Quiz Mukabla :

Online Quiz Competition Management web-app

A Project Work Report

Dissertation Submitted in partial fulfillment of requirement for the Award of

BACHELOR OF COMPUTER APPLICATION

(3 YEAR)

VI Semester

Session January – June

2023

BY

Yuvraj Pandey (IC-2K20-92)

Prateeti Nigam (IC-2K20-57)

Under the Guidance of

Dr. Ramesh Thakur

International Institute of Professional Studies

Devi Ahilya Vishwavidyalaya

Jan,2023

Report Approval

The project work **“QUIZ MUKABLA”** is hereby approved as a creditable study of an engineering/computer application subject carried out and presented in a manner satisfactory to warrant its acceptance as prerequisite for the Degree for which it has been submitted.

It is to be understood that by this approval the undersigned do not endorse or approved any statement made, opinion expressed, or conclusion drawn there in but approve the “Project Report” only for the purpose for which it has been submitted.

Internal Examiner

Name:

Designation

Affiliation

External Examiner

Name:

Designation

Affiliation

Declaration

We hereby declare that the project entitled **"QUIZ MUKABLA"** submitted in partial fulfillment for the award of the degree of Bachelor of Computer Application (3 year) completed under the supervision of **Dr. Ramesh Thakur,** Faculty of Engineering, International Institute of Professional Studies, DAVV, Indore is an authentic work.

Further, we declare that the content of this Project work, in full or in parts, have neither been taken from any other source nor have been submitted to any other Institute or University for the award of any degree or diploma.

Yuvraj Pandey
Prateeti Nigam

Certificate

I **Dr. Ramesh Thakur** certify that the project entitled **"QUIZ MUKABLA"** submitted in partial fulfillment for the award of the degree of Bachelor of computer application (VI semester) by **Yuvraj Pandey and Prateeti Nigam** is the record carried out by them under my guidance and that the work has not formed the basis of award of any other degree elsewhere.

_____

Dr. Ramesh Thakur

Information Technology

IIPS, DAVV, Indore

_____

Acknowledgements

We are incredibly grateful and externally thankful to our honorable project mentor, “Dr. Ramesh Thakur” for serving as a source of inspiration and for his unwavering assistance in the design, implementation, and development of the project. Without Ramesh sir’s continuous help and support, this project would ever have reached to the completion.

It is their help and support, due to which we became able to complete the design and technical report.

Without their support this report would not have been possible.

Many people and team members itself have made valuable comment suggestion on this proposal which give us an inspiration to improve our project we thank you all the for their direct and indirect help.

With Regards

Yuvraj Pandey
Prateeti Nigam

BCA(3year)
VI semester
IIPS, DAVV, Indore

TABLE OF CONTENTS

CHAPTER TITLE

2. System Requirements Analysis

2.1 Introduction

2.1.1 Purpose

2.1.2 Document conventions

2.1.3 Intended audience and reading suggestions

2.1.4 Product scope

2.1.5 References of SRS

2.2 Overall description

2.2.1 Product perspective

2.2.2 Product functions

2.2.3 User classes and characteristics

2.2.4 Operating environment

2.2.5 Design and implementation constraints

2.2.6 User documentation

2.2.7 Assumptions and dependencies

2.3 External interface requirements

2.3.1User interfaces

2.3.2Hardware interfaces

2.3.3Software interfaces

2.3.4Communication interfaces

2.4 Functional requirements

2.4.1 System feature 1

2.4.2 System feature 2

5. Implementation

5.1 Language and database system used for the implementation

5.2 Features of language and database used for theproject

5.3 Reason for selecting language and database used

5.4 Description of third party tools

6. Testing

6.1 White box testing

6.2 Black box testing

6.3 Interface testing

7. Future scope and limitations

8. Conclusion

9. Bibliography and References

Abstract

The '**Quiz Mukabla'** project is a web-based application to conduct quiz competitions online on the web in an efficient manner. The main objective of this project is to efficiently evaluate a candidate via a fully automated system that not only saves time but also gives fast results. The administrator can conduct, reschedule, terminate, re-conduct quizzes according to their convenience. This software can also be used in educational institutions and the corporate world. Can be used anywhere, anytime as it is a web-based application; the user location doesn't matter. This system allows users to sign upor register and take part in any available quiz competition. Any number of participants can take part in any quiz competition at a time but they must be logged in first.

