ADAMA SCIENCE AND TECHNOLOGY UNIVERSITY

SCHOOL OF ELECTRICAL ENGINEERING AND COMPUTING(SOEEC)

DEPARTEMENT OF COMPUTER SCIENCE AND ENGINEERING(CSE)

SENIOR PROJECT SUBMITTED FOR THE PARTIAL FULFILLMENT OF DEGREE PROGRAM

TITLE – Save Geez Learning Aid

Group Members	ID-NO
1. Anteneh Teshome	A/UR4748/09
2. Aser Kassahun	A/UR4151/09
3. Bereket Feyssa	A/UR4669/09
4. Degife Dafursa	A/UR4800/09
5. Solomon Girma	A/UR1657/09

Approval

Name	ID	Signature
1. Anteneh Teshome	A/UR4748/09	_____
2. Aser Kassahun	A/UR4151/09	_____
3. Bereket Feyssa	A/UR4669/09	_____
4. Degife Dafursa	A/UR4800/09	_____
5. Solomon Girma	A/UR1657/09	_____

Advisor Name	Signature
Fetelework Kedir	_____

Table of Contents

List of Figure

List of Tables

Acronyms

- **SGA –** Save Geez App
- **BC –** Before Christ
- **C –** Century
- **OS –** Operating System
- **IOS –** iPhone Operating System
- **RAM –** Random Access Memory
- **DB –** Data Base
- **HTTP –** Hypertext Transfer Protocol
- **UML –** Unified Modeling Language
- **XAMPP –** Cross-platform, Apache server, MySQL, PHP and Perl
- **MS –** Micro Soft
- **UX –** User Experience
- **UI –** User Interface

Abstract

In our world there are thousands of languages in a different places, countries and continents. Even if in one country there might be two, three, hundreds or thousands of languages. Among those countries, our country Ethiopia is a country that has more than 80 languages spoken within the country. Among those languages, Geez is one of the ancient and historical language. But this language is also one of those languages that are going to be extinct. To save this language different bodies of the people are trying their best. But because of the effort is not enough, it is now in danger.

As one body of the society, we want to add our effort to save the language and our culture too. As we see, now a days this generation is so friendly with technologies, mostly to mobile phone. A lot of information is being shared using mobile phone. People can learn different things using their mobile phones. So, our proposed solution or method to save the language is a mobile Application that is Geez language learning aid. The application has different courses classified in different categories and a Quiz game that is also classified in categories to help the users to evaluate themselves that how much they learn the language. So, we hope that our proposed system can make a great contribution on the effort of saving Geez Language.

CHAPTER ONE

INTRODUCTION

1.1 Introduction

Geez is an ancient South Semitic language of the Ethio-Semitic branch language. The language originates from what is now northern Ethiopia and Eritrea in East Africa. Nowadays, Geez is being used only as the main liturgical language of the Ethiopian Orthodox Tewahedo Church and Eritrean Orthodox Tewahedo Church, the Ethiopian Catholic Church and Eritrean Catholic Church, and the Beta Israel Jewish community. So, Ge'ez went extinct as a natural language over 1000 years ago and is no longer spoken as the native tongue of any people.

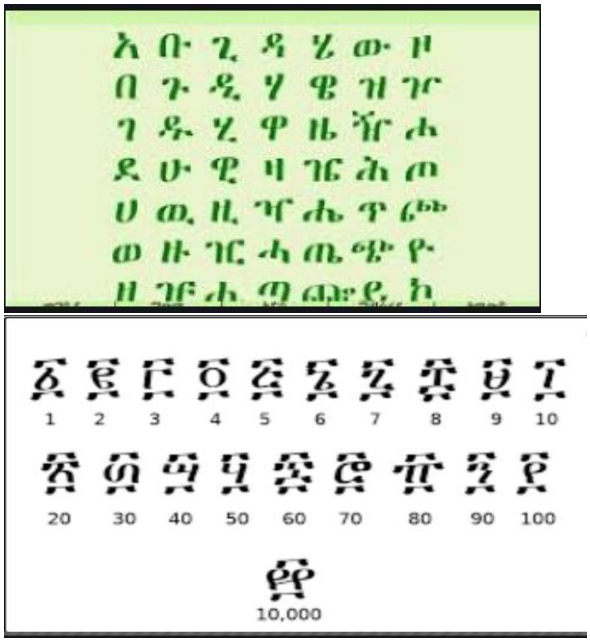

Fig 1.1: Geez Language Letters and numbers

The reason why we came up with this proposal is to help peoples learning a Ge'ez by promoting communication and language development in which people are encouraged to interact and communicate with other people from all over the world learning the same language. We used the Language to translate into a Local language which has a broad speaker that are Amharic and Afaan Oromoo**,** we used to translate it into an international Language English**,** to make peoples from abroad know this historical Language.

The interaction would involve several kinds of activities, such as Quiz games. The system will be easy to understand and use even by non-technical users such as young children and any other people. It will be also easily extendable, allowing programmers to develop new functionality without dealing with low level details.

Fig 1.2: Ancient Books written by Geez

1.2 Background

Geez is an ancient African language that originated in the first millennium BCE in southern Eritrea and northern Ethiopia. It has been a scholarly and liturgical language in Ethiopia since the sixth century CE and remains in use today in the liturgy of the Ethiopian Orthodox Church. Thousands of unique texts are written in this language, including many valuable texts of history, literature, and theology, and the earliest versions of the biblical books of Jubilees and Enoch. As we know that this historical Language is being extinct.

Learning a dead language helps to open the door to a past and history that many modern languages can't offer. It teaches a cultural sensitivity and historical understanding that can, essentially, help us to effectively learn from the past. It gives us interdisciplinary access to the thoughts and ideas of human beings hundreds or thousands of years ago and allows us to hear their voices and learn from their wisdom.

1.3 Statement of The Problem

Geez language is the oldest Language in Ethiopia as well as in Africa. But this Language is being extinct. The death of a language is the death of culture. For language lovers, that's even more tragic than the extinction of a species. And also, there are a lot of books written by this Language, such as Scientific, Astronomy, Medical and so on. So, if we don’t save this Language, we can’t be able to contribute to the Growth of this Fields. So, to save this Language and to contribute to a growth of those fields that are mentioned before, the Good way and a way which allow us to reach a broad number of people is Teaching this Language using a Mobile Application.

There are a few Mobile Apps which are related with Geez Language, which are used for Learning Geez. So, the system (mobile Apps) that are related to Geez are few and

they aren't prepared in a simple or user-friendly way. For example, Most of them are dictionaries. To use the Dictionary First we have to know Geez Language. So, it is so hard to use the App and it can't be used for Learning purpose.

1.4 Objective of the Project

1.4.1 General Objective

The main objective of this project is to develop a geez learning Mobile App which translate Geez language into Amharic, Afaan Oromoo and English.

1.4.2 Specific objectives

- To design and Develop an interactive and user-friendly GUI.
- To develop a quiz gamming feature of the app
- To develop a Mobile App that translate Geez into three other languages and also sign language.
- To develop app that runs on both android and iOS

1.5 Feasibility Study

1.5.1 Economic feasibility

The advantage of this application is, there is deep knowledge written in geez about science, fiction, astronomy, medicine and so on. If this generation knows this language well, it contains metaknowledge which has high Contribution to the growth of the fields that have great contribution on the development of one Country.

1.5.2 Operational feasibility

- usually simple to use for any user who have basic skill about smart phone.
- the system will benefit anybody who uses it.
- Since the system will be available on Android mobile operating system
- Users will not need much time to adapt the system because the system is much user friendly and adaptable, its interface is easy to understand because it's simple which is designed for user to easily understand the system.

1.5.3 Technical feasibility

Technical feasibility study about evaluating if the current technology has a potential develop or unusable to support our proposed system. implementation of the proposed system will use android-based platform. By assuming required hardware and software resources that are available for the development and implementation of proposed system. Therefore, it is technically feasible.

1.5.4 Scheduling feasibility

Scheduling feasibility is the probability of project to be completed within its scheduled time limits. Because of we have divided tasks and commitment to finish the project, it will be completed on the time given. Currently, we are in collecting data and after it is completed, we will go to inserting the data in a database. Then the implementation will be done. After that we will test the App in a different mobile phone and we will collect opinion from the users.

1.6 Scope and Limitation

1.6.1 Scope of the Study

Our proposed system has level of courses. It First teach Letters of Geez Language, then it shows how to make Greeting in Geez and progressively it teaches many things that are classified in group of courses. The alignment of the course is arranged from easier to advanced one to make the user easily learn the language. So, the group of courses are: -

- ✓ Geez Alphabets
- ✓ Geez Numbers
- ✓ Greeting and Daily use
- ✓ Family Members
- ✓ Foods
- ✓ Clothes
- ✓ Days of the week
- ✓ Months of the Year
- ✓ Animals
- ✓ Natures
- ✓ Colors
- ✓ Professions
- ✓ Geez Grammar

So, these categories will not include all the things that are under them. We will store most commonly used or known things or words in our database. So, the user can learn

words that are stored in our database only. But throughout time or while the Application is updated, other words will also be included, as well as other extra courses.

The system also has Sound recording of Letters, words and sentences to teach the users how to pronounce them. It has sound button next to them which makes a sound while it is clicked. And it has a special feature which is used to help people who are unable to hear. Letters in the course will also be translated to sign language to help them to learn Geez language.

The other thing we include in our system is a quiz game. This method is good for users to help them not to forget what they Learn. The quizzes are also classified in a category that is the same with the courses. It also includes questions with pictures to make the quiz more attractive.

The system also has a dictionary that allows the user to translate an unknown word that is included in our database from geez to three of the languages, namely Amharic, English and Afaan oromoo.

1.6.2 Limitation of the Study

- Lack of previous research studies on the topic
- Lack of Geez fluent speakers
- Inability to get adequate data set
- Lack of Latest PC which has sufficient RAM and Processor Speed

1.7 Significance of the Project

- Saving our historical Language from being Extinct
- Teaching the Language as well as our culture
- Contributing for the growth of Science, Astronomy, Medical and other Fields
- Contributing for the growth of the country
- Publicizing our Language and our culture to the World

If the idea of helping to bring a language back to life isn't a reason enough to learn a dead language, learning a dead language may have some very interesting and unexpected benefits.

Much like learning a modern language, learning a dead language also has many of the cognitive benefits of that language learning offers us from an improved memory and decision-making skills to a decreased risk of dementia. Finally, learning a dead language can actually help us to learn many modern languages.

1.8 Methodology

1.8.1 Software Development Method

Development methodology of the project is an evolutionary design or incremental development approach. The model is a refined version of the waterfall methodology allow making changes during the design phase with the use of prototypes as the name suggests. This approach fits the project because we need a prototype to fully understand the requirements before we proceed with the design.

1.8.2 Data Collection Methods

1.8.2.1 Documentation Review

We have determined to do an information gathering by finding a Geez Learning Books in the Book sell stores.

1.8.2.2 Personal Interview

we have determined to do an interview with a person who can speak the language well such as Deacon, Priest ("Meri Geta").

1.8.3 Development Tools

Activities	Tools/ Programs	Purpose/Reason
Client-side coding	Flutter	A language used to develop the mobile Application. This Framework is faster than many other application development frameworks. Test and fix bugs are faster. Also, the code-base can be used for both Android and IOS devices.
Platform	WINDOW 10, android debugger and Android device	The reason we choose window 10 is we have a better experience of it than other OSs. Android device and debugger is used to debug the device.
Database server	firebase, XAMPP	Firebase is used to store and sync data between users in real-time or to make the users share one real time Database instance and

		automatically receive updates with the newest data. XAMPP is used to design physical database structure.
Diagram tools	Enterprise Architect, Lucid chart	To develop UML diagram of the project we use enterprise architect and Lucid chart.
Editors	Visual studio, Adobe Photoshop	Visual studio is an editor used to edit the web interface and Adobe photoshop, to edit the pictures which we used in the APP.
Documentation	MS Word	Application software we used to Document our project.
Group working platform	GitHub	It allows the group members to access the source code remotely that will help the development to be quick.

Table 1.1: Development tools

1.8.4 Testing Method

This method is used to test the implemented system. After we implement the whole system, we will distribute the App to some of our friends and other people. Then after that we will test whether the App works in different mobile phones with different models and other differences.

1.9 Required Resources with Cost

The most influential factors which determine the final cost to build an app are:

- Complexity and the number of app features
- Back-end infrastructure and connected APIs
- Complexity of UX/UI design

- Development approach (native, mobile web, hybrid, etc.)
- Number of platforms to be developed (iOS, Android, web, etc.)

1.9.1 Cost Breakdown

NO	ITEMS	PRICE (in Birr)
1. `	Laptop (Core i5 with RAM-6GB)	25,000
2.	Android Studio Software	Free (installed from website)
3.	Mouth	200
4.	Mobile phone (>27API, >3GB RAM)	7,000
5.	Flash Disk(64GB)	400
6.	Window 10 Operating System	8,000
	TOTAL	40,600

Table 1.2: Cost breakdown

1.10 Task and Schedule

Weeks / Phases	Week 1-2	Week 3-4	Week 5-6	Week 7-8	Week 9-10	Week 11-15	Week 16-18	Week 19-20
Phase 1 Documentation								
Phase 2 Literature Review								
Phase 3 Preparing Datasets								
Phase 4 Implementation								
Phase 5 Testing								

Phase 6 **Releasing the system and introducing it to the users**								
Phase 7 **Maintenance or adding some change after user's response**								

Table 1.3: Phases of tasks with Schedule

1.11 Team Composition

Name	Responsibility	Main Activity
Aser Kassahun	Team Leader	Coordinate and lead the Members, Design, Architecture, Implementation and Documentation
Solomon Girma	Vice Team Leader	Requirement Analysis, Design, Architecture, Implementation and Documentation
Bereket Feyssa	Member	Requirement Analysis, Design, Architecture, Implementation and Documentation
Anteneh Teshome	Member	Requirement Analysis, Design, Architecture, Implementation and Documentation
Degife Dafursa	Member	Requirement Analysis, Design, Architecture, Implementation and Documentation

Table 1.4: Composition of the team

CHAPTER TWO

Description of existing system

Language learning software is specifically designed to help users learn, increase their knowledge, or maintain foreign language skills. Currently there are a lot of Software which are useful for Learning different Languages. Most of this software are Mobile Applications which allows people to use them to learn a new language anywhere and anytime they want. They are also used by companies and individuals across various industries to increase foreign language skills for job assistant, career advancement, or personal satisfaction, as well as to save a language from being extinct.

We have looked software and Books which are developed or written to teach Geez language and this

- Dictionary for geez language
- Geez keyboard for Androids
- Some religious books which are written in Geez Language using Application
- A few books written for learning Geez language
- Geez Language is given as a course in Foreign and Local Universities such as
    - Addis Ababa University
    - Bahir Dar University
    - Debra Markos University
    - Mekelle University
    - Cambridge University Faculty of Divinity
    - Florida State University

2.1 Major function of existing system

There are a few software and Books that are used for learning Geez language. The dictionaries are used to translate geez language to other languages such as Amharic and Tigrigna. The first android application is called “Geez Tig”. It is a dictionary for translating Tigrigna to Geez and also it has some tips for learning Geez language such as Geez Numbers, alphabets and some conversations that is translated to Tigrigna. The other application is a dictionary which translate Geez to Amharic and vice versa. There is also an application called “Fidel”, which is designed to easily teach Geez alphabet characters. Fidel mainly consists of Hahu, Abogida and Ethiopian numbers. The app is designed specifically for kids, but it can be used for anyone who is interested to easily learn Ge'ez or Amharic letters and also there are some books written for Learning Geez language.

2.2 Users of current system

The users of Current Systems are people who can use Smart phones. As we know most people are using smart phones such as androids and iOS. So, currently implementing Mobile platform is a best way to connect with a mass of People and to achieve what we want to tell to the world.

Other than the Software, there are also traditional books written by Geez language for spiritual purpose which has been used in the Churches and also there are books written to teach Geez language. So, the users of this manual system are people who are the servant of the churches, people who love to read books and students of the universities which we mentioned them above.

2.3 Drawback of current system

- If we take an application called Fidel, it has some drawbacks
  - It only teaches letters of Geez.
  - It can be used only for beginners or for people who have no idea about Geez language.
- "Geez-Tig" Application
  - It can only translate Geez to Tigrigna or vice versa, so people who can't speak Tigrigna can't use it.
  - It only contains some courses.
  - It doesn't use much conversations to teach.
  - It doesn't use progressive way to teach. Because it tries to teach conversations without teaching words and their meanings.
  - The interface isn't attractive or user-friendly.
- Books for Learning Geez
  - Because of it is a book, we can't learn the right pronunciation unless there is a person, who can guide us.
  - They are limited in number and found in some places.
  - Most of them are too old.
  - Most of them are written for spiritual purpose, so people who aren't the followers of that religion might not be interested to read.
  - Because of the reading habit of current people is very low, it might not be reached to large number of people.

2.4 Business rule of current system

There are few mobile apps we found on internet which support geez learning. First of all, the user should know how to use smart phone and he/she must download the app from google, play store or App store.

They are: -

- **Geez tig**
  - Downloaded from google
  - Installed on smart phone
  - A word should be entered in the space provided to translate the word
  - List of number
  - Alphabet
  - Short static conversation
- **Geez Amharic translator**
  - Downloaded from google
  - Installed on smart phone
  - Space provided to enter the word which is wanted to be translated.
  - Translate word from Amharic to geez vice versa
- There are also few books provided to support geez learning. So, the user should buy the book from book store or from the place where the book can be found.

