

A tradeoff is a situation that involves losing one quality or aspect of something in return for gaining another quality or aspect. We list the following trade-off: -

- **Performance vs. Security:** - We don't use a complex security system that affects the proposed system performance. Only we use simple encryption for username and password, and Django framework for back end web development for more security that doesn't more affect the proposed system performance. We lose some advance security algorithms to increase the proposed system performance because we don't have much sensitive user data to care about.
- **Performance vs. Usability:** - When we compare performance with the usability of the system is easy to learn and operate. The User interface for this system will be simple and clear, and we decrease the size of the image to load in time and increase performance. We use native android so that we have great performance the disadvantage of this is only a limited widget we overcome this by customizing own style of the user interface.
- **Security vs. Availability:** - User can access and use the download file offline mean the system is available all the time and this download file is only accessible from his/her device mean doesn't affect the proposed system security unless he/she lost their device and for we're not responsible.



### 5.3. Current system architecture

Currently, there is no system architecture because sign language learning is carried out manually. The instructor of sign language prepares his/her self on their class, and teaches the learner by signing the sign and writing the text representation or drawing the image if possible and the learner learns by watching his/her instructor's gesture and text or image drawn.

### 5.4. Proposed system architecture

Systems Architecture is a generic discipline to handle objects (existing or to be created) called "systems", in a way that supports reasoning about the structural properties of these objects. Systems Architecture is a response to the conceptual and practical difficulties of the description and the design of complex systems. [4]

The proposed system is consisting of 3-tier architecture namely presentation layer, business logic layer, and the data layer.

- **The presentation layer:** - Is client layer and topmost layer of the application. This is the layer users see when he/she use this system. It is the interface to the proposed system which takes information from the user. The main function of this layer is to communicate with the application layer. This layer passes the information given in terms of keyboard action and mouse click to the business logic layer.
- **Business logic layer:** - It is an application layer that interacts with the data layer and sends information retrieved from the database toward to presentation layer. It acts as the mediator between the presentation layer and the data layer. For example, if user selects learn in presentation layer the business logic layer is responsible to fetch all related signs from the database (data layer) and send information towards to presentation layer.
- **The data layer:** - Which is used to store all levels of signs, its associated text and user information. In general users of the proposed system can access some signs without any interaction with the data layer using his/her Ethiopian sign language learning application installed on his/her mobile devices offline. But if user wants to get more sign from this application, he/she uses to connect with data and download extra sign or use browser to access the system using the internet. In this case when the user enters input and takes certain action

application server process client request to interact with the database server. Figure 5.1 below is the proposed system architecture.

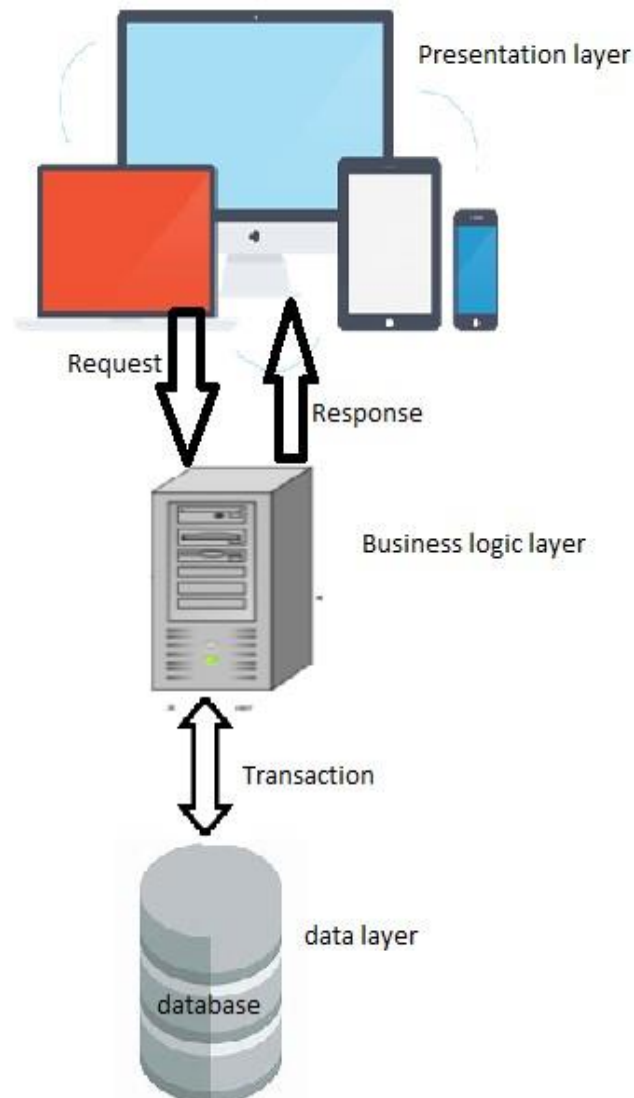


Figure 5-1 Proposed system architecture

#### 5.4.1. Subsystem decomposition and description

Subsystem decomposition is the process of dividing or decomposing the system into manageable and smaller subsystems from the analysis model of the proposed system. The goal of the system decomposition is to reduce the complexity of the design model and to distribute the class of the system into large scale and cohesive components. Components are generally units of computation

or data stores in the system. A component has a name, which is generally chosen to represent the role of the component or the function it performs. The different components of a system are likely to interact while the system is in operation to provide the service expected of the system. The purpose of this activity is to divide the system into self-contained components that can be managed by individuals. As we address other design issues, such as access control and persistency management, we will refine or modify this initial subsystem decomposition. In the proposed system we draw the following subsystem decomposition diagram based on the functional requirement of the system.

Figure 5.2 below shows the proposed system subsystem decomposition in the component diagram.