The administrator doesn't have to be present while the quiz is being conducted. The application shows the participants' score to the participant in real time.Participants will be able to see their standing in the leaderboard after theycomplete the quiz. Participants can also see quizzes they participated in the pastand their scores/standings in the same. This software provides a competitiveplatform, where a student not only judges their knowledge/skill but also they can improve the same and earn cash at the same time.

List of tables

.

List of Figures

Chapter 1
INTRODUCTION

1. Introduction

1.1. Literature Review:

What is Java?

Java is a **programming language** and a **platform**. Java is a high level, robust, object-oriented and secure programming language.

Java was developed by *Sun Microsystems* (which is now the subsidiary of Oracle) in the year 1995. *James Gosling* is known as the father of Java. Before Java, its name was *Oak*. Since Oak was already a registered company, so James Gosling and his team changed the name from Oak to Java.

Java is an object-oriented, class-based, concurrent, secured and general-purpose computer-programming language. It is a widely used robust technology.

JVM

JVM (Java Virtual Machine) is an abstract machine. It is called a virtual machine because it doesn't physically exist. It is a specification that provides a runtime environment in which Java bytecode can be executed. It can also run those programs which are written in other languages and compiled to Java bytecode.

JVMs are available for many hardware and software platforms. JVM, JRE, and JDK are platform dependent because the configuration of each OS is different from each other. However, Java is platform independent. There are three notions of the JVM: *specification*, *implementation*, and *instance*.

JDK

JDK is an acronym for Java Development Kit. The Java Development Kit (JDK) is a software development environment which is used to develop Java applications and applets. It physically exists. It contains JRE + development tools.

JDK is an implementation of any one of the below given Java Platforms released by Oracle Corporation:

- Standard Edition Java Platform
- Enterprise Edition Java Platform
- Micro Edition Java Platform

The JDK contains a private Java Virtual Machine (JVM) and a few other resources such as an interpreter/loader (java), a compiler (javac), an archiver (jar), a documentation generator (Javadoc), etc. to complete the development of a Java Application.

Introduction of Java Swing

Swing has about four times the number of User Interface [UI] components as AWT and is part of the standard Java distribution. By today's application GUI requirements, AWT is a limited implementation, not quite capable of providing the components required for developing complex GUI's required in modern commercial applications. The AWT component set has quite a few bugs and really does take up a lot of system resources when compared to equivalent Swing resources. Netscape introduced its Internet Foundation Classes [IFC] library for use with Java. Its Classes became very popular with programmers creating GUI's for commercial applications.

- Swing is a Set Of API (API- Set Of Classes and Interfaces)
- Swing is Provided to Design Graphical User Interfaces
- Swing is an Extension library to the AWT (Abstract Window Toolkit)
- Includes New and improved Components that have been enhancing the looks and Functionality of GUIs'
- Swing can be used to build(Develop) The Standalone swing GUI Apps Also as Servlets And Applets

1.2. Purpose:

The purpose of this document is to present a detailed description of the "**Quiz Mukabla**" software system. It will explain the purpose and features of the system,the interface of the system, what the system will do, the constraints under which itmust operate and how the system will react to external stimuli. This document is intended for both the stakeholders and the developers of the system.

1.3. Scope:

This document can be read and/or edited only by the "requirements gathering team", "architect team" or "design team" of "Quiz Mukabla" software project but any other team involved in this project must first propose any changes they want tomake to the above mentioned teams.