CHAPTER THREE

Proposed System

3.1 Overview

Save geez learning aid is a mobile based application that is used to teach people geez language. This app will make an effort to save geez language from being extinct by giving different courses that are classified under different categories to make users learn the language easily. It also has quizzes to evaluate the level of understanding of the users. it has dictionary to translate geez word and Voice with the geez words and sentences to make the users know the right accent of the word or sentence.

3.2 Functional requirement

- **Sign up**

The user must sign up to use the Application. And once it sign up it can use the App without signing up again.

- **Teaches letter**

Our system will teach letters with sound for beginners who have no idea about the language.

- **Teaches number**

Our system will teach numbers with sound for beginners who have no idea about the language.

- **Teaches word**

Our App will teach words with picture and sound.

- **Translate words**

Our App will translate geez words to Amharic, English and Afaan Oromoo as user preference and vice versa.

- **Quizzes**

App will give quiz to test user progress on language

- **Updates**

Our App will be updated with new word throughout time.

3.3 Nonfunctional requirement

These are the constraint on the services or functions offered by the system that also in composes time constraint on development process and standards. They are more critical than individual requirement. So, to specify the following major emergent properties of save geez Learning aid app: -

- **Usability**

App should be easy to learn and understandable for user. Since the app will be developed by considering all users, who have no basic skill to skilled users.it must be easy to use and learn.

- **Compatibility**

App should have to be compatible: -Since this system is mobile based system, it is compatible with both android and iOS.

- **Accessibility**

App should be accessible at every time and every place with internet to load content and without internet if contents were once loaded.

- **Performance**

Performance requirements define acceptable response times for system functionality.

- ✓ The load time for user interface screens shall take no longer than thirty-five seconds
- ✓ The log in information shall be verified within eight seconds
- ✓ Queries shall return results within eight seconds

- **Maintainability**

The App should be maintainable. Because the interaction between subsystems will be loosely couple, changes made on our App such as adding other functionality shouldn't affect the existing functionality of the system.

❖ **Documentation**

At the completion of the project, every activity of the entire development, design and other process will be documented for future reference. Also, there will be a documentation of implementation for maintenance during application failure. Furthermore, this will help for further maintenance and reusability of our system.

3.4 System Model

3.4.1 Use case Scenarios.

Scenario: 1

Name of use case: Sign up

Participating instance actor: any user.

Entry condition:

- ✓ Internet connection have to be available until the contents are load.
- ✓ User should fulfill all data required to insert

Normal flow of events:

1. The app display the Sign-up form.
2. Fill the form and click Sign up button.

Abnormal flow/Alternative condition:

- ✓ If the user fails to login by inserting incorrect email or password the App displays an error message (“incorrect email or password”)
- ✓ If the user inserts a password that is less than 8character, the system displays a message “please Enter a password 8 char long”.

Exit Condition:

- ✓ It displays Sign-up successful message

Scenario: 2

Name of use case: Login

Participating instance actor: registered user.

Entry condition:

- ✓ Internet connection have to be available until the contents are load.
- ✓ User have valid password

Normal flow of events:

1. The app display the login form.
2. Fill the form and click login button.

Abnormal flow/Alternative condition:

- ✓ If the user fails to login by inserting incorrect email or password the App displays an error message (“incorrect email or password”)
- ✓ If the user inserts a password that is less than 8character, the system displays a message “please Enter a password 8 char long”.

Exit Condition:

- ✓ The Log in page will pass to homepage.

Scenario: 3

Name of use case: Take course

Participating instance actor: any user

Entry condition:

- ✓ The user must log into the App

Normal flow of events:

1. The course icon will be displayed in the Homepage.
2. Course have to be clicked.
3. The App displays categories of the course.
4. Each category has different types of courses.

Exit condition:

- ✓ View words or sentences in order

Scenario: 4

Name of use case: Take quiz

Participating instance actor: any user

Entry condition:

- ✓ The user must log into the App
- ✓ The user should take some courses to take the quiz.

Normal flow of events:

1. The Quiz icon will be displayed in the Homepage.
2. Quiz icon have to be clicked.

3. Categories of the quizzes will be displayed on the page.
4. Selecting the category.
5. Questions will be displayed which are related with the selected category.
6. Choosing the answer and click the next button.

Abnormal flow/Alternative condition:

- ✓ if the user answers a wrong answer, it displays wrong answer message

Exit condition:

- ✓ after the user complete the questions, it gives a result and depending on the result it gives a star as a reward.

Scenario: 5

Name of use case: Translate

Participating instance actor: any user

Entry condition:

- ✓ Letter or Word or Sentence that is needed to be translated should be clicked or inserted.

Normal flow of events:

1. The user has to enter to course page.
2. The course category should be selected.
3. Lessons that are found in the category will be displayed.
4. Click on the Letter or Word or Sentence which is found in the category, to get the translation.
5. The translation will be displayed in Amharic, Afaan Oromoo or English languages as the user choice.

Abnormal flow/Alternative condition:

- ✓ If the user inserts a word that is not available in the database, it shows “word not found” message

Exit Condition:

- ✓ After the user gets the translation option and select it, the translated word will be available and the voice is also available.

Scenario: 6

Name of use case: Update word

Participating instance actor: Admin

Entry condition:

- ✓ The admin must be verified and have to get an access to the database

Normal flow of events:

1. The Admin have to use firebase which has previously stored database.
2. The database has to be edited.
3. The admin must save the change.

Abnormal flow/Alternative condition:

- ✓ if the admin tries to update the word that is already updated, there will not be change on the App.

Exit Condition:

- ✓ the App will add the newly added words, when the App get an internet connection and be updated.

Scenario: 7

Name of use case: Update course category

Participating instance actor: Admin

Entry condition:

- ✓ The admin must be verified and have to get an access to the database

Normal flow of events:

1. The Admin have to use firebase which has previously stored database.
2. The Admin should create another category on the Category list in the database
3. The admin must save the change.

Abnormal flow/Alternate conditions:

- ✓ if the admin tries to update the content that is already updated, there will not be change on the App.

Exit Condition:

- ✓ the App will add the newly added courses, when the App get an internet connection and be updated.

3.4.2 Use case Model

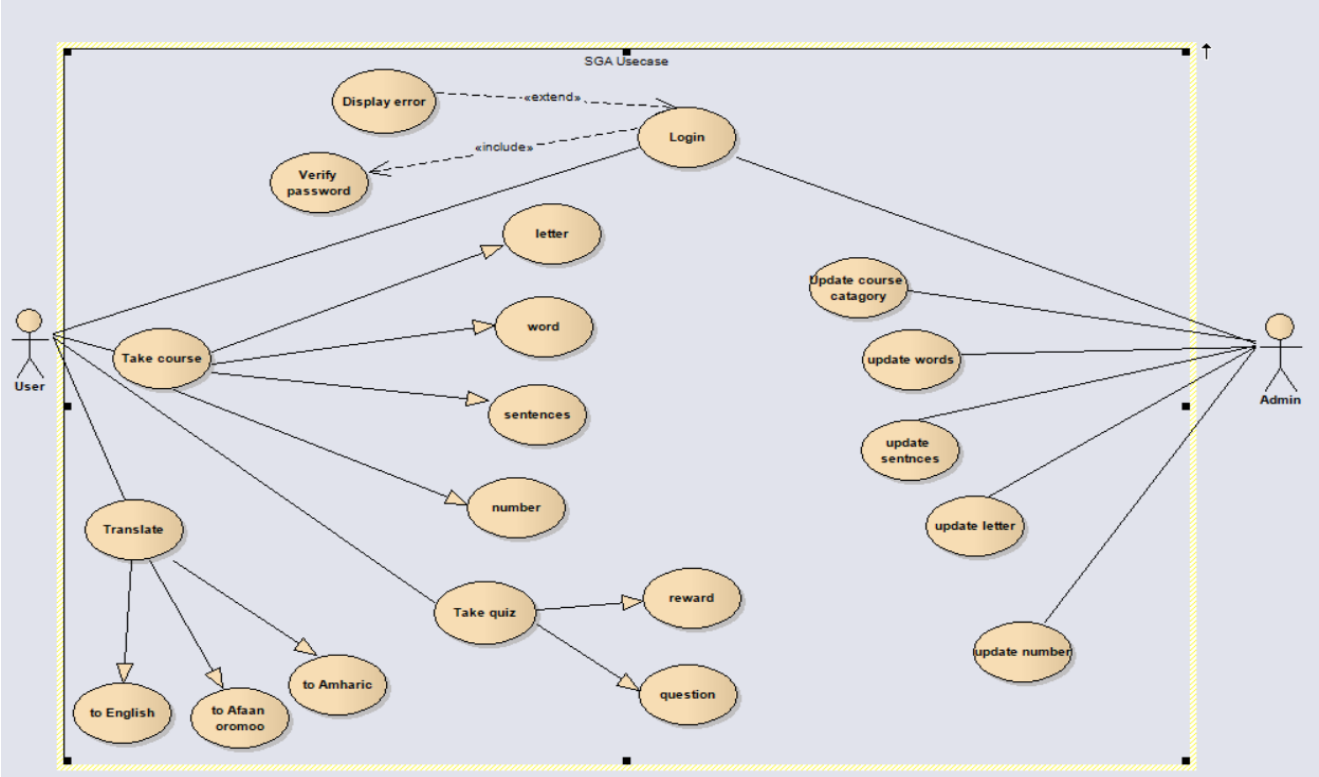

Fig 3.1: Use case model for the system

Description 1	
Use case name	Login
Use case number	1
Use case description	Verification to get access to the App
Participating actor	Users
Pre-condition	The user must sign up to the system before Logging in
Flow of event	1. The app display the login form including password form. 2. user inserts username and password and click login button. 3. User Login in to the app
Post condition	The system displayed homepage
Alternative flow-of events	If the user inserts wrong username or password then the system will generate an error message.

Table 3.1: User description for Log in

Description 2	
Use case name	Take course
Use case number	2
Use case description	It contains different types of lessons of Geez language by classifying them in Category
Participating actor	Users
Pre-condition	The user should pass the login page to get to home page where Courses are found in
Flow of event	1. The course icon will be displayed in the Homepage. 2. Course have to be clicked. 3. The App displays categories of the course. 4. Each category has different types of courses.
Post condition	Categories of courses with their lessons will be displayed.
Alternative flow-of events	Null

Table 3.2: use case description for Take course

Description 3	
Use case name	Take quiz
Use case number	3
Use case description	It contains questions which helps the user to know how much they learn from the course.
Participating actor	Users
Pre-condition	The user should login to the system
Flow of event	1. The Quiz icon will be displayed in the Homepage. 2. Quiz icon have to be clicked. 3. Categories of the quizzes will be displayed on the page. 4. Selecting the category. 5. Questions will be displayed which are related with the selected category. 6. Choosing the answer and click the next button.

Post condition	It gives mark or tells the result that how many questions the user answered correctly from the given questions.
Alternative flow-of events	It will display a message that tells to answer a question first to get or pass to the next question.

Table 3.3: use case description for Take quiz

Description 4	
Use case name	Translate
Use case number	4
Use case description	It will translate every geez letter, word and sentences to Amharic, Afaan Oromoo ,English.
Participating actor	Users
Pre-condition	The user should login, get to course page, choose category of the course
Flow of event	1. The user has to enter to course page. 2. The course category should be selected. 3. Lessons that are found in the category will be displayed. 4. Click on the Letter or Word or Sentence which is found in the category, to get the translation. 5. The translation will be displayed in Amharic, Afaan Oromoo, English and Sign languages.
Post condition	The user gets the meaning of the selected letter or word or sentence on the selected category.
Alternative flow-of events	Null

Table 3.4: Use case description for Translate

Description 5	
Use case name	Update course category
Use case number	5
Use case description	To add new course such as new categories.
Participating actor	Admin
Pre-condition	The authorized admin should use the firebase where the database is stored and can access it.
Flow of event	1. The Admin have to use firebase which has previously stored database. 2. The Admin should create another category on the Category list in the database 3. The admin must save the change.
Post condition	The newly added courses will be added on the Application and the users can get them while they reload their app using internet connection in the online content.
Alternative flow-of events	Unauthorized admin will get error message while trying to access the database of the system

Table 3.5: use case description for update Course

Description 6	
Use case name	Update word
Use case number	6
Use case description	To change the application to its better version which has new features or interfaces
Participating actor	Admin
Pre-condition	Checking the admin is authorized to get access and then getting to the firebase.
Flow of event	1. The Admin have to use firebase which has previously stored database. 2. The database has to be edited. 3. The admin must save the change. 4. The previously existing App will be replaced by the downloaded upgraded version of the app.

Post condition	The updated word will be available on the online content.
Alternative flow-of events	Unauthorized user will get error message while tries to access the firebase.

Table 3.6: use case description for Update feature

3.5 Object model

3.5.1 Data dictionary

Object model is a description of an object-oriented architecture, including the details of the object structure, interfaces between objects and other object-oriented features and functions.

Class	Attribute	Operation	Description
User	Name Id User Level	VerifyLevel()	The authenticate person.
Login	Email Password	VerifyPassword() Displayerror()	Verify the authorized user
Admin	Name Id Email	Create() Update()	Adding and update new features to the system.
Course	Course Name Course Id Course type	ProvideCourse()	It provides Courses which are classified in categories.
Quiz	Quiz Name Quiz Id Quiz Level	Text() Result() Award()	It display questions and finally gives result and star.
Translation	Language	Translate()	It translates the Geez language to Amharic, Afaan Oromoo and English
Text	Course Name Course Id	ProvideTextCourse() ProvideTextQuiz()	The Courses and Quizzes are

	Course type Quiz Name Quiz Id Quiz Level		displayed in text format.
Voice	Course Name Course Id Course type Quiz Name Quiz Id Quiz Level	ProvideVoiceCourse() ProvideVoiceQuiz()	Letters, words and sentences of the courses and Quizzes are recorded in voice and provided in Voice format.

Table 3.7: Data Dictionary for Classes

3.5.2 Class diagram

The Class diagram captures the logical structure of the system; the classes and things that make up the model. It is a static model, describing what exists and what attributes and behavior it has, rather than how something is done. Class diagrams are most useful to illustrate relationships between classes and interfaces.

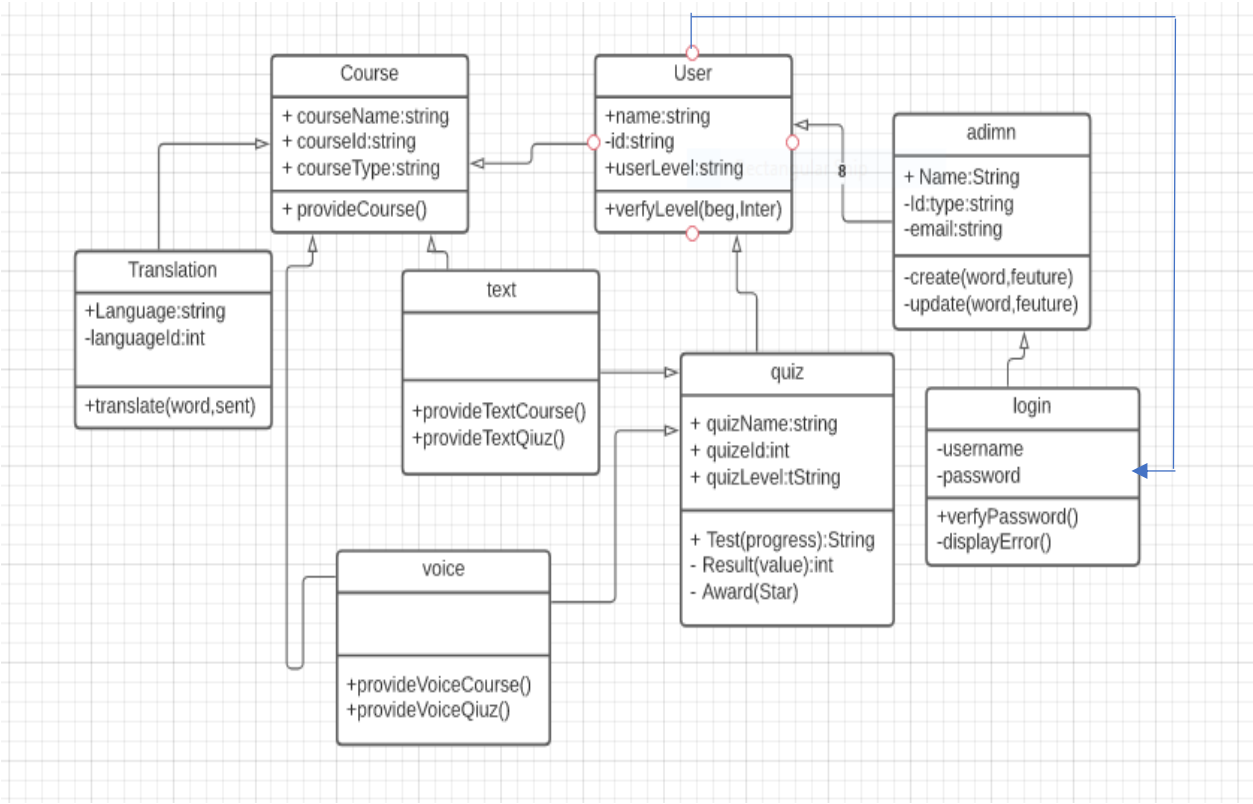

Fig 3.2: Class diagram

3.5.3 Sequence diagram

A sequence diagram in a UML is a kind of interaction diagram that shows how processes operate with one another and in what order. 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.