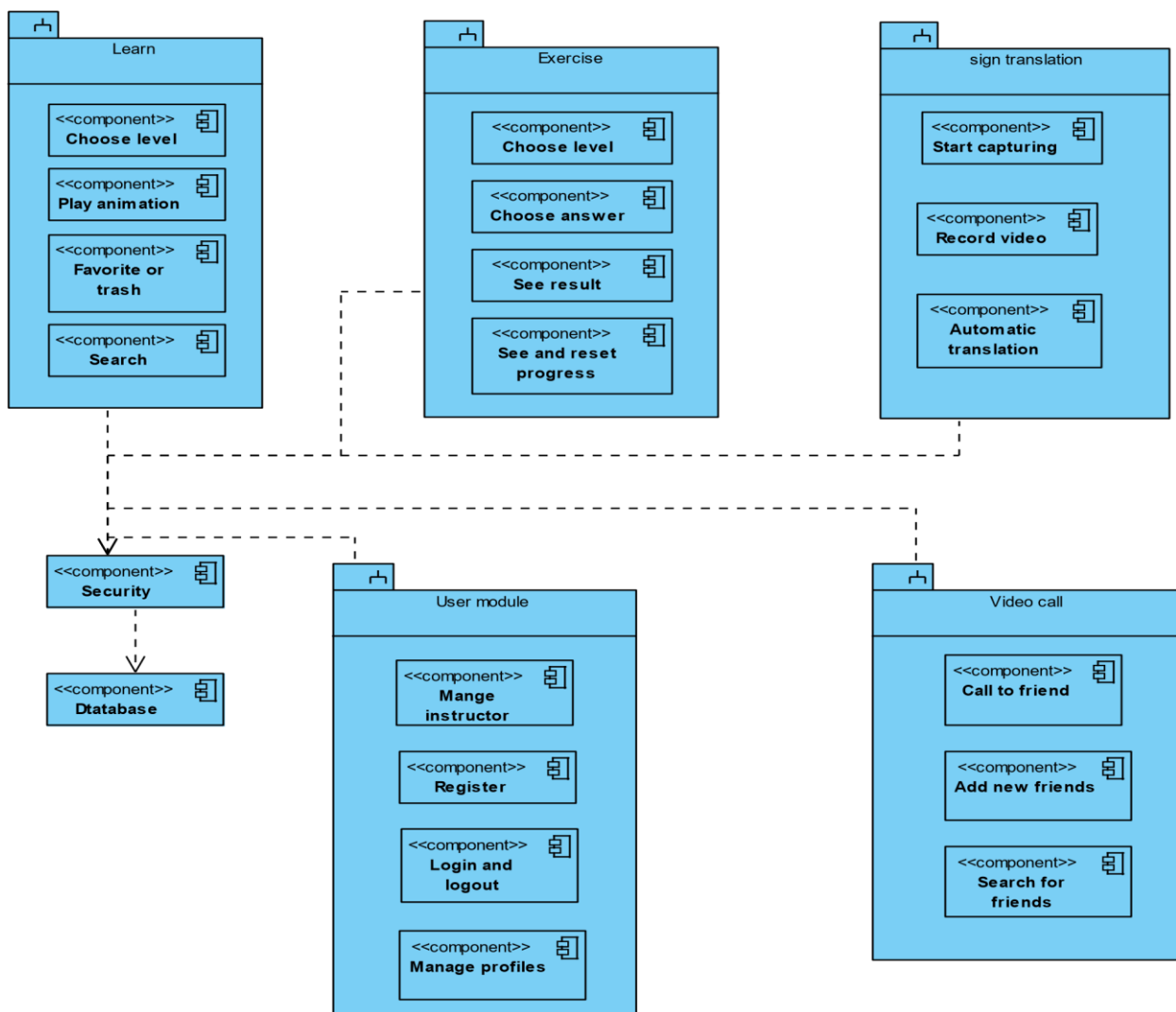


Figure 5-2 Subsystem decomposition

### 5.4.2. Hardware/Software mapping

The hardware-software mapping is described to indicate the various hardware devices and equipment used in the system and its interaction with the software components. The deployment diagram depicts a static view of the run-time configuration of processing nodes and the components that run on those nodes. In other words, deployment diagrams show the hardware for your system, the software that is installed on that hardware, and the middleware used to connect the disparate machines to one another. If you want to create a deployment diagram for applications that are deployed to several machines. It also shows how the software and hardware components work together. Deployment diagram used to show the hardware of the system, the software that is installed in the hardware and also the middleware that is used to connect the disparate machines to one and another. It also shows how the software and hardware components of the system work together.

Figure 5.3 below shows the deployment diagram of the proposed system.

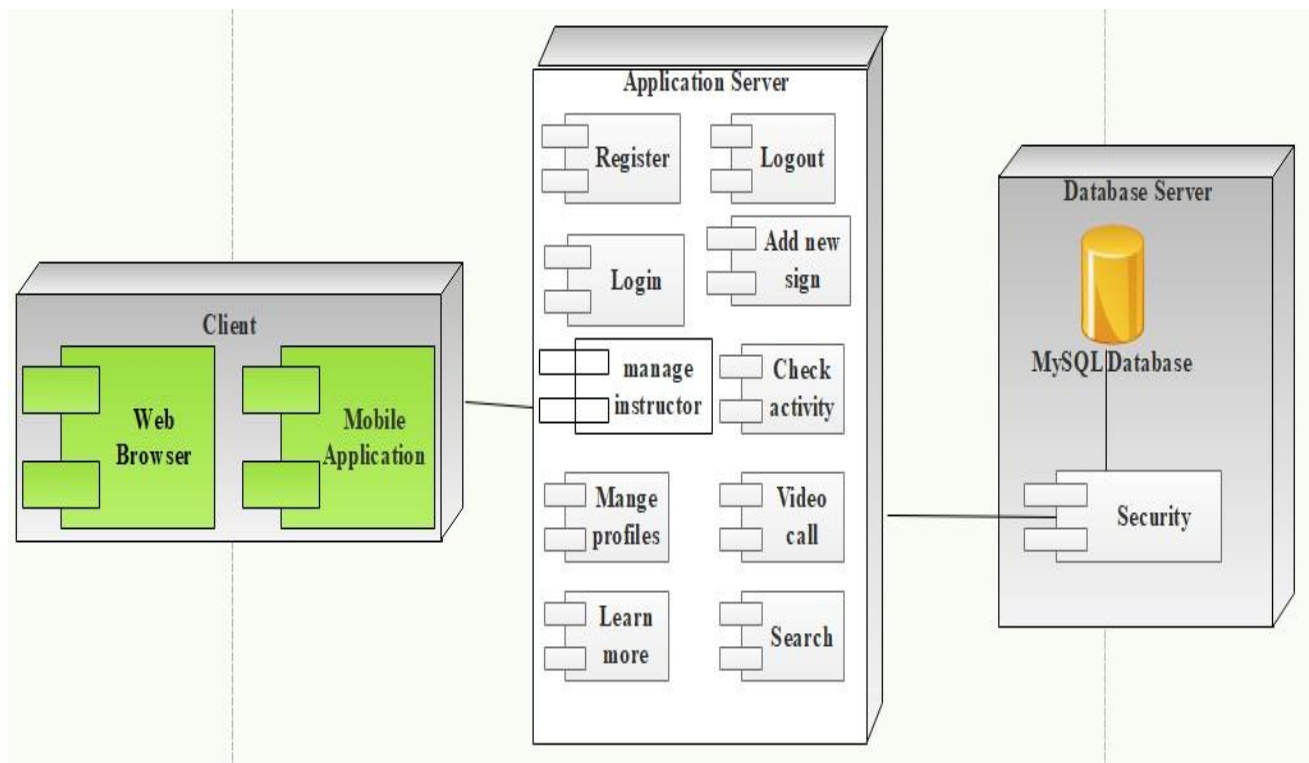


Figure 5-3 Deployment diagram

### 5.4.3. Detailed class diagram

In software engineering, a class diagram in the Unified Modeling Language is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations, and the relationships among objects. Detail class diagram is a class diagram that includes attributes, methods, attribute data types, visibility of attributes and methods, inheritance, association, aggregation, composition, dependency, and municipality (cardinality and optionality). The following figure uses the UML class diagram to specify attributes and operations with their visibility information.