1.4. Problem in existing system:

The various problems in the existing systems are listed below:

a. Currently, there is no good software system in the market which provides an easy and seamless experience of organizing quiz competitions in a centralized manner.
b. Quiz participants face a lot of issues like the results are declared very late and are not correct sometimes.
c. They have to retake the entire quiz if they face any problem in their internet connection.
d. There is no good centralized way to manage quiz competitions.
e. There are many problems faced during a quiz competition.

1.5. Statement of Problem:

"Quiz Mukabla" software system will provide a centralized platform for conduction of quiz competitions to both quiz organizers and quiz participants. Newparticipants can register while the existing ones will be able to login to the "Quiz Mukabla" software system. Quiz participants will be pre-informed about the rules and regulations of the quiz every time they participate in a quiz competition. Quiz participants will be able to answer one question at a

time with four options provided based upon the results of which the participants' scores will be affected. As the quiz ends, the final score will be displayed to the participants' screen.

CHAPTER 2
SYSTEM REQUIREMENTS ANALYSIS

2. System requirements analysis:

2.3. Introduction:

2.3.1. Purpose:

The purpose of the "Quiz Mukabla" software project is to provide a centralized platform for conduction of quiz competitions to the quiz organizers. To provide a seamless experience of participation to the quiz participants**.**

2.1.2. Documentation Conventions:

S. No.	Label	Font size	Font style	Bold	Italic
1.	Heading	14	Times New Roman	Yes	No
2.	Subheading	12	Times New Roman	Yes	No
3.	Sub Subheading	11	Times New Roman	Yes	No
4.	General Text	11	Times New Roman	No	No

2.1.3 Intended Audience and Reading Suggestions:

The intended audience for this documentation are stakeholders, software requirements analysis team, software architecture team, software testing team, and other teams involved in the engineering of this project.

2.1.4 Product Scope:

This software system will be a quiz competition management system for the quiz organizers. To provide a system where there is no scope of errors and the results are much faster. This system will be designed to save the time taken while conducting quiz competitions and assist in automation of quiz competitions. By providing a common platform to the quiz participants and quiz organizers the system will ease the process of conduction of quiz competitions.

More specifically, this system is designed to allow quiz organizers to conduct quiz competitions by providing them a centralized platform. The system is based on a relational database for its "conduct quiz competition" funcion. Above all, we hope to provide a comfortable user experience.

2.1.5 References of SRS

Roger S. Pressman, and Bruce R. Maxim, *Software Engineering: A practitioner'sapproach, Eighth Edition*. New York, US: McGraw Hill Education, 2015.

2.2 Overall Description:

2.2.1. Product Perspective:

To ease the experience of "practice for competitive examination" by providing a platform to prepare for the same.

2.2.2. Product Functions:

The functions are divided according to the user types such as:

➔ Administrator (admin):

- Registration (on first usage)
- Login
- Add Competition
- Update Competition
- Delete Competition
- Logout

➔ User (participant):

- Registration
- Login
- Join Competition
- Give Competition
- View score
- View leaderboard
- Logout

2.2.3. User classes and characteristics:

The users of "Quiz Mukabla" web-app are listed below:

➔ Administrator (admin):

The administrator is the owner of the web-app. Only the administrator can schedule quiz competitions. The administrator must have a basic understanding of "systems administration" to efficiently utilize this web-app. The functionalities available to an administrator are:

- Add quiz competition
- Update quiz competition
- Delete quiz competition
- Monitor participants

➔ Participant:

The participants are those who will take the quiz competitions that are scheduled by the administrator. The participant must have a basic understanding of using web browsers to participate. The functionalities available to the participants are:

- Join quiz competition
- Give quiz competition

2.2.4. Operating environment:

The "Quiz Mukabla" web-app can be operated on any Linux-kernel-based operating system.

2.2.5. Design and implementation constraints:

The following constraints ought to be satisfied when implementing "Quiz Mukabla" web-app:

➔ At least 50 participants should be able to participate in a quiz competition at a time.

➔ Admin should be able to manage quiz competitions via a dashboard.