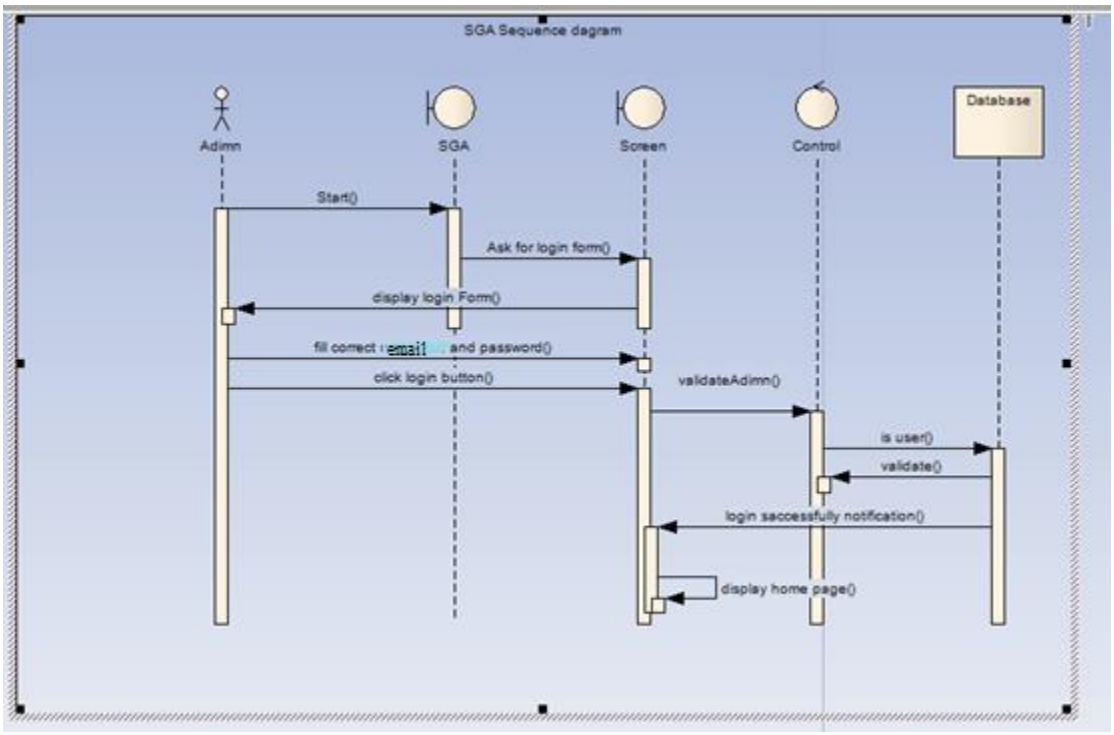

Fig 3.3: Sequence diagram of Log in

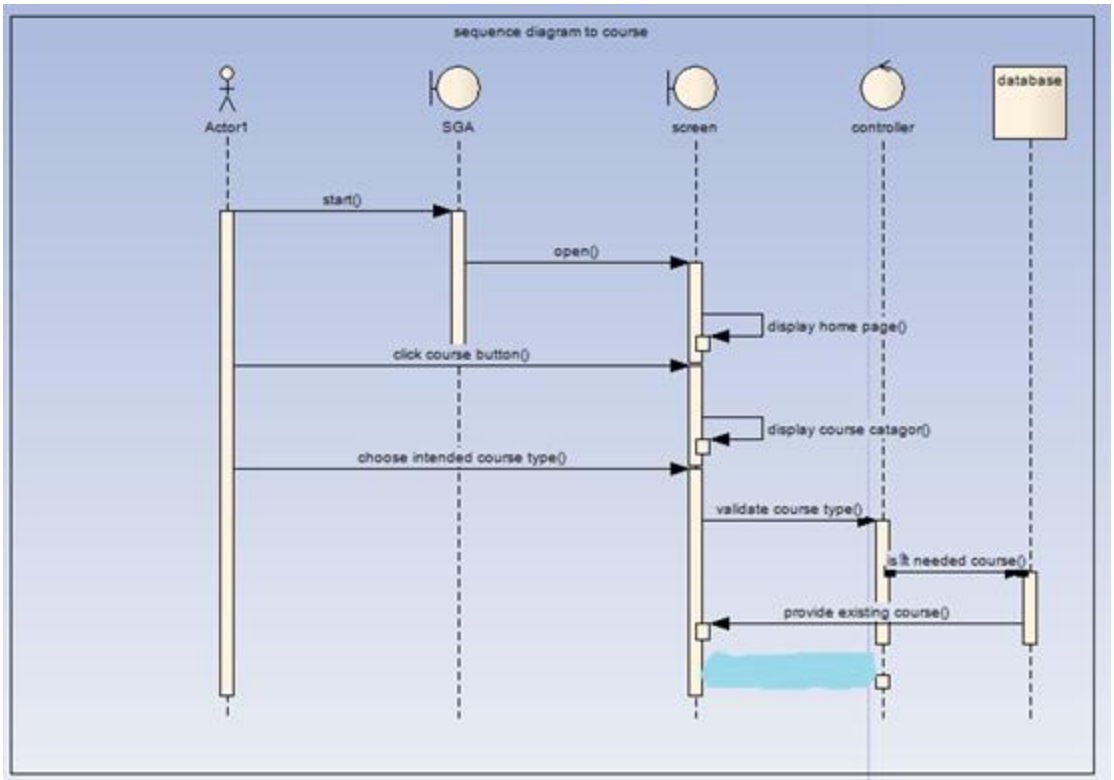

Fig 3.4: Sequence diagram of Take courses

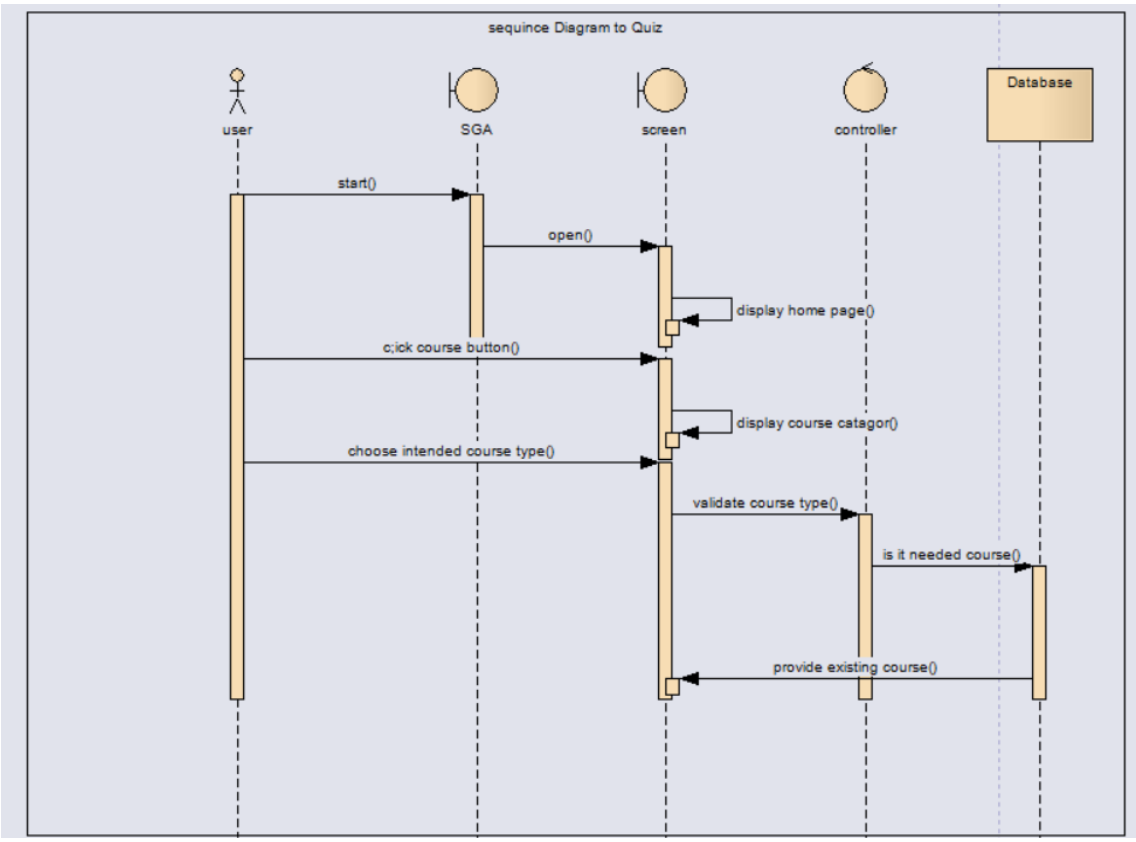

Fig 3.5: Sequence diagram of Take quiz

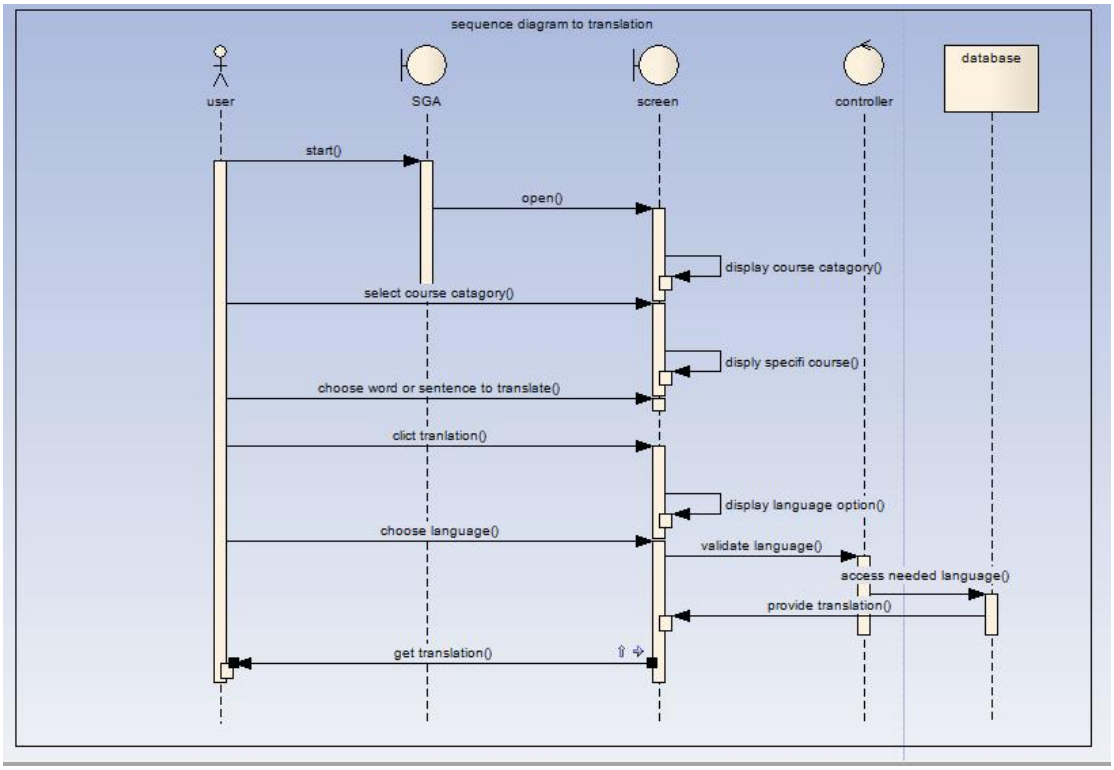

Fig 3.6: Sequence diagram of Translate

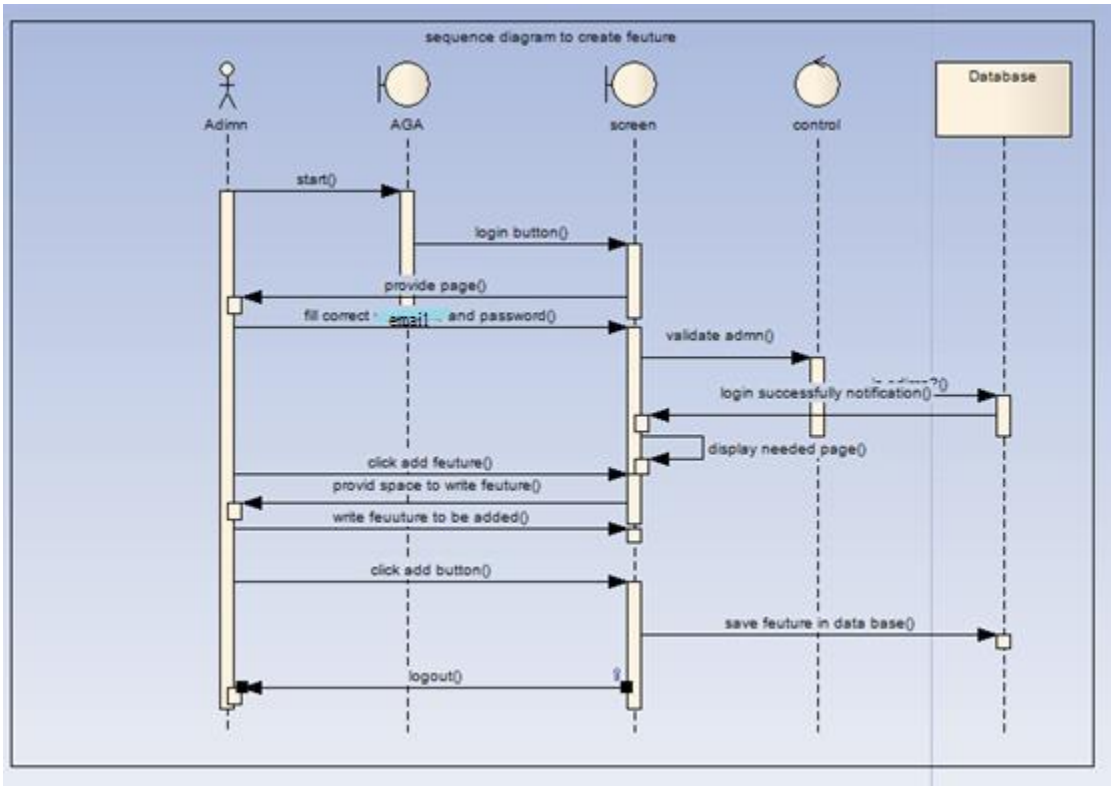

Fig 3.7: Sequence diagram of Update course category

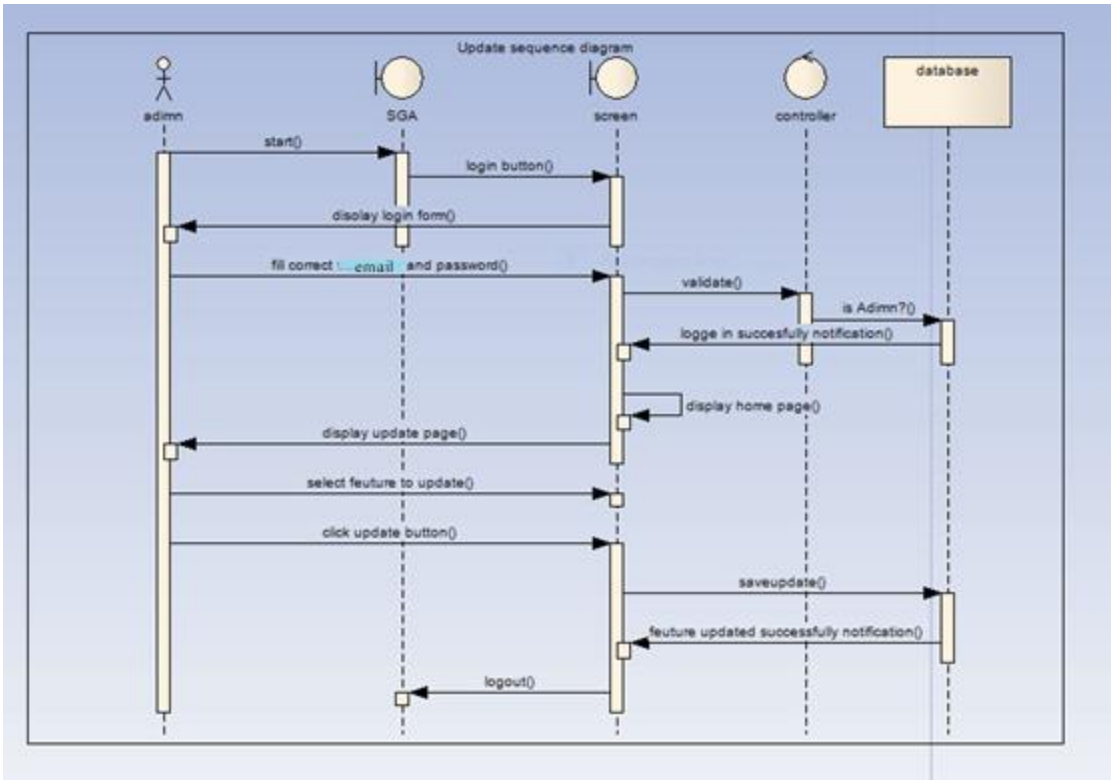

Fig 3.8: Sequence diagram of Update word

3.5.4 Activity diagram

An activity diagram describes a system in terms of activities. Activities are states that represent the execution of a set of operations. The completion of these operations triggers a transition to another activity. Activity diagrams are similar to flowchart diagrams in that they can be used to represent control flow (i.e., the order in which operations occur) and data flow.