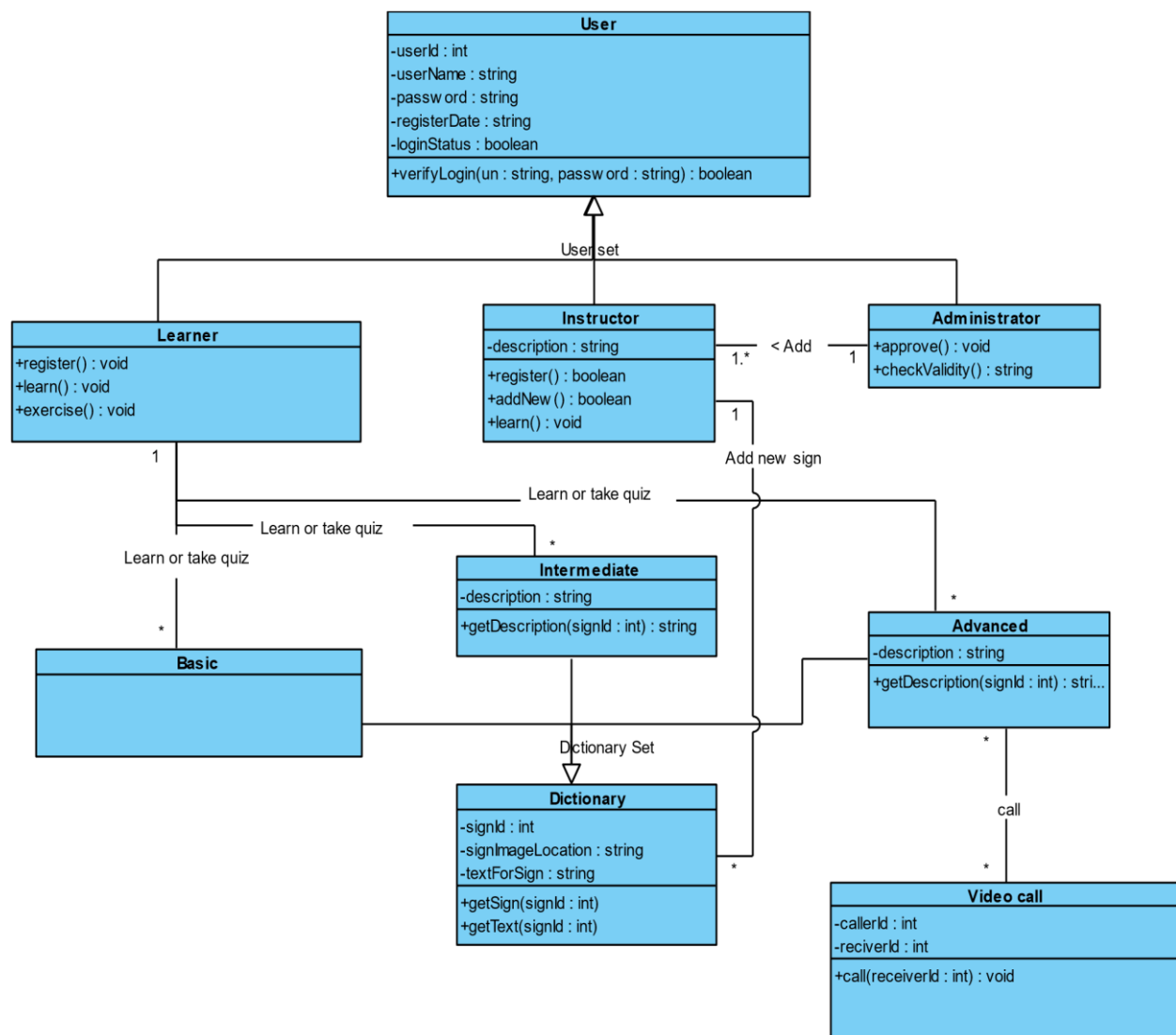


Figure 5-4 Detailed class diagram



#### 5.4.4. Access Control and Security

Access control is a security technique that regulates who or what can view or use resources in a computing environment. Its objective is to ensure that only authorized individuals gain access to information or systems services necessary to undertake their duties

In the proposed system, different actors have access to different information and data. Access control and security specify what the user can access or what cannot perform by some users. This access control is verified by a username and password. The proposed system follows a multi-user system. In a multi-user system, different actors have access to different functions and data. Then the proposed system will have: -

- **Confidentiality:** Only authorized persons can see the information. Private data is kept private; personal privacy is respected. The learner can video call with his/her friends or instructor to add new signs only using their personal account.
- **Integrity:** There are limits on who can change the data in this system. For example, the only person having an instructor role can add new sign and person having role admin can manage instructor and check the activity of all instructors.
- **Availability:** The system is available at all times to authorized users. The proposed system will work in offline and online mode and the mobile application can available 7/24 for the user.

We will use different techniques to ensure the security of the proposed system. Like session to control the access of page without authentication, and different validation for user input function like to protect from javascript and SQL injection. The best thing is we use the Django python web framework for web application development it gives us more security features.

Table 5-1 Access control and security

| <b>Operation</b><br><b>Actor</b> | Manage account | Manage profile | Learn and exercise | Add new sign |
|----------------------------------|----------------|----------------|--------------------|--------------|
| Administrator                    | CRUD           | -RU-           | -R--               | ----         |
| Instructor                       | ----           | CRUD           | CR--               | CRU-         |
| Learner                          | ----           | CRUD           | -R--               | ----         |

## 5.5. Packages

This section describes the decomposition of subsystems into packages and the file organization of the code. This includes an overview of each package, its dependencies with other packages, and its expected usage.

Package is an organized and functionality-based set of related interfaces and classes. Packages organize classes that belong to the same category or provide similar functionality. In the proposed system we category into learning management package, exercise management package, account management package, video call management package, Sign translation management package, and security management package.

**Database management package:** - The database subsystem will be implemented by the relational database management system which is used to store the persistent data of the proposed system.

**Learning management package:** - It is a subsystem that helps sign language learners to learn a selected level of signs by watching, searching, favorite and also trashing signs.

**Exercise management package:** - It is a subsystem that helps the learner to practice sign language after finishing his/her learning. This package allows the learner to test her/his self by watching the sign and choosing the corresponding text for given sign and playing the next question, and also the quiz progress management, total result and resetting progress management is done in this package.

**Video call package:** - It is a subsystem that helps the proposed system users to make a video call with his/her friends by hand gesture using his/her device camera.

**Sign translation package:** - It is a subsystem that helps the learner to practice sign language by signing and the system automatically translates to corresponding Amharic or English text.