2.2.6. User documentation:

Null

2.2.7. Assumptions and dependencies:

It is assumed based on the given below user classes that:

➔ Administrator:

- The administrator has installed "Quiz Mukabla" on a client/server based system, i.e., on a server with a Linux-kernel-based operating system.
- The administrator is using a web-browser for interacting with "Quiz Mukabla" web-app's user interface.

➔ Participant:

- The participant is using a web-browser for interacting with "Quiz Mukabla" web-app's user interface.

2.3. External Interface Requirements:

2.3.1. User Interfaces:

Registration Screen:
Various fields available on this screen will be:
-Name
-Email Id
-Password

Login Screen:
Fields available on this screen are:
-Email Id
-Password

OTP Screen:
Fields available on this screen:
-User Otp

Display Quiz Competitions Screen:
Fields available on this screen are:
-Join Competition
-Start Competition

Leaderboard Screen:
Fields available on this screen are:
-Participate Name
-Score
-Rank

User Profile Screen:
Fields available on this screen are:
-Set Image-Edit Name

2.3.2. Hardware Interface(s):

A minimum 512Mb RAM is required for "Quiz Mukabla" web-app to function.

2.3.3. Software Interface(s):

The “Quiz Mukabla” web-app requires the following softwares to function:

- ➔ Platform: *Java Runtime Environment (OpenJDK 16)*
- ➔ Web server: *Apache Tomcat 9*

2.3.4. Communications Interface(s):

“Quiz Mukabla” web-app is based on the Client/Server communication system therefore it requires an active internet connection of function.

2.4. Functional Requirements:

This section gives the list of Functional requirements which are applicable to the Online Quiz System. Functional requirements are nothing but the services provided by the system to its end users.

There are two sub modules in this phase.

- User/Participant module.
- Administrator module.

User/Participant module:

2.4.1. New user registration

This page provides registration for new users. Students have to fill a form with Name, Email ID and Password.after successful OTP verification which sends on user’s Email ID users have registered on this web application**.**

2.4.2. Login with Email and Password verification

User login with Email and Password which provide at time of registration .If Email and Password are correct user login in Application and can see home page.

2.4.3. Register in upcoming competition

Users can register with a click on button in upcoming competitions of their choice which show on the homepage.

2.4.4. Give competition

Users can attempt a competition at the time duration of the competition which is registered by the user. If a user is not registered for a competition then the user can not attempt the competition.

2.4.5. View Score of competition

Users can view their score after finishing the competition and after that.

2.4.6. View leaderboard of competition

Users can view the leaderboard of a particular competition which was attempted by the user.

2.4.7. Logout user

Users can Log Out from the application with a click on button.

Administrator module:

2.4.8. Login Administrator

Admin login with Email and Password which is stored in the database.

2.4.9. Add competition

Admin can add new competition of particular topic with entry fees, timing of competition and maximum number of user

2.4.10. Update competition

Admin can update the competition information like entry fees and timing of competition and also can cancel or delete competition.

2.4.11. Logout Administrator

Admin can logout from the application with clicking on the logout button.

2.5. Non-functional requirements:

2.5.1. Performance Requirements:

1) **Response Time**-The system shall give responses in 2 sec after user login.

2) **Capacity**-The system can support multiple computers but it needs to be installed on every computer separately.

2.5.2. Safety Requirements:

All logged information, updates, user activities are securely stored.

2.5.3. Security Requirements:

Any modification for the Database shall be synchronized and done by system admin.

3. Analysis

3.1. Methodology Used:

The Programming language used for the development of this project is JAVA.