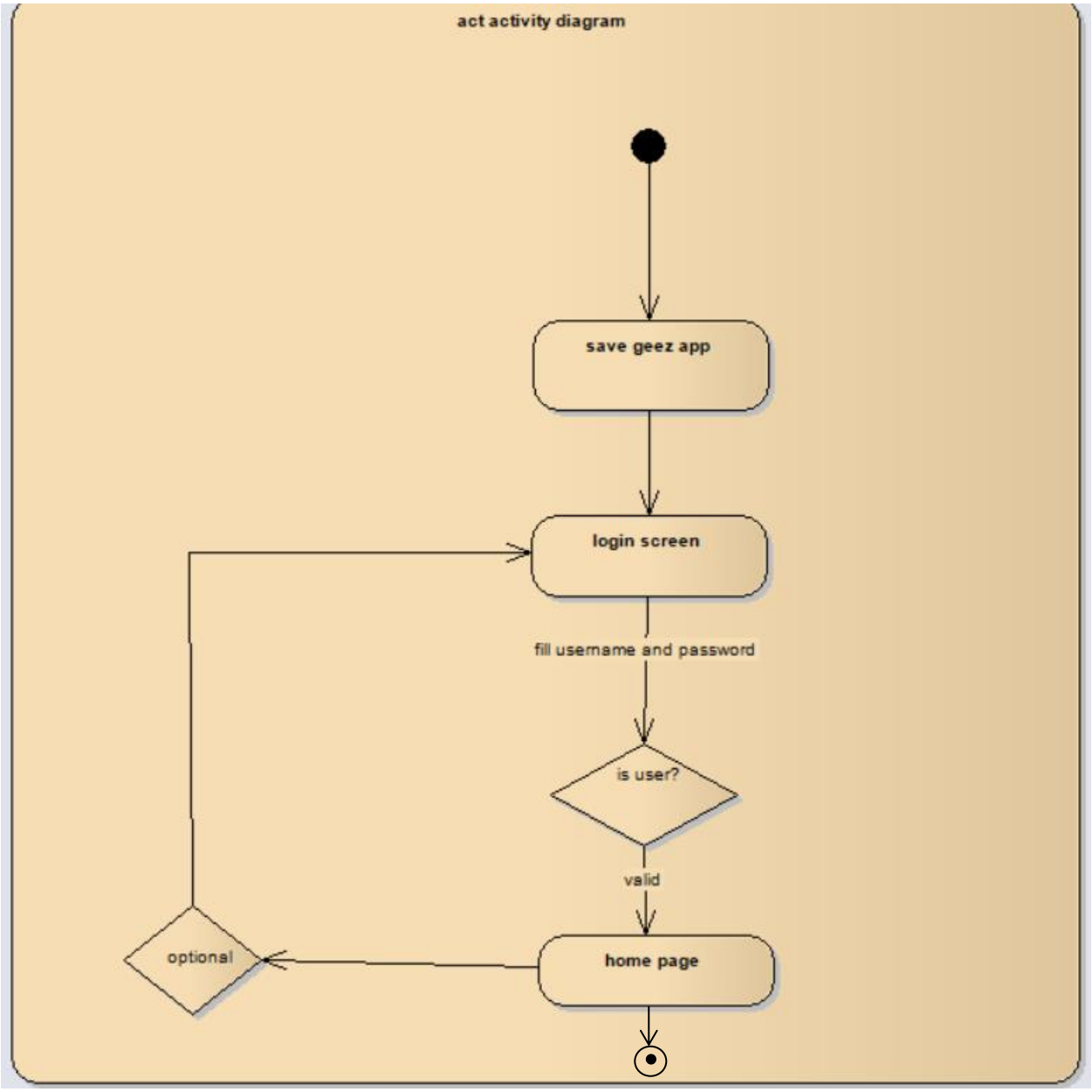

Fig 3.9: Activity diagram for login

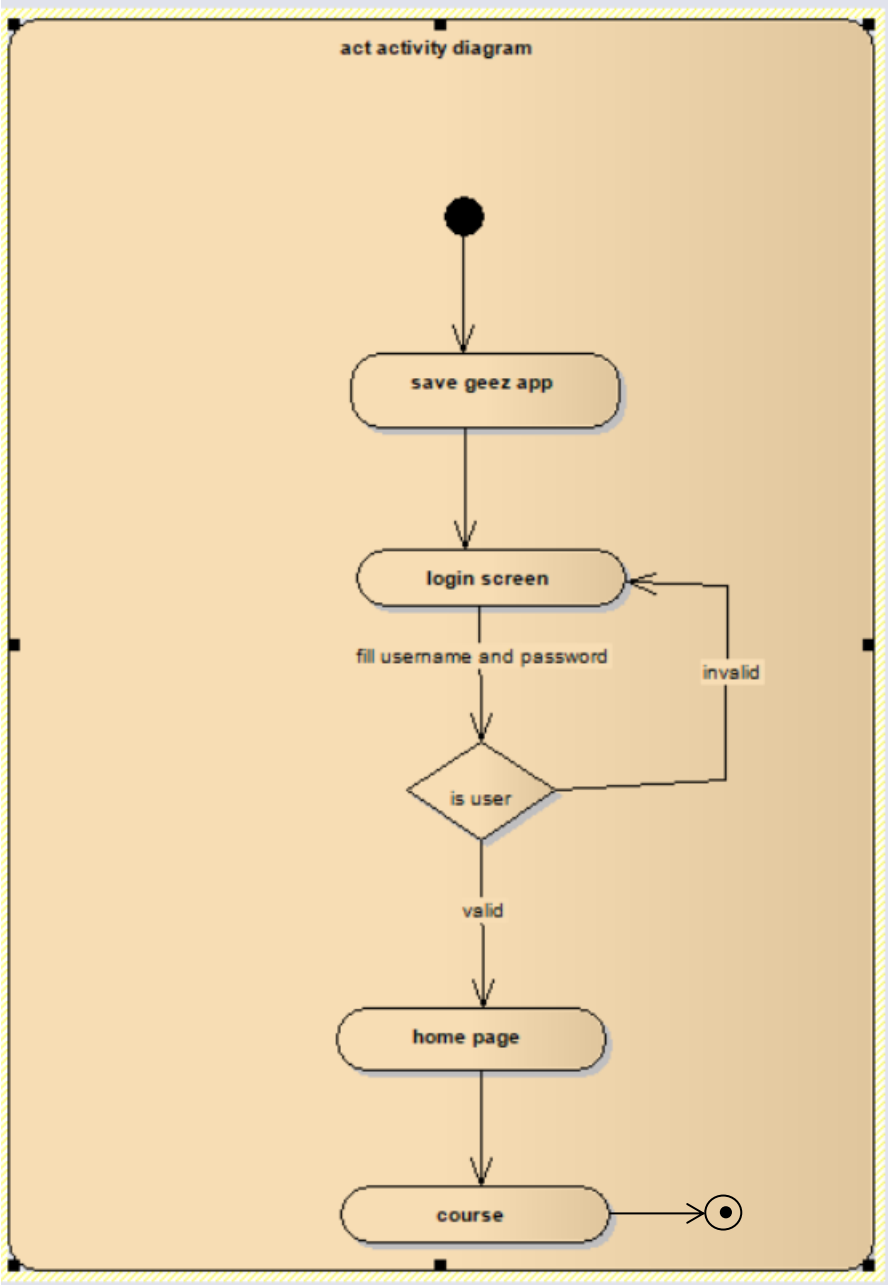

Fig 3.10: Activity diagram for course

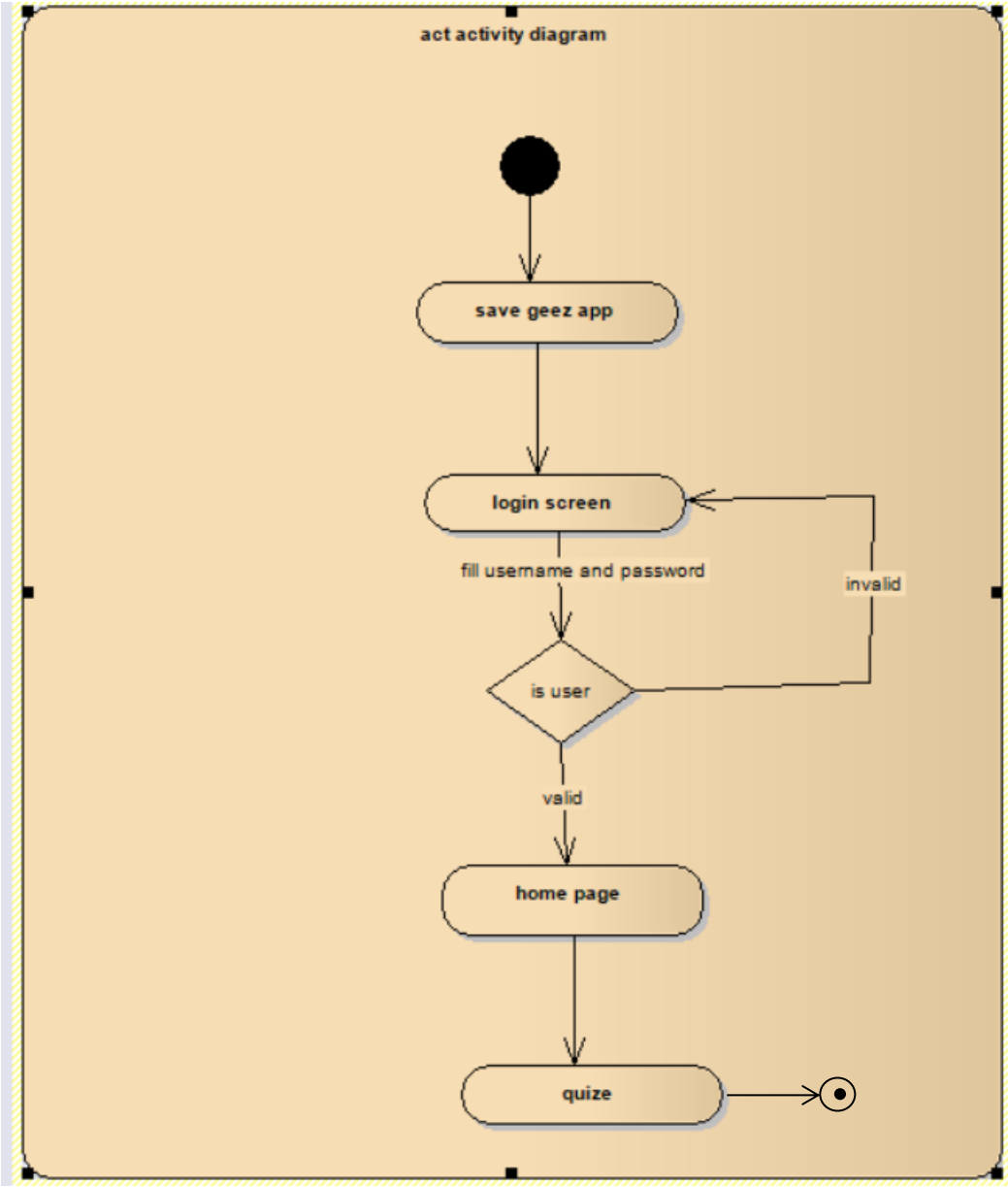

Fig 3.11: Activity diagram for quiz

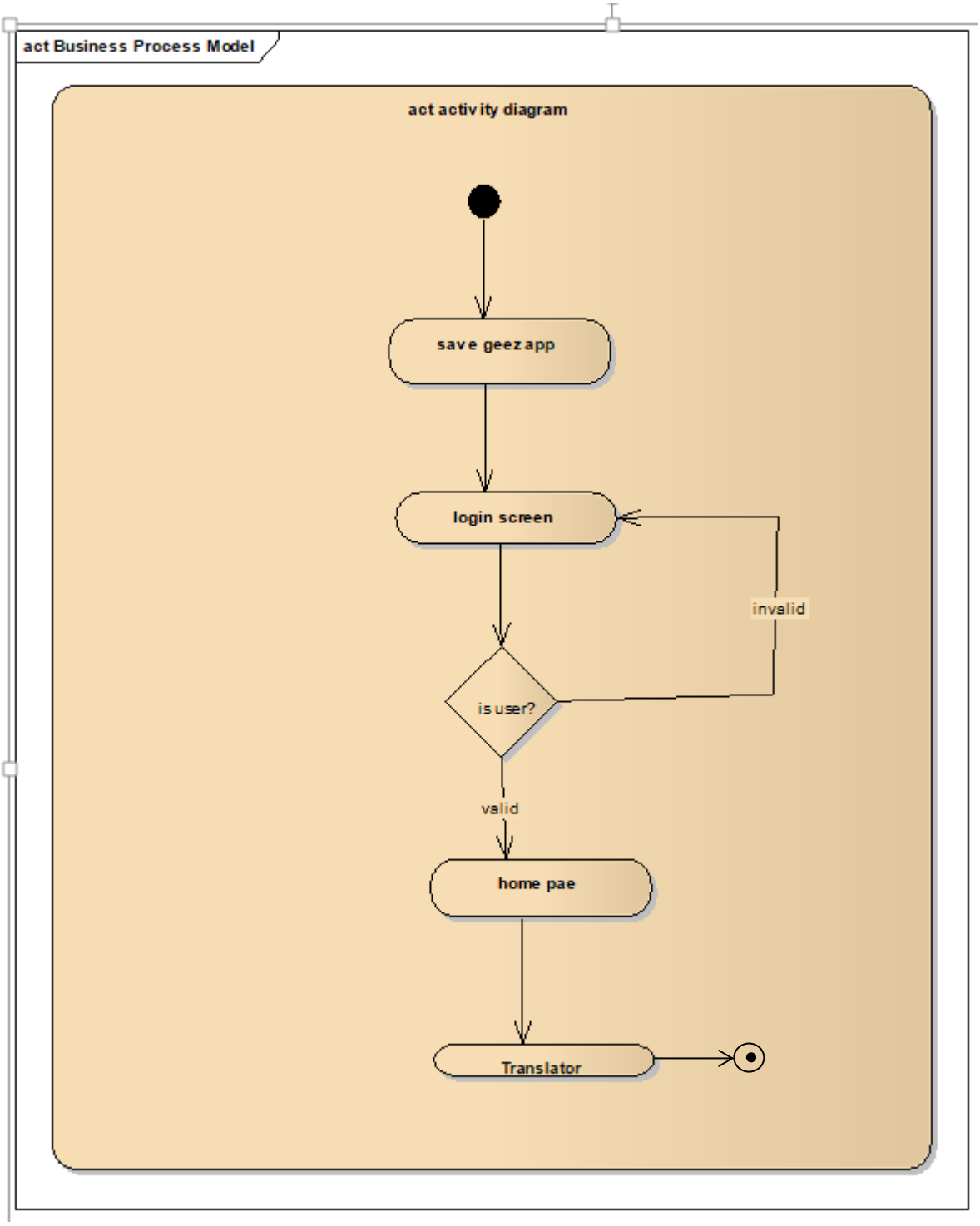

Fig 3.12: Activity diagram for translation

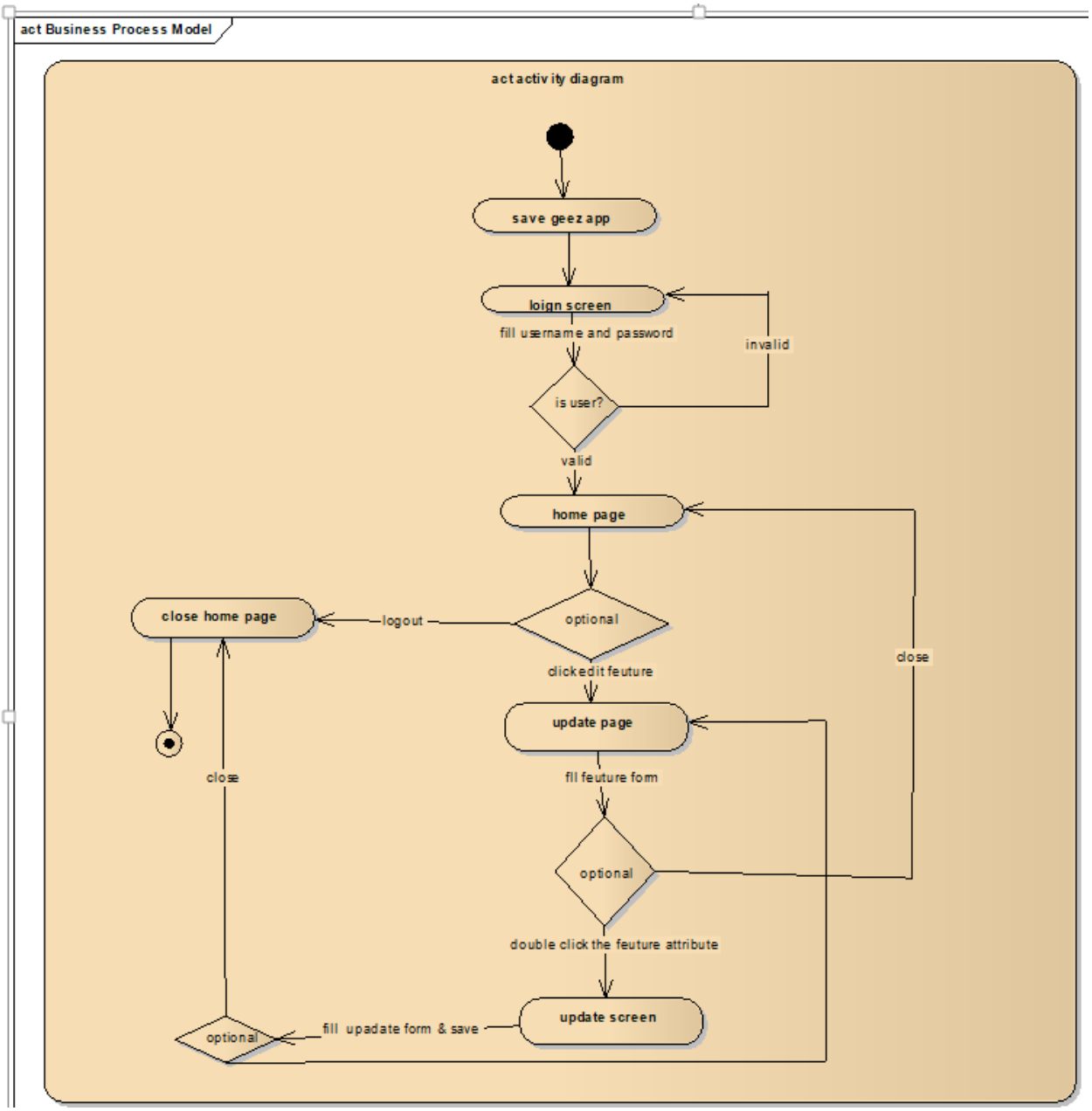

Fig 3.13: Activity diagram for update

3.5.5 State chart diagram

State diagrams are used to give an abstract description of the behavior of a system. This behavior is analyzed and represented in series of events that could occur in one or more possible states