**Account management package:** - It is a subsystem that helps an administrator to manage instructors account for approval, deny, add new instructor or delete (deactivate).

**Security management package:** - It is a subsystem that helps to secure the proposed system.

Package diagram can show both structure and dependencies between sub-systems or modules and simplify complex class diagrams; it can group classes into packages. The following figure represents the general package diagram of the proposed system.



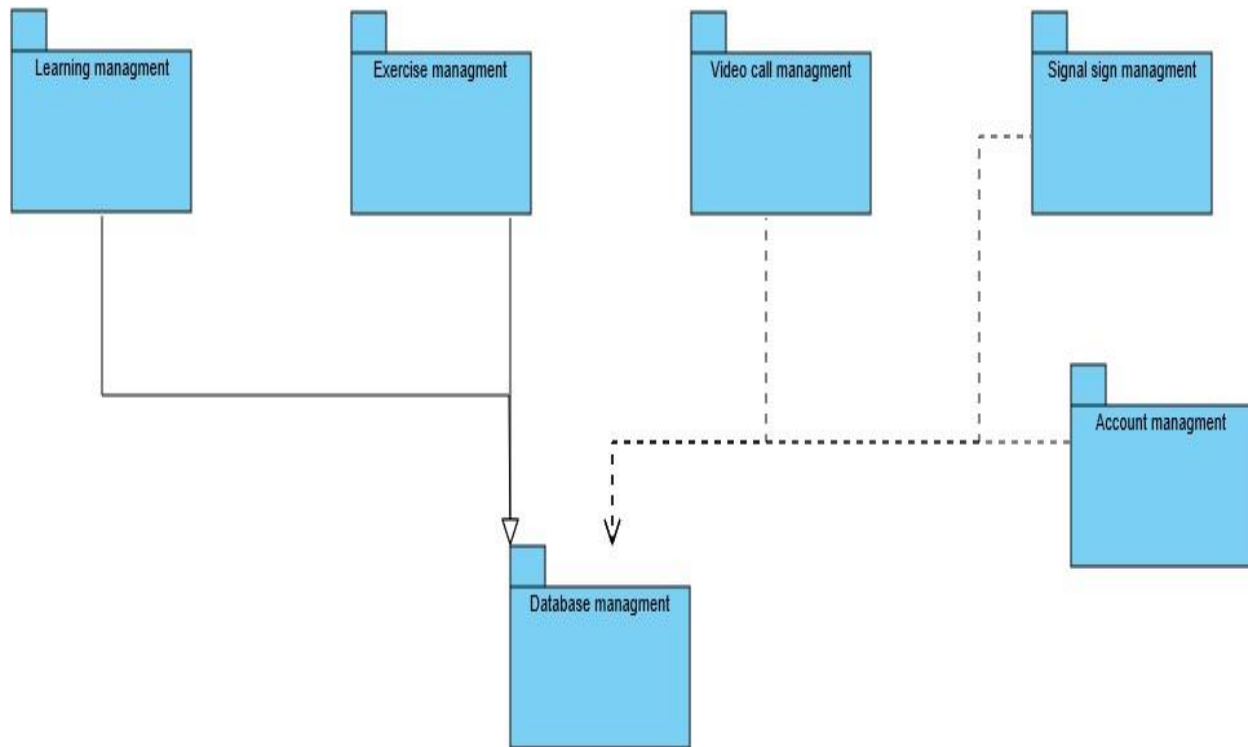


Figure 5-6 Package diagram

## 5.6. Algorithm Design

Algorithms are designed to show the flow of programs in the system. They are semantic driven rather than syntax driven. That means, the rule of syntax is not respected as other programming language but it has a complete meaning as that of syntax-based programming language. Also, Algorithms show the flow and steps of logic in each function. This design part is important in the coding part of the implementation. Some of the algorithms are listed below. Example:

### ➤ Login

- ✓ Home page displayed.
- ✓ User click on login in link
- ✓ System display login form.
- ✓ User enters username and passwords.

*If* username and password are correct, then:

*If* user is learner, then system displays private learner page:

*Else if* user is instructor, then system display instructor page:

*Else if* user is admin, then system display admin page:

*Else* username and password are not correct, then the system displays the wrong username or password message and stays in login form.

➤ Register

- ✓ Home page displayed.
- ✓ User clicks on register □ System display register form.
- ✓ User fulfills its information.

*If* the entered information is correct, then:

*If the* phone is verified, then user information registered (an account created) successfully:

*Else* phone number is not verified, then system display your phone number is not verified message and in the register form.

*Else* user information is not correct, then the system displays your input is an incorrect message and stays in the register form.

➤ Add new sign

- ✓ Instructor home page displayed.
- ✓ Instructor click add a new sign □ System display add new sign form.
- ✓ Instructor fills all required fields.

*If* the entered information is correct, then:

*If* activity is verified by admin, then a new sign added correctly.

*Else* new sign does not display to the learner, waiting for checking by admin. *Else* new sign information is not correct, then system display your input is an incorrect message and stay in add new sign form.

➤ Exercise

- ✓ Home page displayed.
- ✓ Learner click exercise
- ✓ System choice of level.
- ✓ Learner choose level of sign language

*If* selected level is basic, then system display question for basic sign language

*Else if* selected level is intermediate then system display question for intermediate sign language

*Else if* selected level is advance then system display question for advance sign language

- ✓ Learner choose answer for displayed question.

*If* selected choice is correct

*If* learner is the first time to choose then increment total result

*Else If* end of question then displays total result

*Else* display next question

*Else if* selected choice is not correct then system display wrong answer message.

*If* learner click next button, then system display next question

*Else* give another tries the same question

## 5.7. User Interface Design

User interface design or UI design generally refers to the visual layout of the elements that a user might interact with a website or technological product. User interface designs must not only be attractive to potential users but must also be functional and created with users in mind.

User interface design or user interface engineering is the design of user interfaces for machines and software, such as computers, home appliances, mobile devices, and other electronic devices, with the focus on maximizing usability and the user experience. The goal of user interface design is to make the user's interaction as simple and efficient as possible, in terms of accomplishing user goals (user-centered design) so in the proposed system we have designed user interfaces that increase the user experience.

The following figures show some of the proposed system user interface.

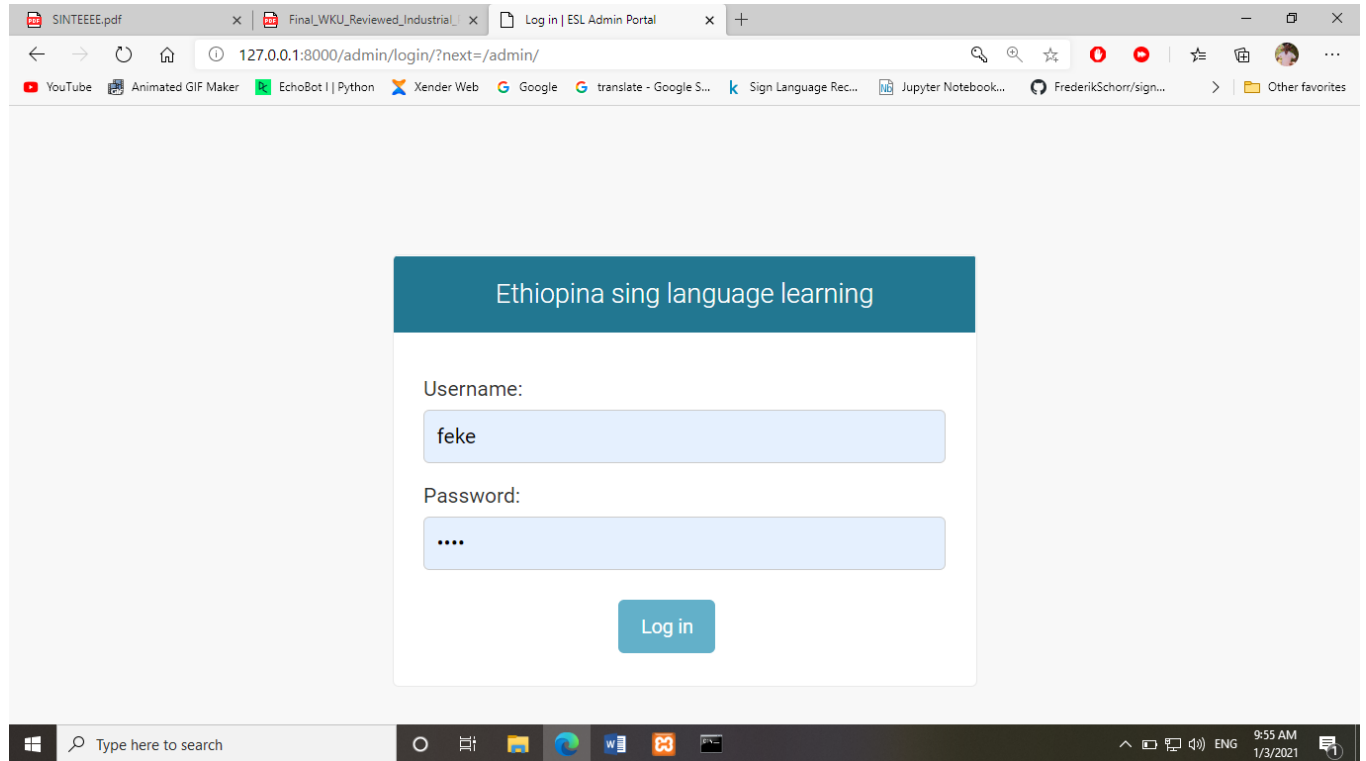


Figure 5-7 Login interface desktop

127.0.0.1:8000/admin/

# Ethiopia sing language learning

WELCOME, FEKE VIEW SITE / CHANGE PASSWORD / LOG OUT

## Welcome to Ethiopian sign language

### AUTHENTICATION AND AUTHORIZATION

|        |       |        |
|--------|-------|--------|
| Groups | + Add | Change |
| Users  | + Add | Change |

### SIGNS

|              |       |        |
|--------------|-------|--------|
| Destinations | + Add | Change |
|--------------|-------|--------|

### Recent actions

#### My actions

- softwarefeke@gmail.com  
User
- + fekenew  
User
- + Destination object (8)  
Destination
- feke1  
User
- feke1  
User
- feke1  
User
- feke1  
User
- fekedew  
User
- feke1  
User
- Destination object (4)  
Destination