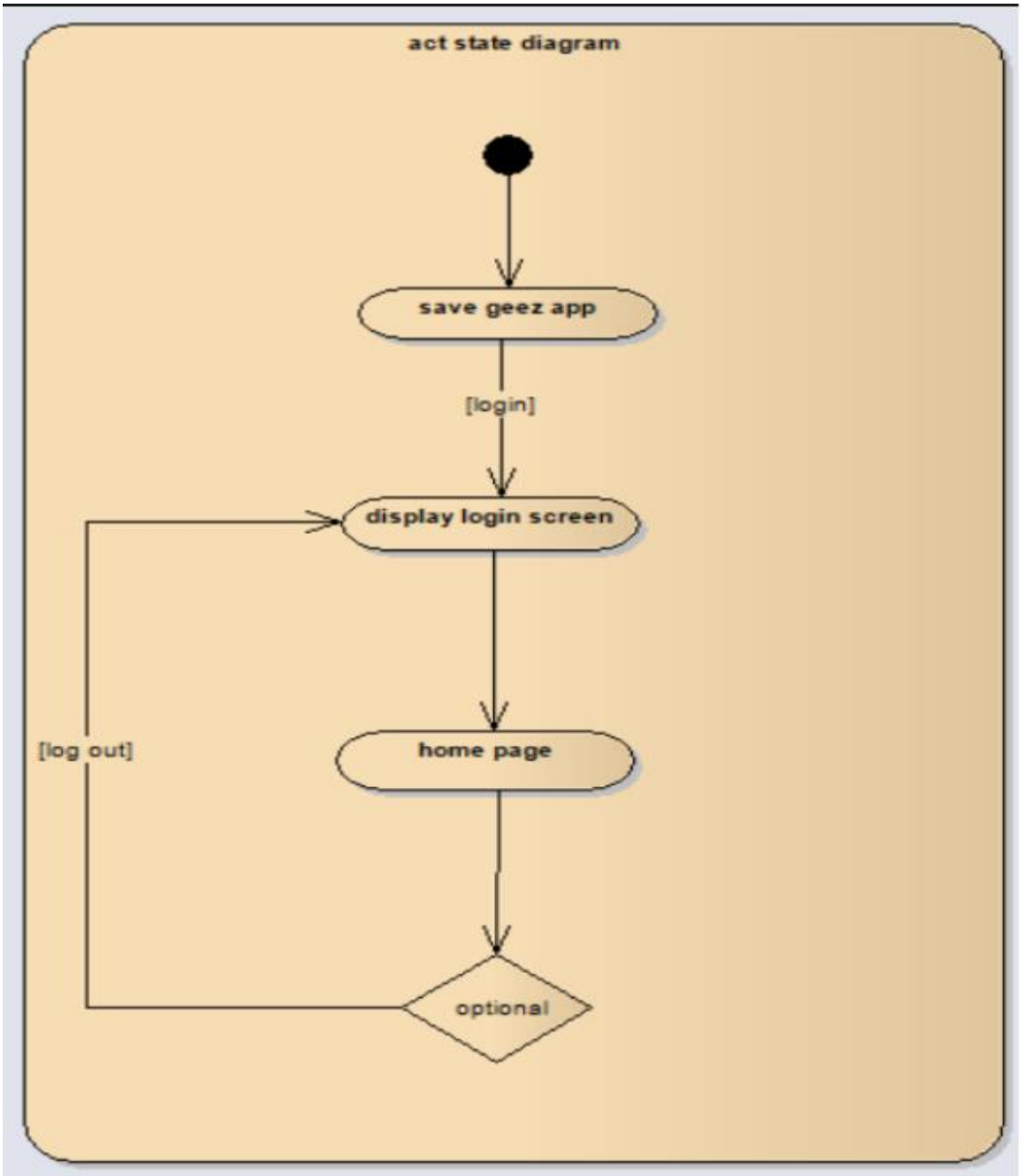

Fig 3.14: State diagram for login

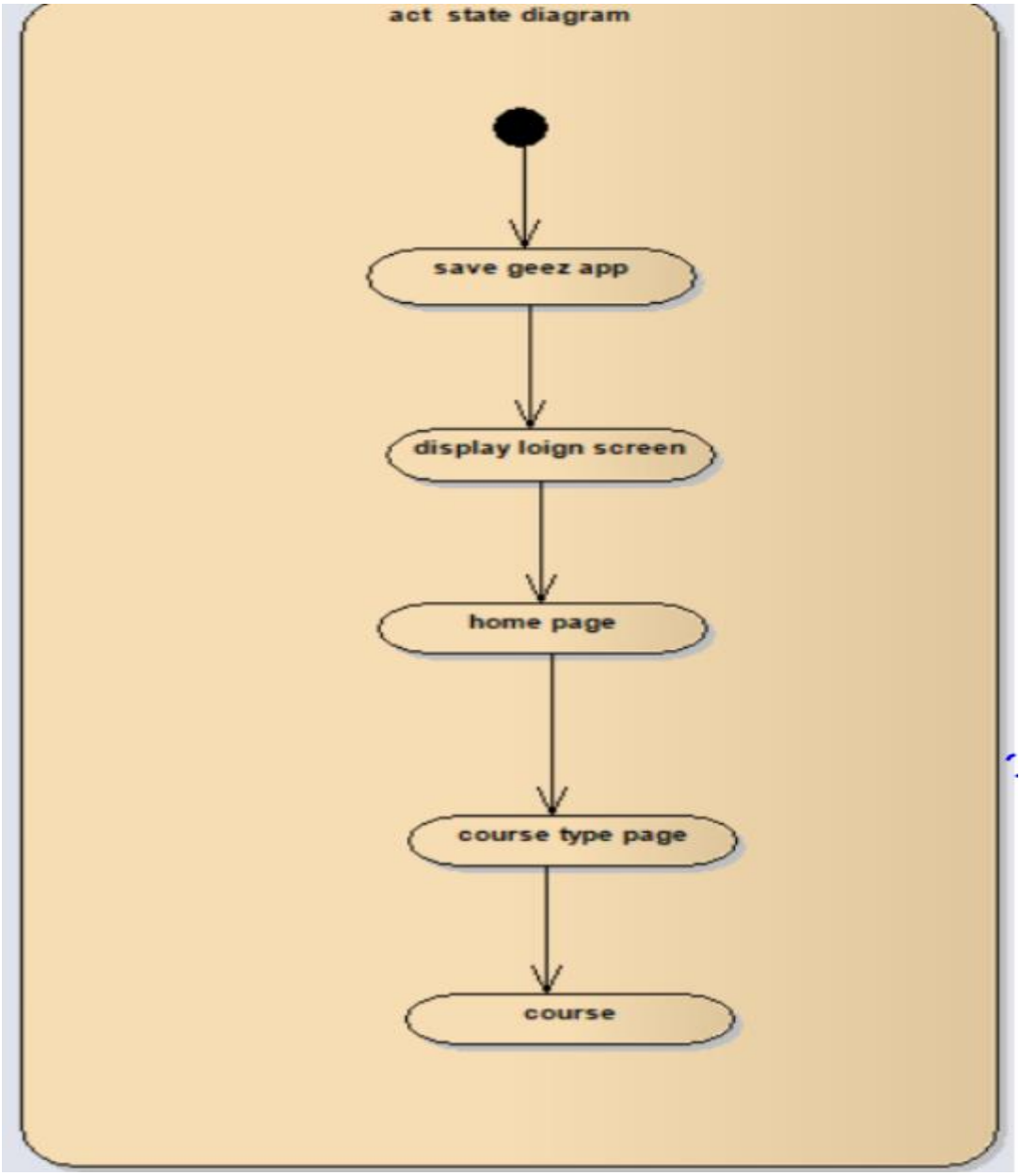

Fig 3.15: State diagram for course

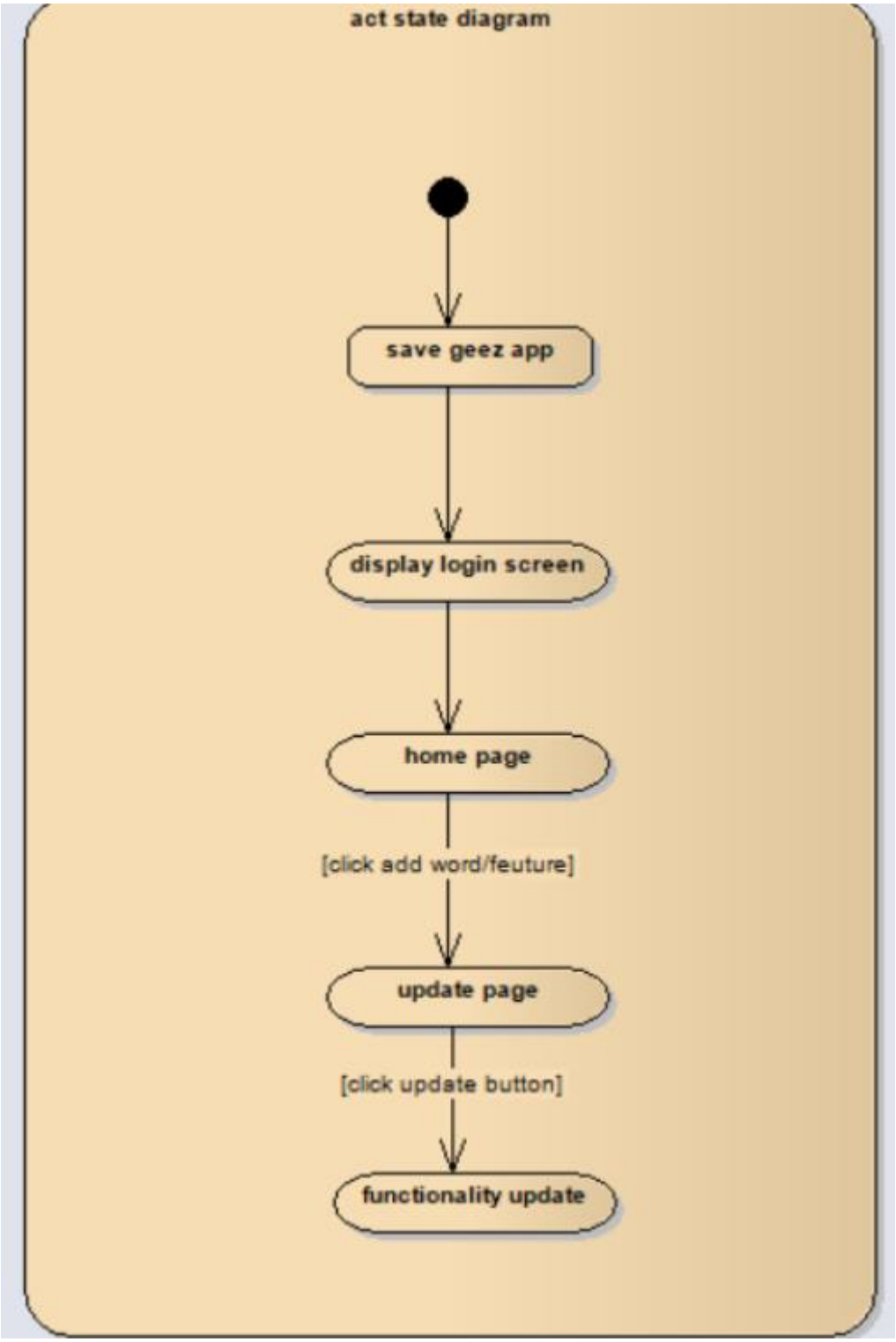

Fig 3.16: State diagram for update

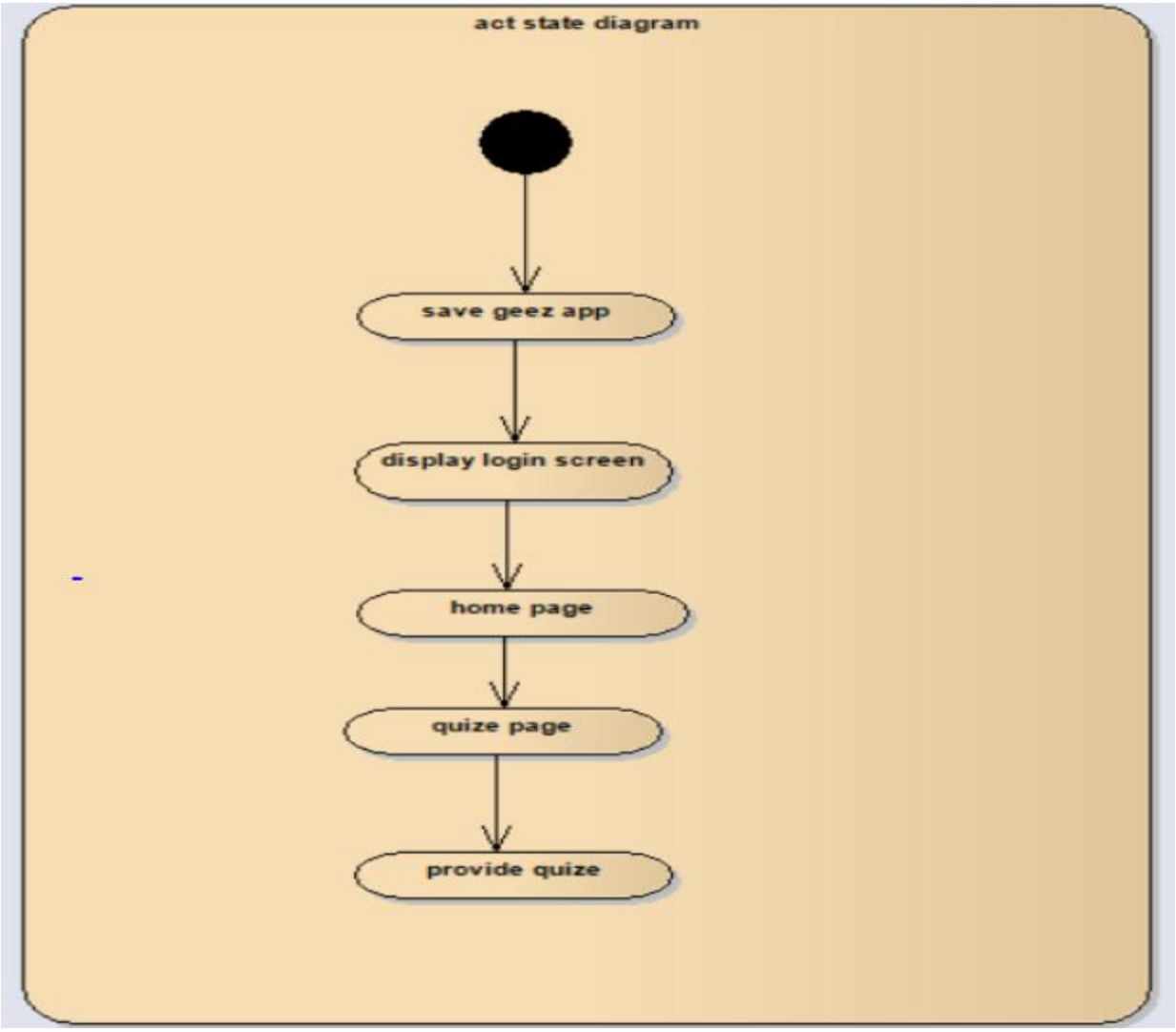

Fig 3.17: State diagram for quiz

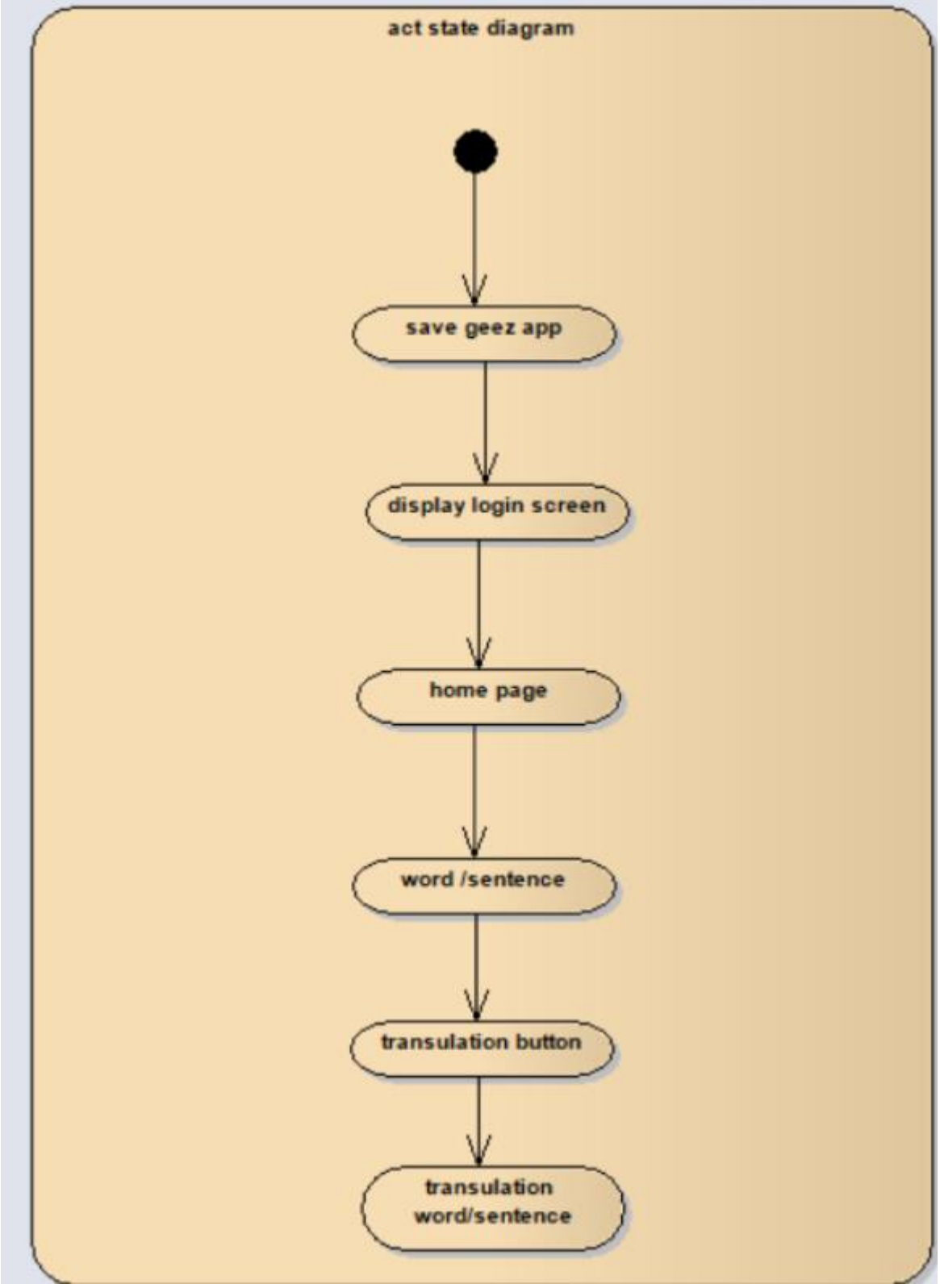

Fig 3.18: State diagram for translation

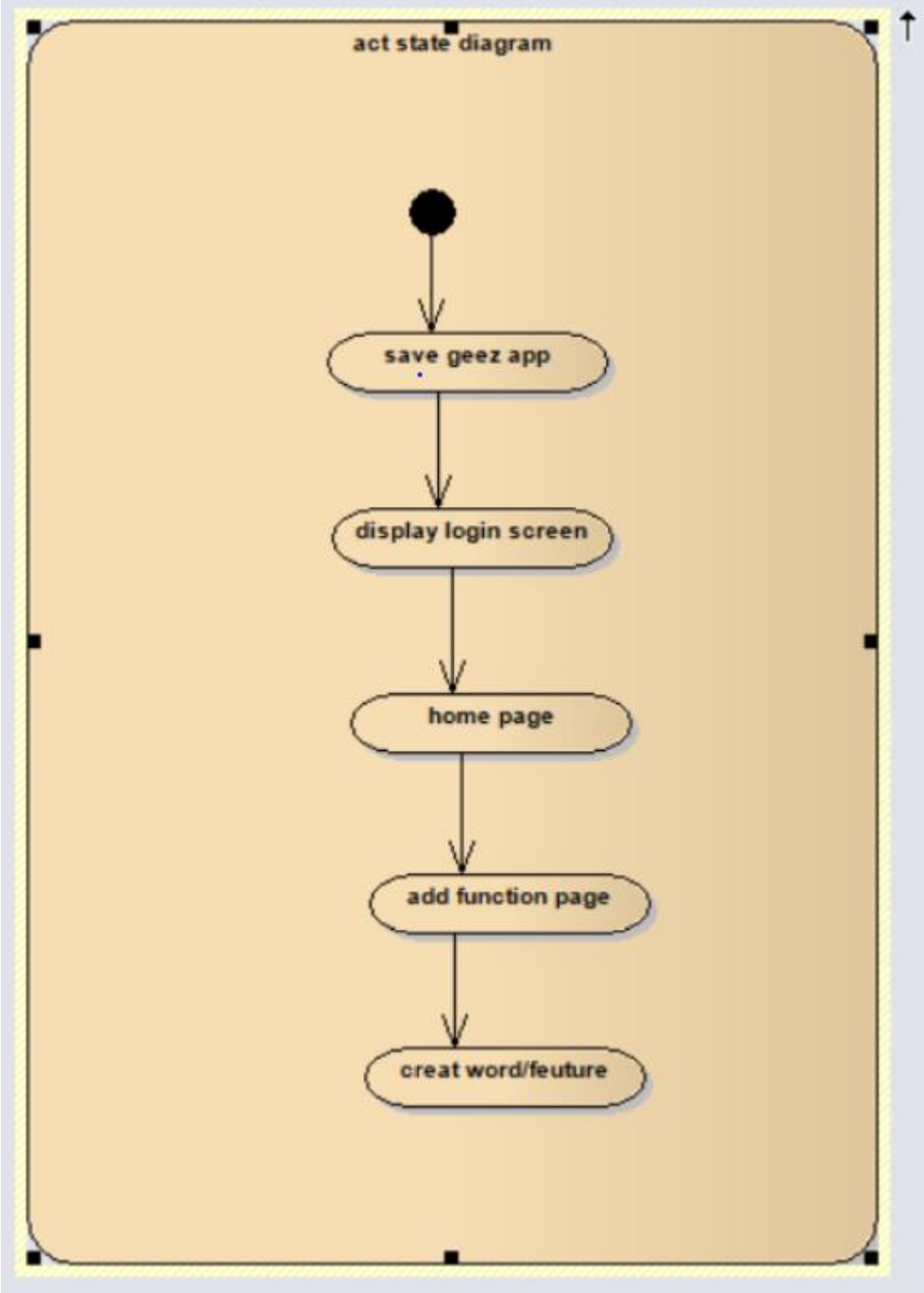

Fig 3.19: State diagram for create

CHAPTER 4

System Design

4.1 Overview

This is the system design document to the Save Geez Learning aid system for English, Amharic and Afaan Oromoo Speakers and sign language users all around the world. The document includes the design goal, the proposed system design and object design.

4.1.1 Purpose of the system design

This document describes the design issues of the overall system. It provides the complete architectural overview of the proposed system. It is intended to capture and express the significant architectural decisions which have been made on the system as well as to obtain the information needed to manage all citizens of the country.

4.1.2 Design goal

The design goals describe the qualities of the system that are derived from the non-functional requirements which can lead to decisions of developers. The designing part is very important so as to make the implementation or the coding part very easy. The different types of the system modelling techniques that are used for the implementation of the system such as deployment and component modelling are show in detail. Not only the system modelling techniques but also some system design techniques such as system decomposition design are cover in detail in this phase. These goals can be inferred from the non-functional requirements. These are:

- Performance
- Dependability
- Maintenance
- End user

4.1.2.1 Performance

In order for the Save Geez Learning aid Application system is to make Distributed system than the system to give the services more than 5000 users per month the system satisfy the following condition.

Response time: - Depending on the network connection that the user machine has the system is going to interact and respond to user's request in a maximum of a second, if the user is just viewing the pages, but if the user's request requires the processing of the data base, like searching for Word, is going to take an average of 1-5 seconds of communication latency with the server system. And getting Courses is going to have a response time up to 2-4 seconds.