71

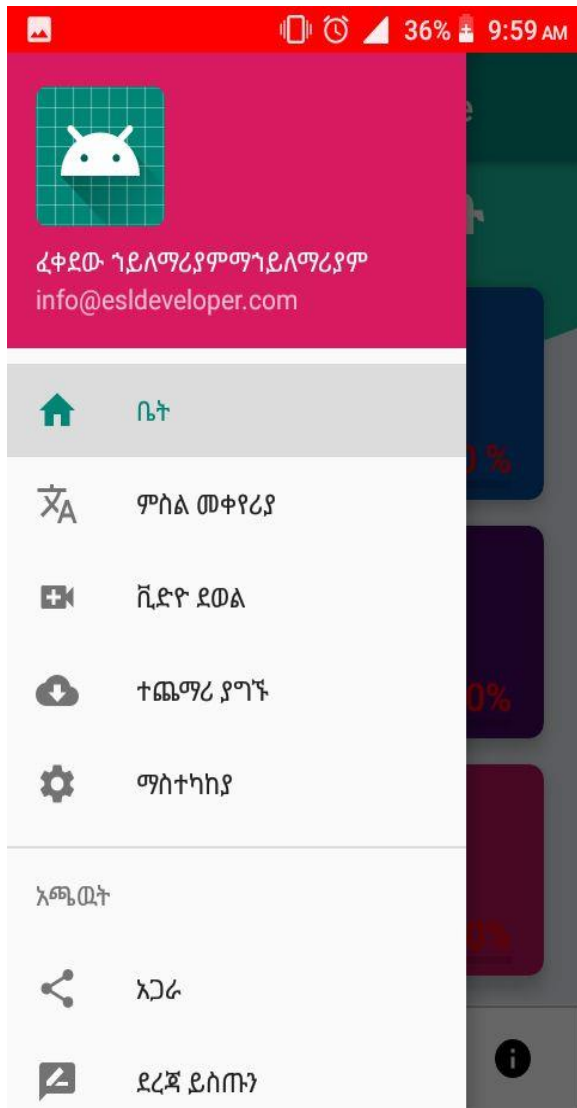


Figure 5-9 Mobile app navigation in Amharic local

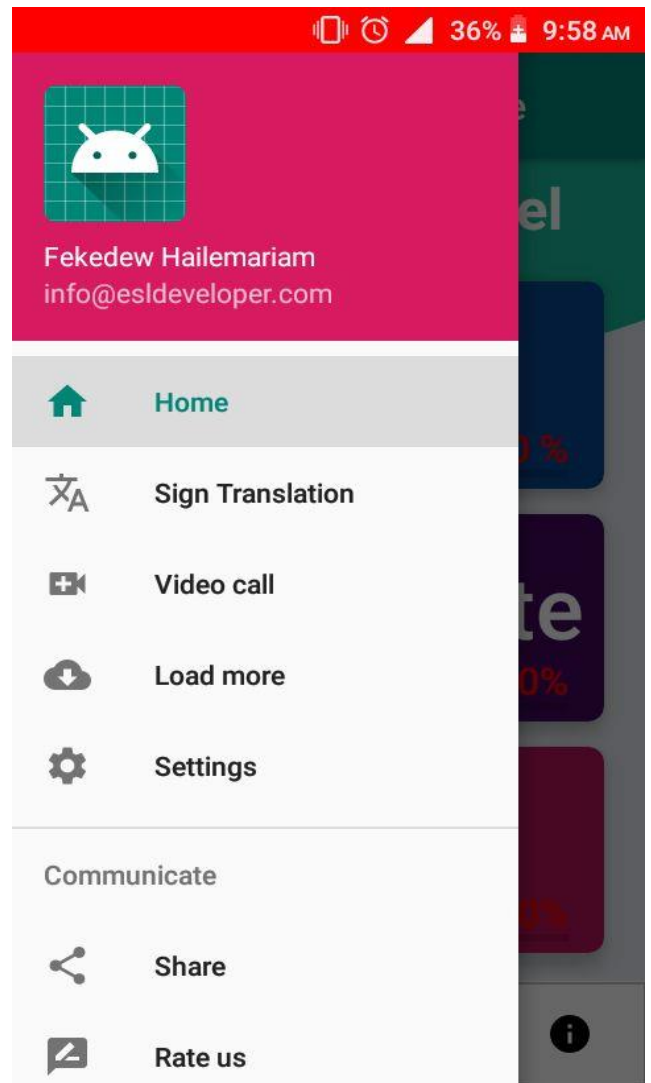


Figure 5-10 Mobile app English navigation

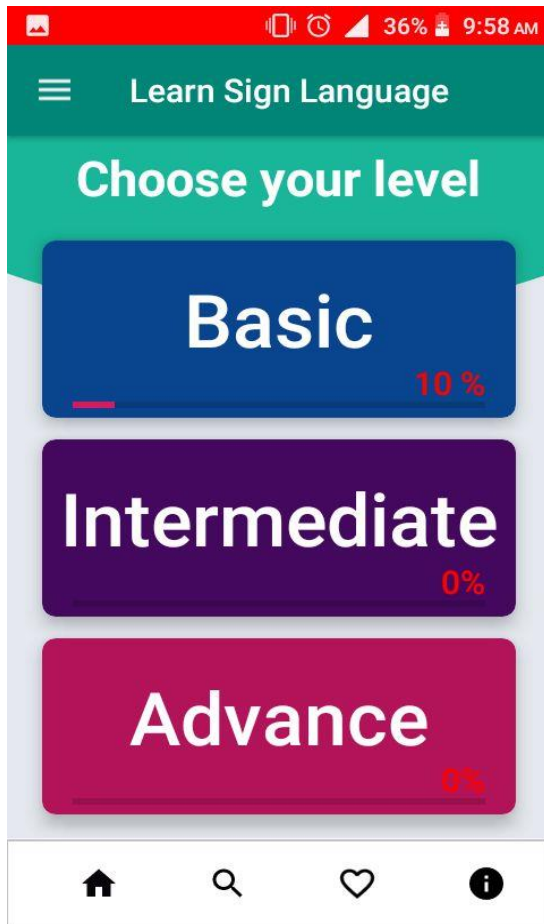


Figure 5-12 Mobile learner home page

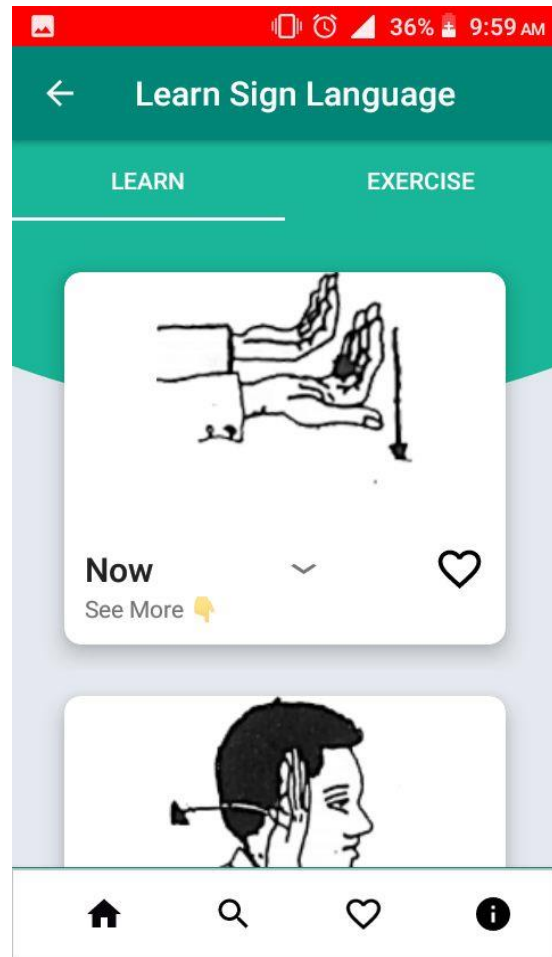


Figure 5-11 Learn day page

## CHAPTER SIX

### 6. IMPLEMENTATION AND TESTING

#### 6.1. Implementation of the database

We use relation database management system; we don't have many table to relate but we choose this database management system because of its simplicity to integrate to our development framework.

All the table we identified in persistence diagram are created and implemented with its primary foreign key relationships. All table satisfy 3<sup>rd</sup> normal form and more there is no duplication of column.

#### 6.2. Implementation of class diagram

We implement all the class shown in class diagram by following standard code style. We comment function and variable to show its usability and functionality (how it works).

Examples are:

```
package com.feke.esl1;
```

```
public class LearnItem {  
    private String imageResource;  
    private String key_id;  
    private String favStatus;  
    private String name;  
    private String descName;  
    private String descTip;
```

```
//empty constructor
```

```
    public LearnItem() {  
    }  
}
```

```
//Constructor that accept all necessary values and assign to private variables
```

```
    public LearnItem(String imageResource, String name, String descName, String descTip, String key_id,  
String favStatus) {  
        this.imageResource = imageResource;  
        this.name = name;
```



```
this.descName = descName;
this.descTip = descTip;
this.key_id = key_id;
this.favStatus = favStatus;
}

public String getKey_id() {
    return key_id;
}

public void setKey_id(String key_id) {
    this.key_id = key_id;
}

public String getName() {
    return name;
}

public void setName(String name) {
    this.name = name;
}

public String getDescName() {
    return descName;
}

public void setDescName(String descName) {
    this.descName = descName;
}

public String getDescTip() {
    return descTip;
}
```

```
public void setDescTip(String descTip) {
    this.descTip = descTip;
}
public String getImageResource() {
    return imageResource;
}
public void setImageResource(String imageResource) {
    this.imageResource = imageResource;
}
public String getTitle() {
    return name;
}
public void setTitle(String name) {
    this.name = name;
}
public String getkey_id() {
    return key_id;
}
public void setkey_id(String id) {
    this.key_id = id;
}

public String getFavStatus() {
    return favStatus;
}

public void setFavStatus(String favStatus) {
    this.favStatus = favStatus;
}
}
```

### 6.3. Configuration of application security

We use MySQL and SQLite database; SQLite is local database mean no security issue. And we use django python web application web framework it's amiable on its security. Use login with its privilege defined by superadmin. User created by super admin can access only group, or module defined by super admin.