Memory: The client system requires an average of 20-30megabites of memory to install the Application.

4.1.2.2 Dependability

The Save Geez Learning Aid Application system should achieve the following dependability characteristics in order to resist crash and be available and reliable.

- **Robustness**: since the system is a Mobile Application system, that mainly use a menu driven entry there wouldn't be an input problem by the user side.
- **Availability**: Because of it is a mobile application and it is offline to use after the user signup and load the data using internet connection, the system is available 7 days a week and 24 hours a day.
- **Security**: - by the user side there is no security issue to be considered. Because of it is language learning application in which there is no data to be secured.
- **Reliability**: the information provided by the system is as reliable as it is presented on the page interface, and this is maintained by the persistent database.

4.1.2.3 Maintenance

In time of failure or need modification the system needs to be maintained. To be maintainable the system should 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 and the user can get the upgraded Version of the Application by upgrading the it from Play Store or App Store.

- **Modifiability**: if in the system some functionality or features requires to be modified, this modification must be done specifically to that function or page without affecting the overall system organization, but the user have to upgrade it to get the modified version.
- **Portability**: the system is developed to be viewed and retrieved from Android phones and iPhones.

4.2 Proposed system architecture

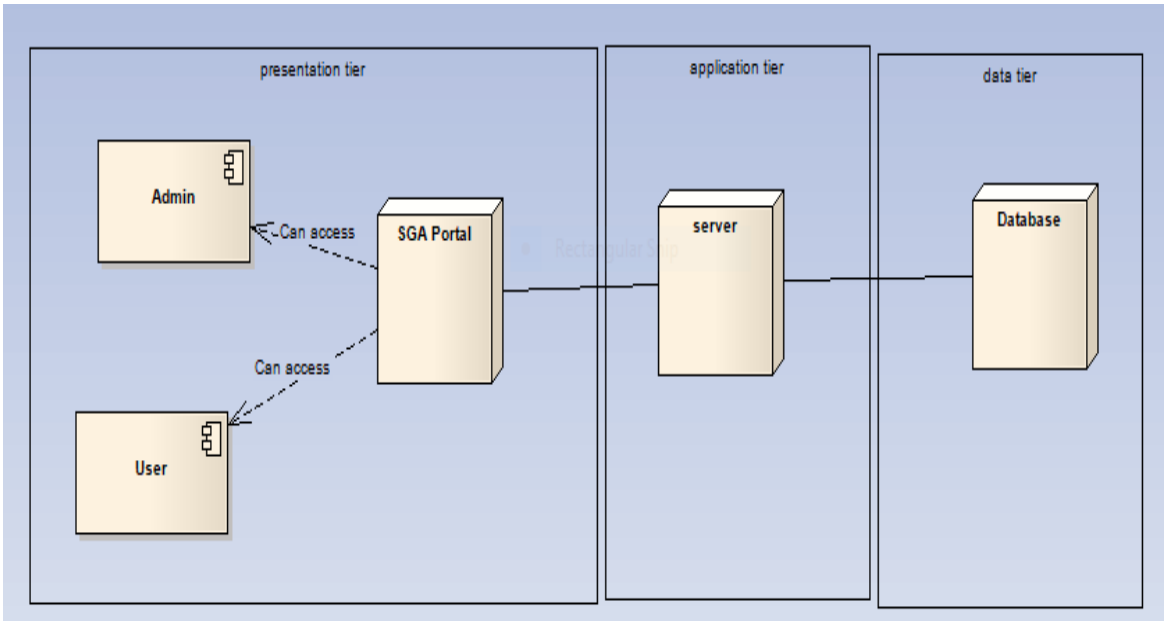

Fig 4.1: System Architecture

4.2.1 Overview

System design deals with persistent data management, which illustrate as to how persistent data (file, database, etc.) are stored and managed and at last Access control will be presented. Our system is three tires architectural it has client side, server and database.

- **Client side**: here in the client-side user interface will be existing.
- **Server side**: here the web servers to connect the data base application are found; mean that the application logic to perform the application by the client is found.
- **Data base**: here the data bases that store the information are found.

4.2.2 System process

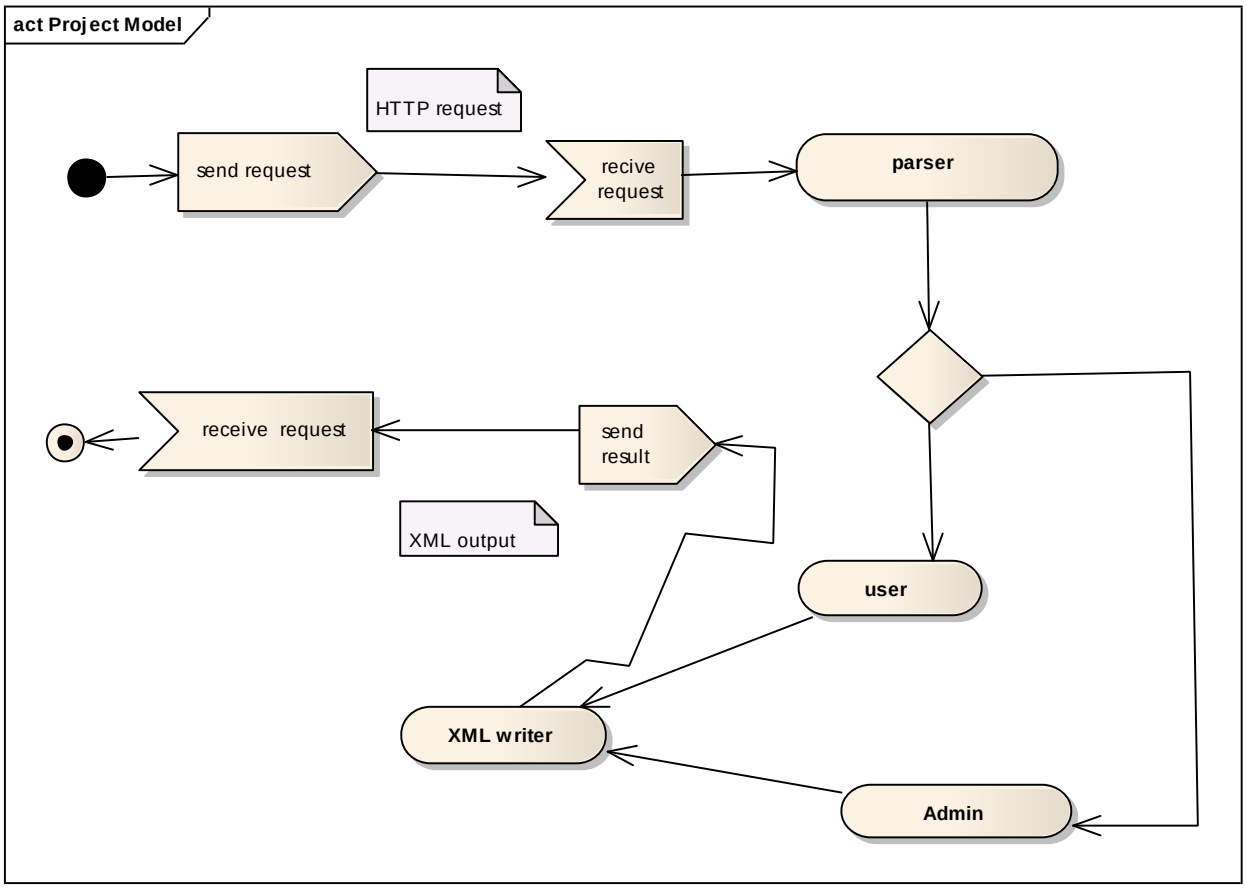

Fig 4.2: System process

Internet service portal to load initial SGA (Save Geez App)

Internet	**SGA Server**
	Database

Table 4.1: Internet Service portal

4.2.3 Sub-system Decomposition

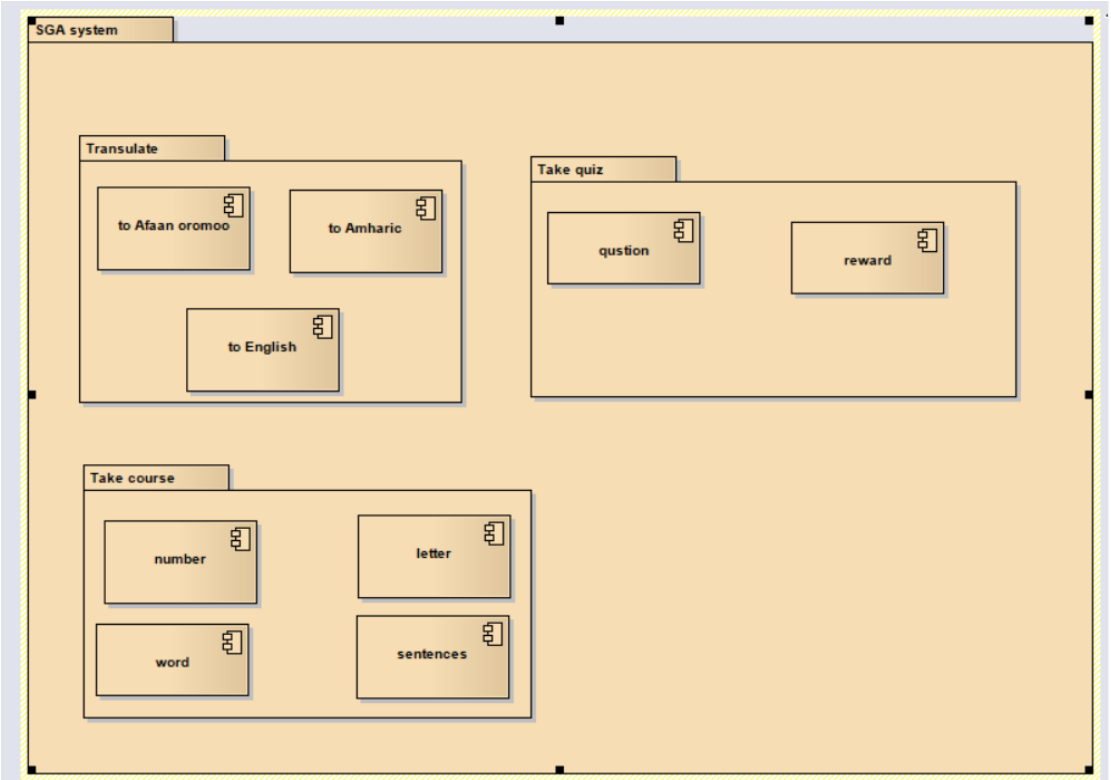

Fig 4.3: Sub-system decomposition

4.2.4 Hardware/Software mapping

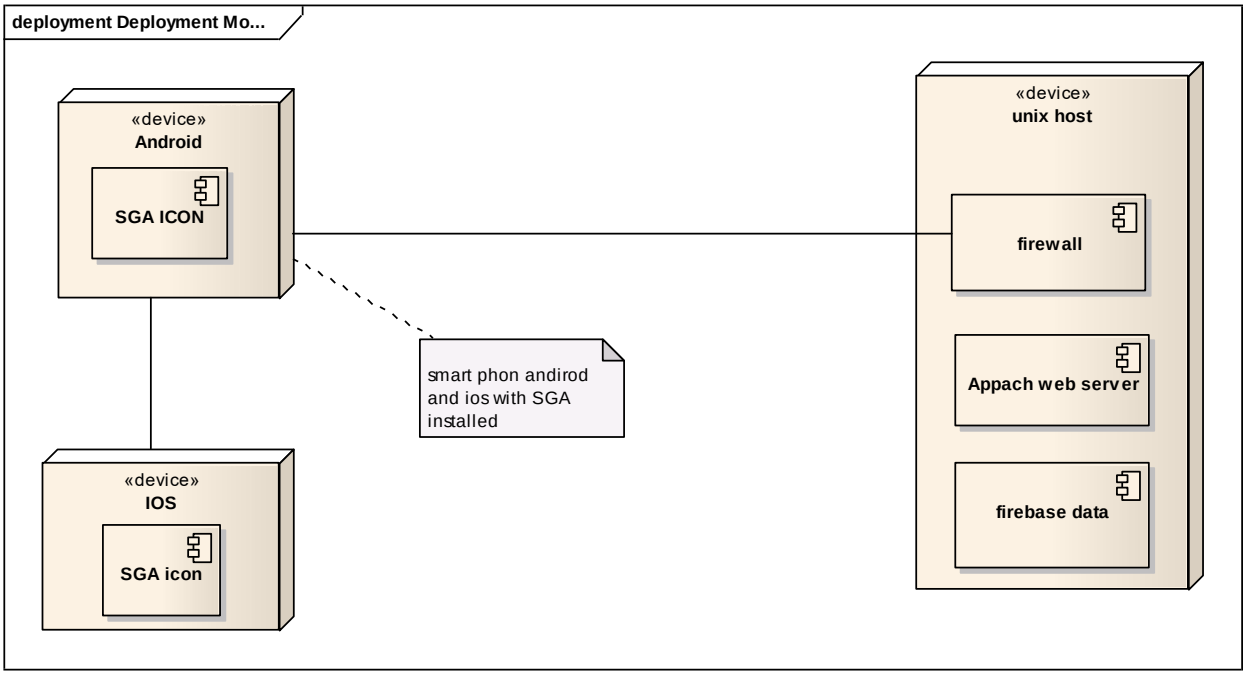

Fig 4.4: Software mapping of the system

4.2.5 Persistent data management

The purpose of this section is to show the mapping of the objects/classes of the system, identified during the analysis stage, in to the corresponding relational database.

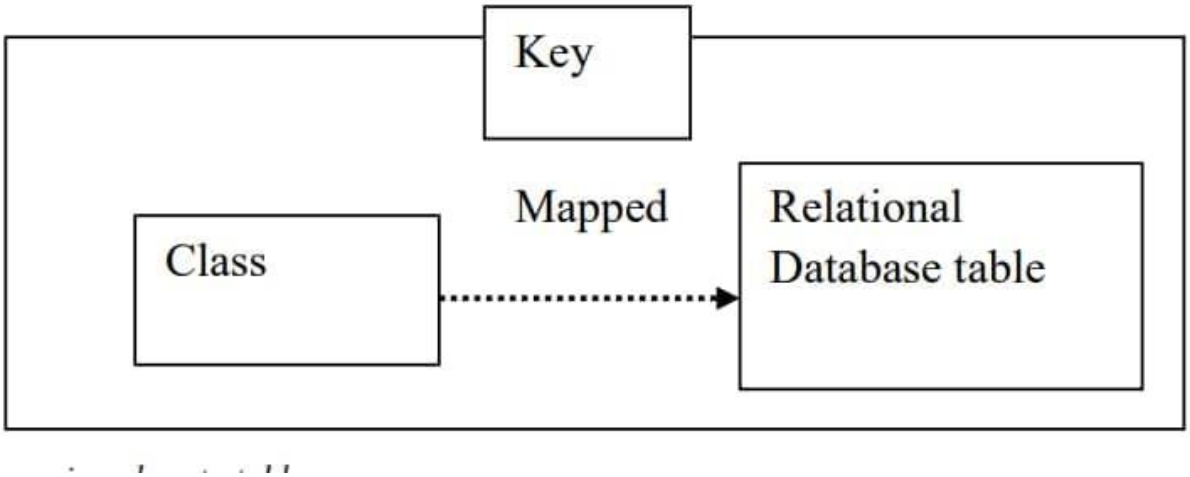

Fig 4.5: Mapping Class table

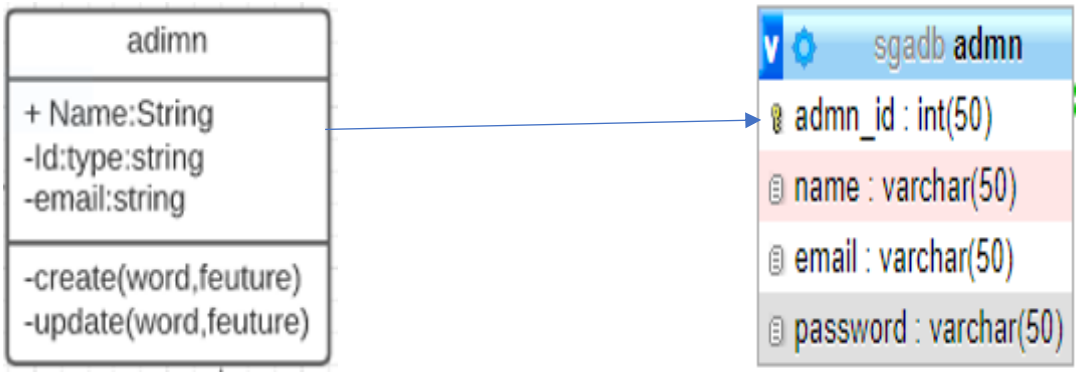

Fig 4.6: object admin mapping

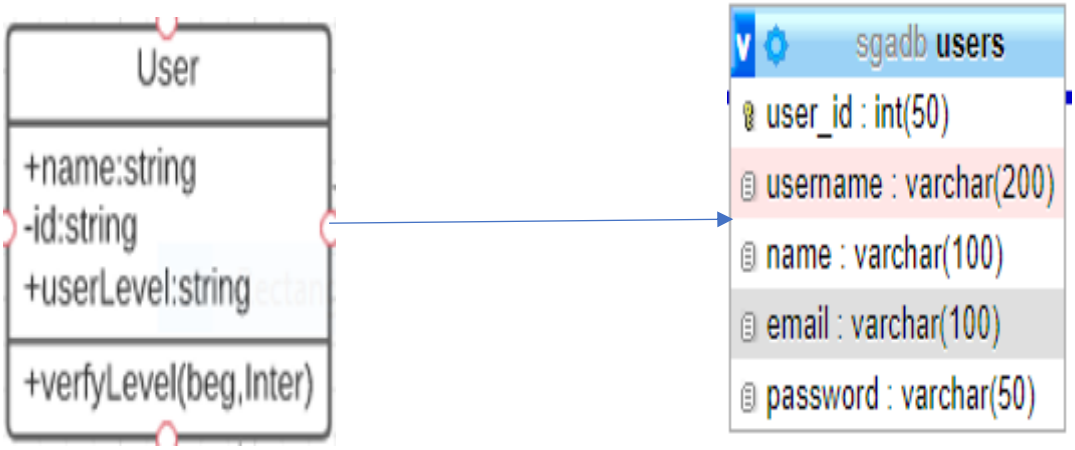

Fig 4.7: object user mapping

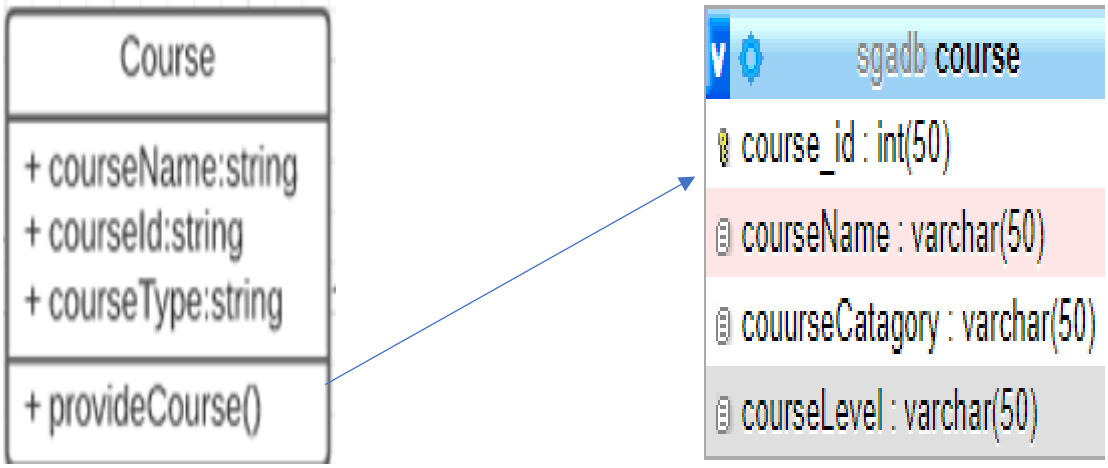

Fig 4.8: object course access mapping

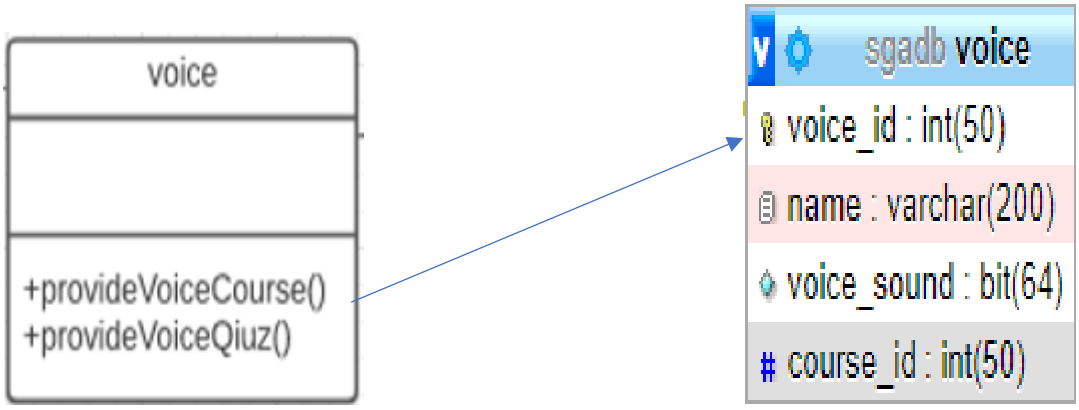

Fig 4.9: object voice communication mapping

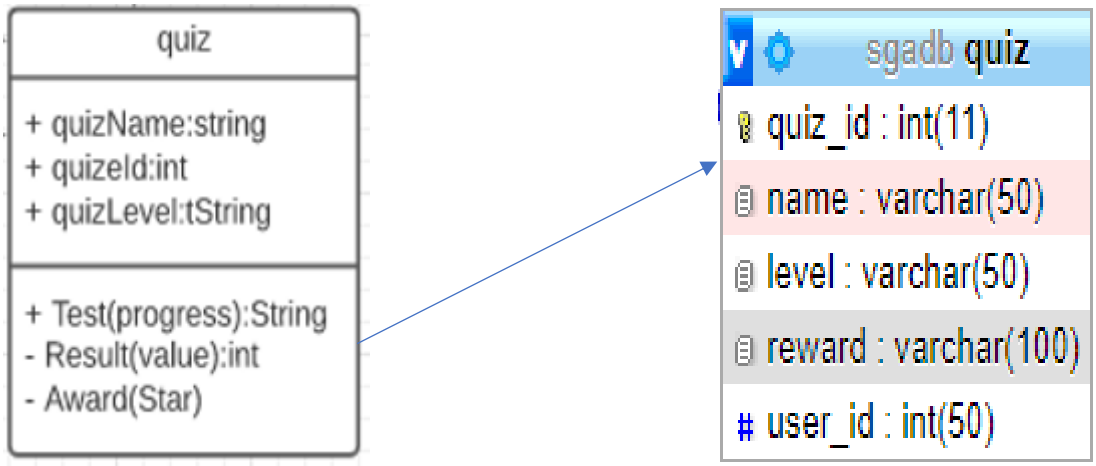

Fig 4.10: object quiz mapping

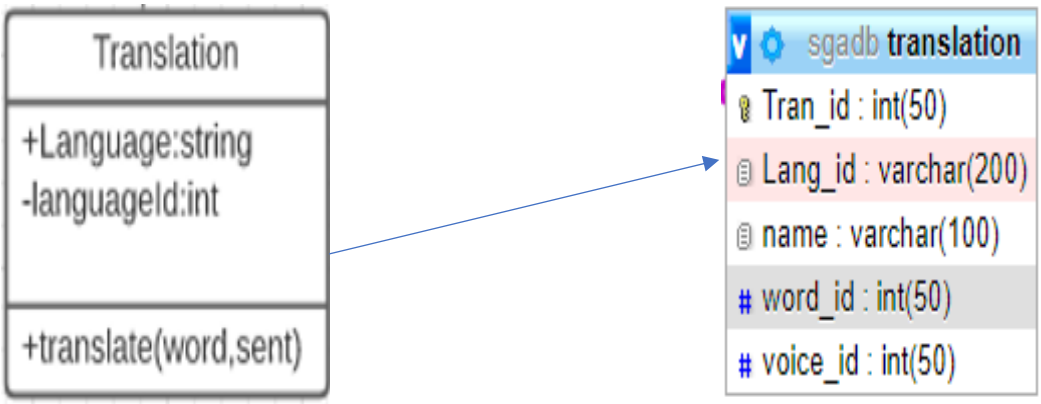

Fig 4.11: object language translation mapping

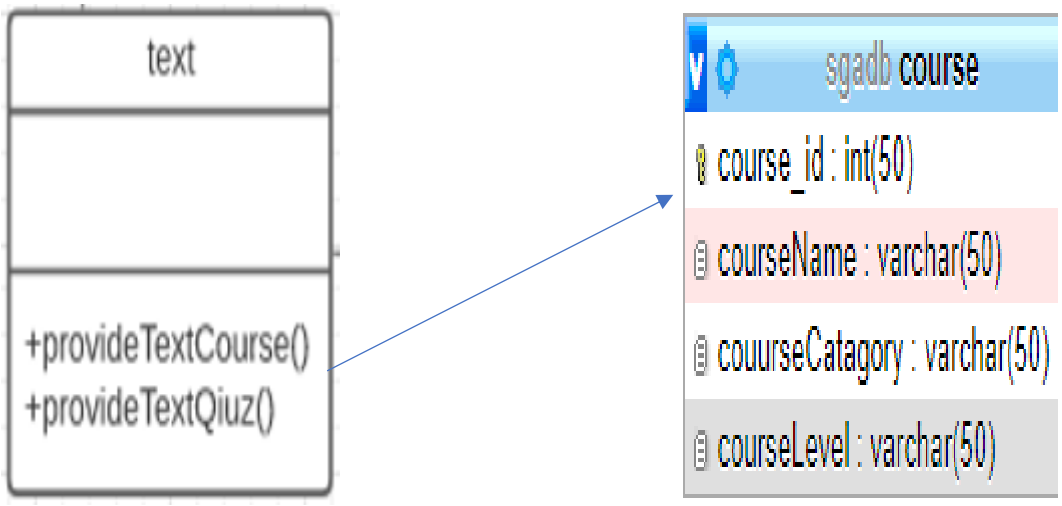

Fig 4.12: text course type mapping

4.2.6 Component diagram

It is Describes all components in a system, their interrelationships, interactions, and the interface of the system. It is an outline of the composition structure of components or modules.