### 6.4. Implementation user interface

The interface of the proposed system provides easy to understand, easy to use and user-friendly interface and users of the system easily use and perform their task. Everything stems from knowing our users, including understanding their goals, skills, preferences, and tendencies. Once we know about our users, we can develop a system that satisfies our user needs. The proposed system has nice look as shown in user interface design, we follow human computer interaction rule.

### 6.5. Testing

#### 6.5.1. Test case

Table 6-1 Login test case

|   |
|---|
| Test case objective: Login to the system  |
| Description: user fills username and password, the system checks in frontend error using javascript for fast interaction, if it passes the system checks for the existence and privilege of the username and password to display authorized page. |
| Test Environment: Apache MySQL and Django server must be in running state, Database Should contain appropriate table and link must be established between server and client program.  |
| Test Setup/Pre-Conditions: Apache and Django server should be in running state and User name and Password fields should be filled correctly.  |
| If user name and password are not filled correctly the system display to fill the user name and password again.   |

Table 6-2 Add new sign

|  |
|--|
| Test case objective: Add new sign  |
| Description: user first login to the system, and if user have privilege to upload sign he/she can upload image/gif fill its name and description and save to the database.           |
| Test Environment: Apache MySQL and Django server must be in running state, Database Should contain appropriate table and link must be established between server and client program. |
| Test Setup/Pre-Conditions: Apache and Django server should be in running state and every required field must be fill.  |
| If user name and password are not filled correctly the system doesn't allow to add new sign.   |

### 6.5.2. Unit testing

In this phase of testing we tried to eliminate faults in procedure and functions point of view by using black box testing. Tasks that we have performed under this phase

- Prepare unit test plans.
- Identify the unit test objectives.
- Prepare a test case that includes information such as set of test inputs, execution condition and expected output.
- Perform the tests according to our plan
- Analyses the test results

| Tested unit  | Input specification                            | Output specification         | Environment          | Result  |
|--------------|--|------------------------------|----------------------|---------|
| login page   | Valid username and password                    | Authorized home page display | Login form           | success |
|              | Invalid username and password                  | Error message display        | Login form           | Success |
| Add new sign | Valid imange/gif file, sign title, description | Success message              | Authorized home page | Success |

|  |                    |               |                         |         |
|--|--------------------|---------------|-------------------------|---------|
|  | Invalid file type, | Error message | Authorized home<br>page | Success |
|--|--------------------|---------------|-------------------------|---------|

#### **6.5.3. System testing**

We check the entire system for usability, performance, security and reliability it works well.

#### **6.5.4. Integration testing**

We test by integrating every function (module) of the system, and it works well.

#### **6.5.5. Acceptance testing**

We give the application for different end user and we receive different comment on user interface appearance and different areas.

## **CHAPTER SEVEN**

### **7. CONCLUSSION AND RECOMMENDATION**

#### **7.1. Conclusion**

Sign language is primary language for thousands of Ethiopians for their day to day activities, but there is no software or website to learn sign language at home or everywhere. By considering the problems of this our team found a solution which greatly help anyone who want to learn and exercise sign language. With the contribution of each member of the team, and advisor in developing the new system from starting of data (requirement) analysis to the implementation, we reached to the final result. During the developing this project, team face some challenges, but by the cooperation's of all team members reach to their goal and come with this absolutely new and curios project.

#### **7.2. Recommendation**

We do all this in just 3 weeks we haven't enough time to do our best you can add more sign language words, sentence and make the application great. And in machine learning (sign translation side) we only train English alphabet because of shortage of time and resource you can simply train more sign and use this application to test your model. The application is extensible you can use different model for sign language translation or even for other image classification application.