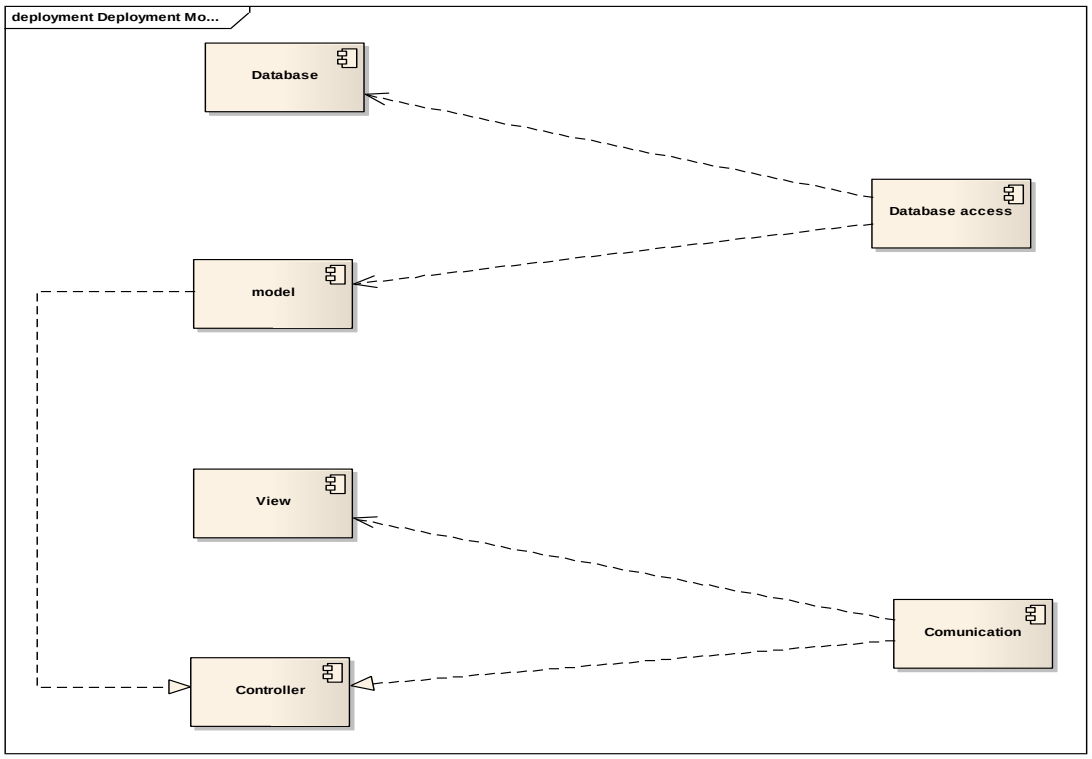

Fig 4.13: Component diagram

Component Description

- **SGA controller**

Description: This is a class responsible to authenticate a user

Role: it is responsible to check if the validity of a user trying to login to the system

- **Model**
  - **Module model**

Description: this is a class which is connected with the module table in the database and linked with the pages controller for further process.

Role: this class will be used to retrieve info from and to the module table and deliver to the page's controller class.

  - **Configuration model**

Description: it is a class which is used to connect he component controller and the database table for data exchange.

Role: is responsible to get access to the database table and return the data to the component controller for further process.

- **Communication**

Description: it is a class which is used to communicate the controller and view component for view process.

Role: is responsible to ger access of content to view through controller.

- **Database**

Description: it is class which is used to connect database and database and data base access control to exchange and update data.

Role: is responsible to access data from database and store it as needed.

- **View**

Description: it is class which is used to connect to communication class to view content through it.

Role: responsible to make available content to viewer.

- **Database accessor**

Description: it is class which is used to connect interface and database.

Role: responsible to retrieve content through interface.

4.2.7 Database design

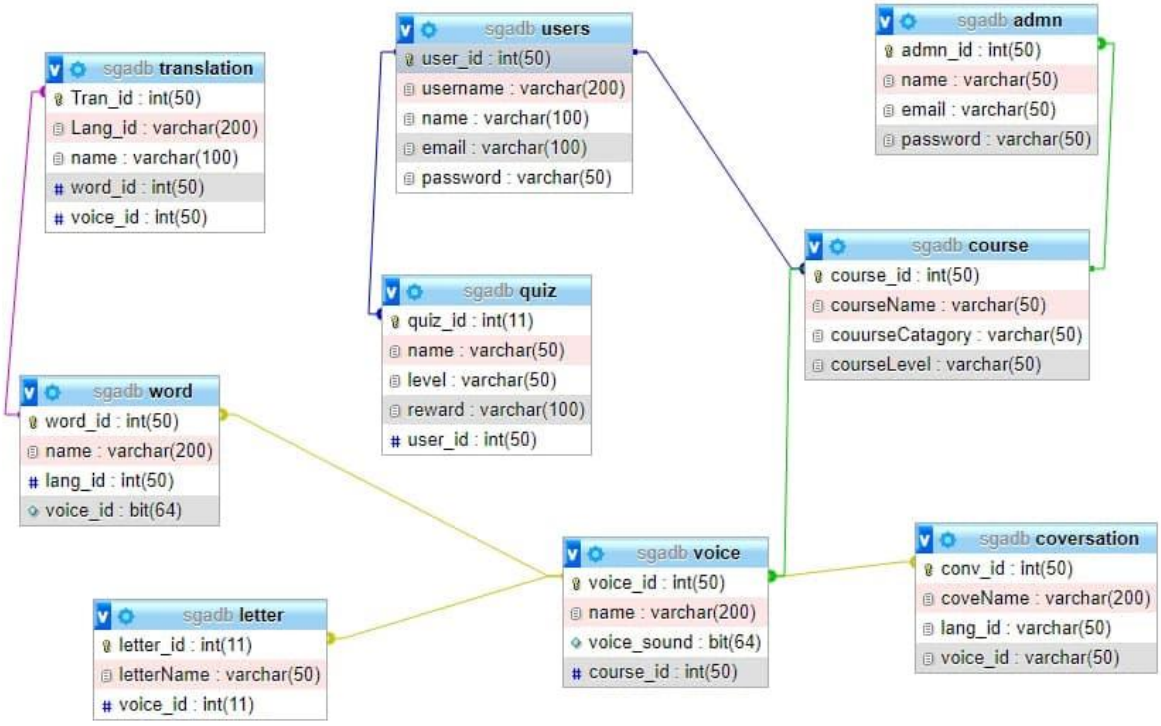

Fig 4.14: Design for the Database

4.2.8 Access control

The table below shows the global access table, describing the access relation between the actors, objects and operations in the system:

Actor	Update course category	Update word	Take course	Take quiz
Admin	UpdateCourseCatagory() UpdateWord() UpdateSentence()	Updateword()		
User			ViewCourse() TakeCourse() TakeQuiz()	Viewquiz() Takequiz() EvaluateLevel()

Table 4.2: Access Control

4.2.9 How User access our App

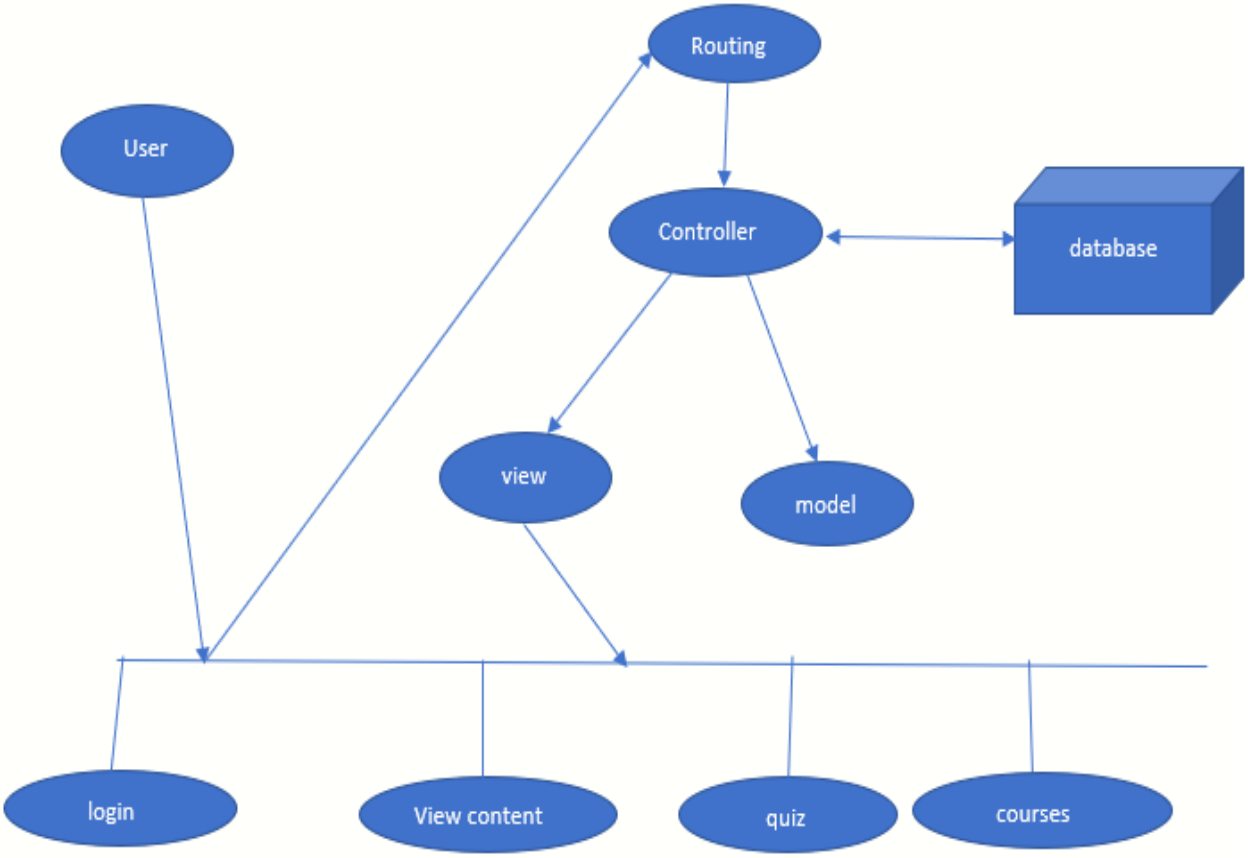

Fig 4.15: User access

4.2.10 User interface design

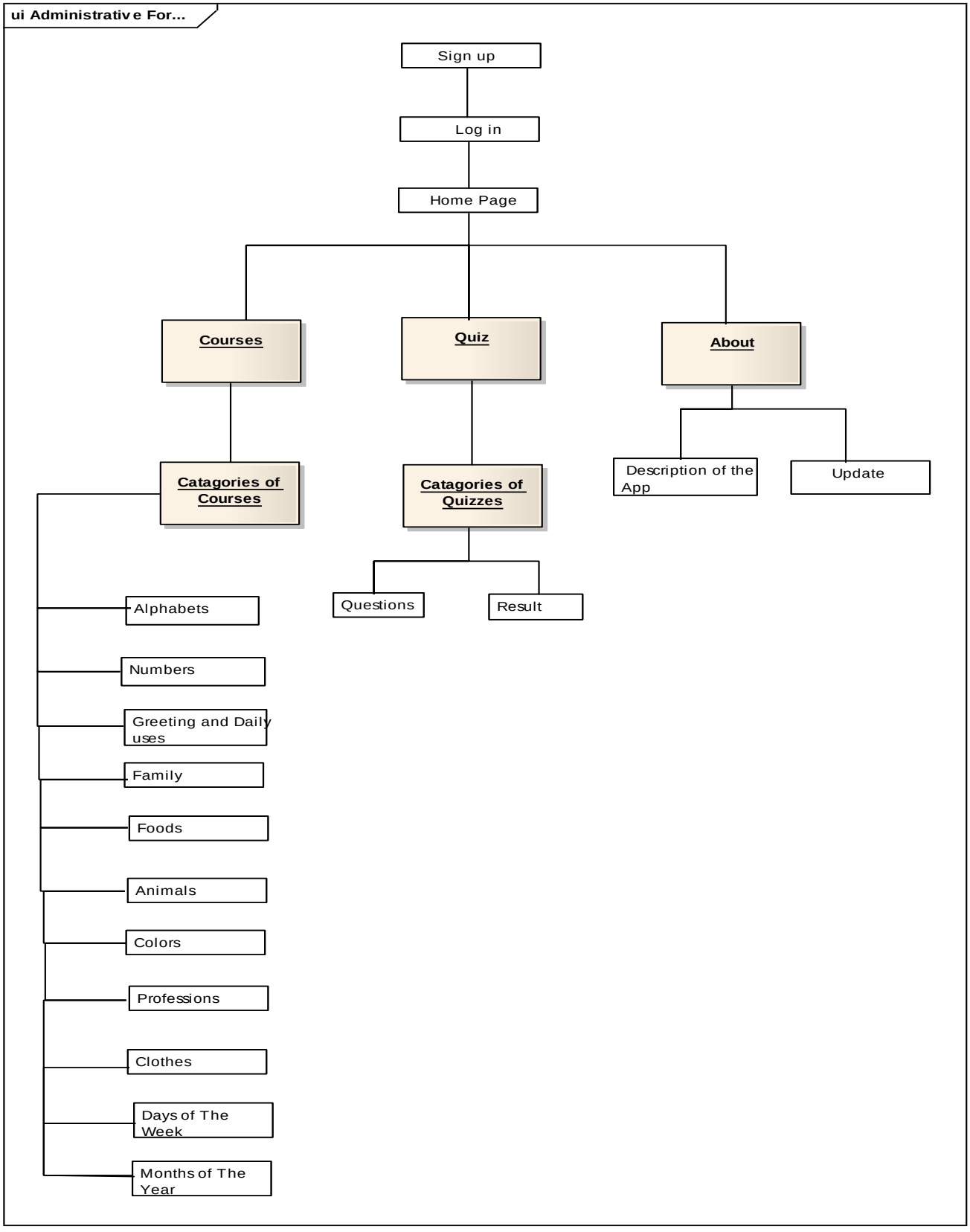

Fig 4.16: User interface

CHAPTER 5

Implementation

5.1 Overview

Implementation is the process of integrating the system functionality. Our project implements the functional and non-functional requirements of the system. Generally, our implementation is the focus on whatever functions of the developed system.

5.2 Tools and technologies utilized during system development

We used different tools and technologies to develop our system. These tools include client side and server-side tools to develop the system. The client-side tools include every tool that the clients use to get a package of courses to learn Geez language or to load information that are stored in the database by the admin. And the server-side tools include the server side that give response for the users' request. The programming instrument we used to develop our system is Flutter and Firebase for database. The client and the server are communicate using the query builder and eloquent we used. Example:

DB: table('table_name')->select('column_name')->get() etc. the DB management system is done by the admin of the system and it is done by controlling the DB system.

5.3 Prototype

In order to use the application, the user must register and logged in to the application system. So, the working principles (block diagrams) of our systems are as follows:

Fig 5.1: block diagram of the system

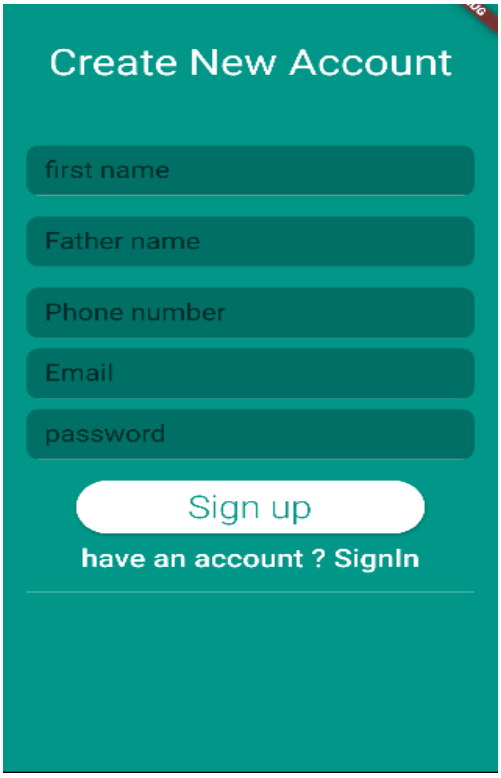

Fig 5.2: Prototype of Signup page

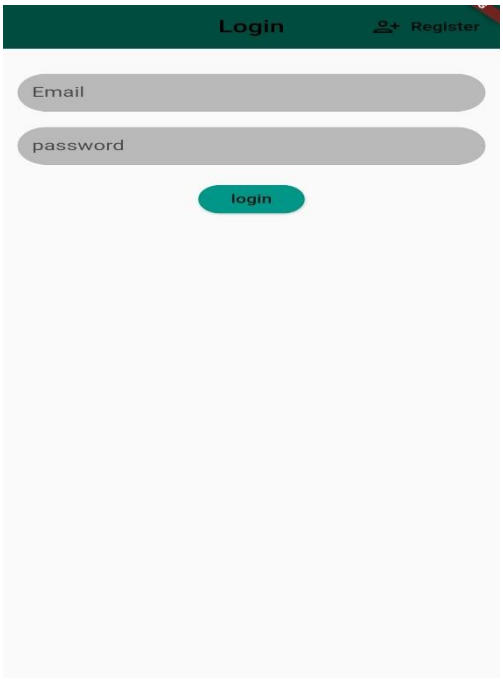

Fig 5.3: Prototype of Login page

Fig 5.4: Prototype of Homepage

5.4 Implementation detail

Component name	Implementation detail
Software	Implemented using Flutter and is responsible for viewing details of Courses, quizzes and dictionaries.
Database	Implemented using Firebase
Profile model	This is a class which is linked with the profile table in the database and responsible for the extraction of record from and to the table.
User view	this is the view class using the blade template

	engine connected to the component controller class and used to create the user view to the user.

Table 5.1: implementation detail of the system