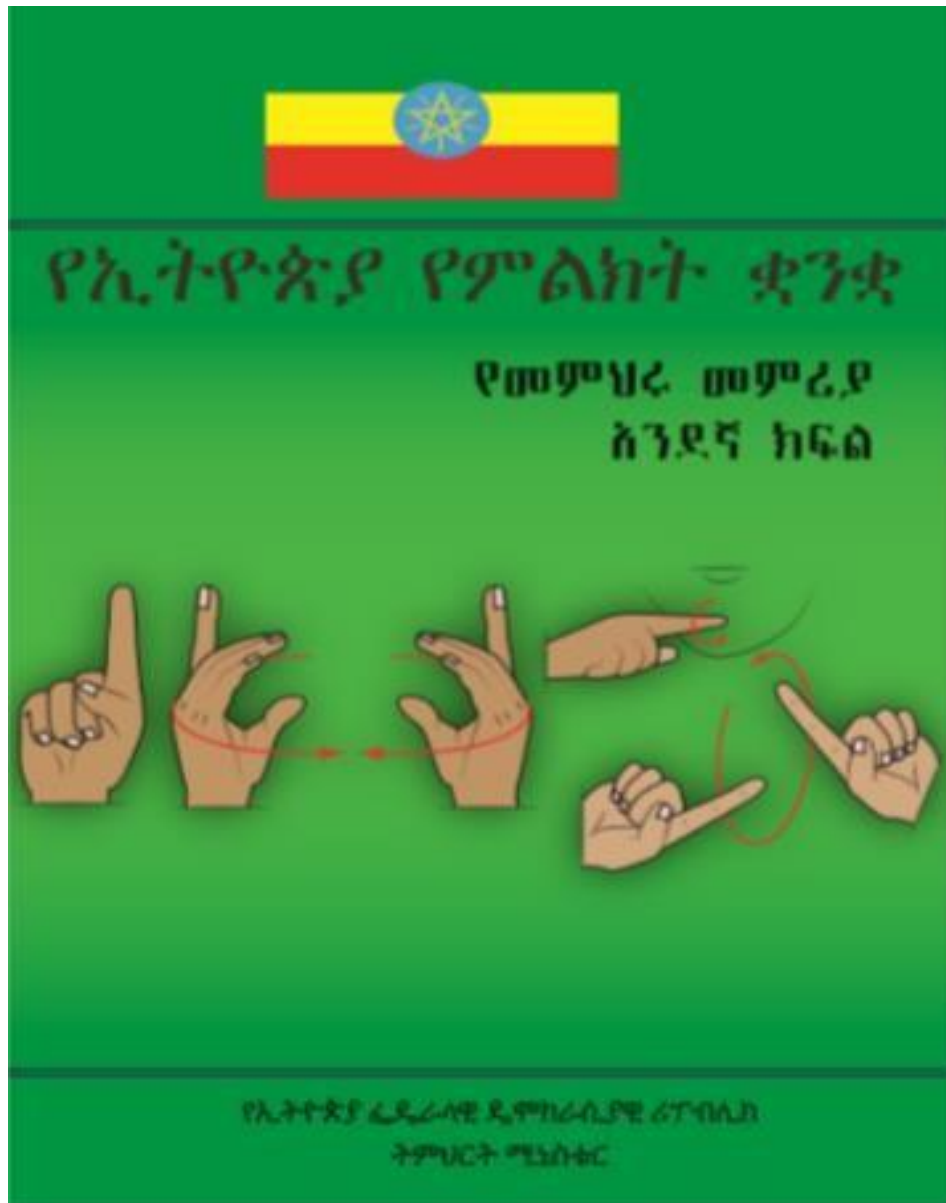
## References

- [1] D. F. WOLDE, "MACHINE TRANSLATION SYSTEM FOR AMHARIC TEXT TO ETHIOPIAN SIGN LANGUAGE," *ADDIS ABABA INSTITUTE OF TECHNOLOGY*, pp. 1-2, October 2011.
- [2] L. A. Wubshet, "Example Based Amharic Text to Ethiopian Sign Language Machine Translation," *Reaserch* , pp. 16-17, June 2018.
- [3] S. John, "39 Examples of Design Goals - Simplicable," simplicable.com, 24 11 2018. [Online]. Available: <https://simplicable.com/new/design-goals>. [Accessed 17 01 2020].
- [4] B. Golden, "What is Systems Architecture ?," .lix.polytechnique.fr, [Online]. Available: [http://www.lix.polytechnique.fr/~golden/systems\\_architecture.html](http://www.lix.polytechnique.fr/~golden/systems_architecture.html). [Accessed 17 1 2020].

## APPENDICES

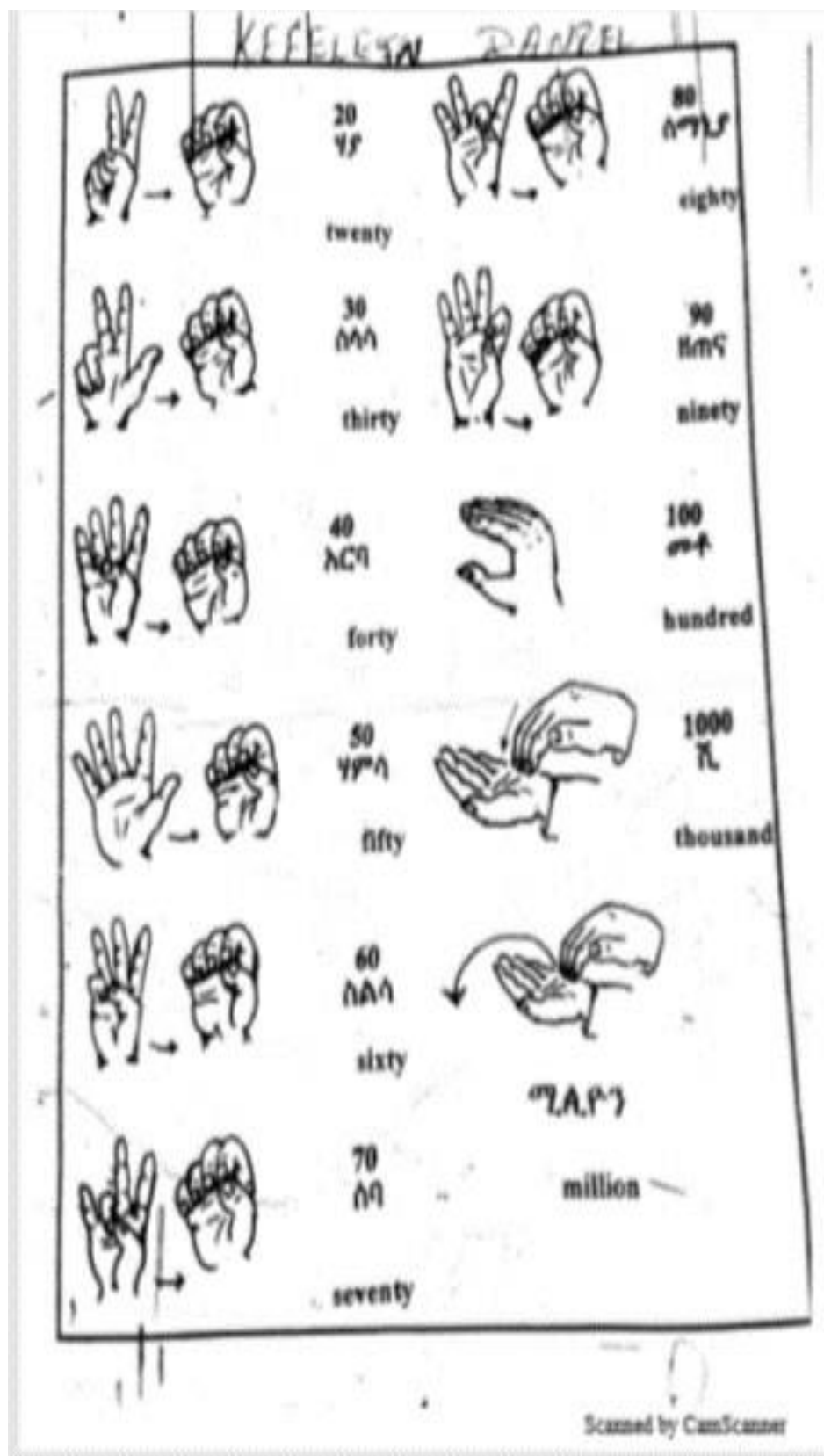
### Appendix 1

Some of the book used during traditional teaching learning system











## Appendix 2

Some interview person

(K=Kefelegn, male, sign language instructor at Wolkite, selam ber school; I=interviewer)

We had face-to-face interview with Mr. Kefelegn and some of the general conversation during the interview looks like as follows: -

I: How you teach your deaf student?

K: First I show the sign and write the text for corresponding sign, and draw the image for sign if possible.

I: Where can I get book of sign language and other information?

K: It's good to ask college instructor for more information, and I give you student book (he gives me see appendix 1 image 3&4).

I: What is the main problem you face during teaching and learning process?

K: Misunderstanding of sign specially for sign their image cannot be drawn and it's tedious.

I: Please guess which type of application may help you (we give some choice like image with animation, only image, and video).

K: Image with its animation is easy to understand and personally I choose image with animations if possible.

(A= Mrs. Aynie Belete, female, sign language instructor at Debre Birhan college; I=interviewer)

We had telephone interview with Mrs. Aynie Belete and some of the general conversation during the interview looks like the followings: -

I: What are the criteria to classify sign language learner as basic, intermediate and advance?

A: She tell me about BR4, BR5 and BR6

I: What should learner do to learn out of class and for normal person?

A: Normal person or deaf person to learn sign language should have to pay monthly fee as school or instructor regulations.

I: Where can I get some book of sign language?

A: Send teacher guide book, see appendix 1