3.2. Use Case diagram

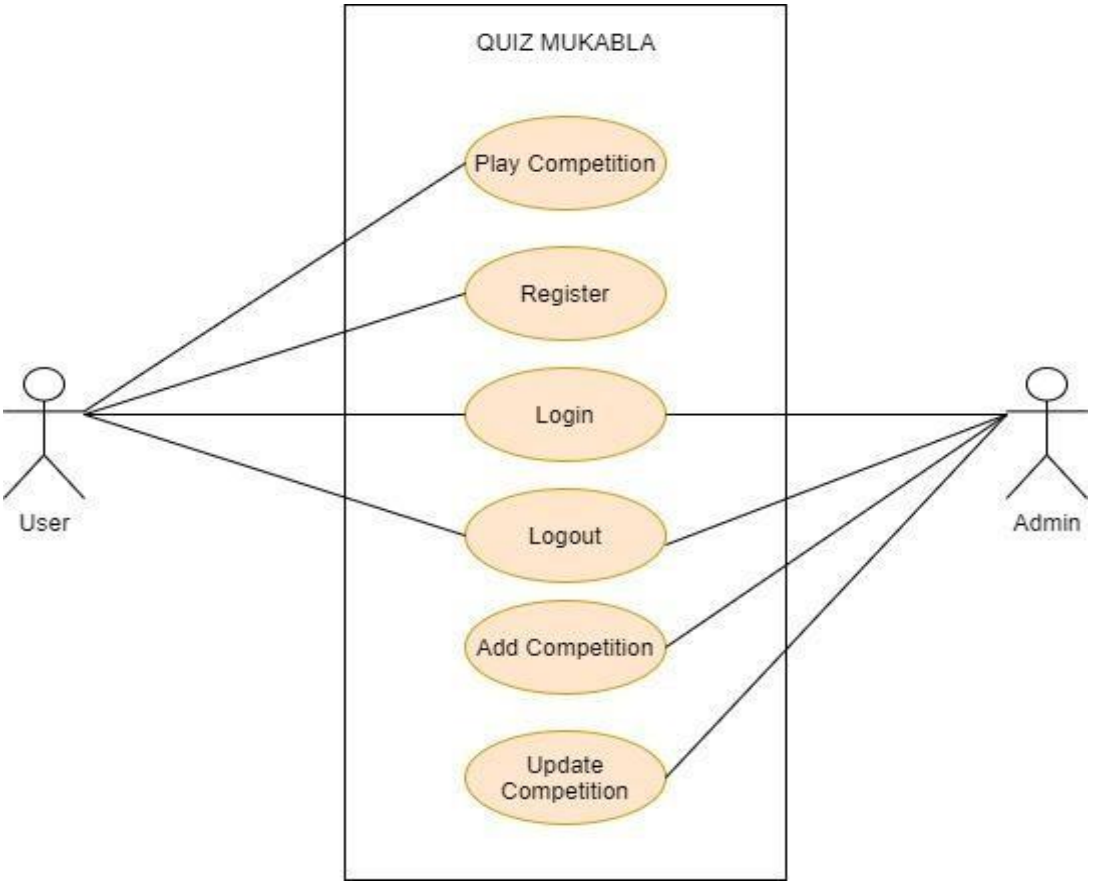


Fig. 3.1 Use Case of Quiz Mukabla app

I. Use Case Specification

Name of the use case	Registration
Actor	User/Student
Precondition	None
Primary flow of events	Name, Email Id, Password
Alternate flow of events	If email id or password is wrong then show message “incorrect details”
Post condition	If successful, register then come to the login page.
Use case termination	Cancel

Table 3.1 Use Case Specification for Registration

Name of the use case	Log-In
Actor	User/Student
Precondition	None
Primary flow of events	Email Id, Password
Alternate flow of events	If email id or password is wrong then show message “incorrect email id or password”
Post condition	If successful log then come to the home page.
Use case termination	Cancel

Table 3.2 Use Case Specification for Log-In

Name of the use case	Home page
Actor	User/Student
Precondition	Login Successfully.
Primary flow of events	Can join any of the given Competition
Alternate flow of events	If your Coins is less than entry fees of Competition then show message “You have less Coins”
Post condition	Will Start Competition
Use case termination	Cancel

Table 3.3 Use Case Specification for Home Page

Name of the use case	Home page
Actor	Admin
Precondition	Login Successfully.
Primary flow of events	Can Add/Update Competition and Add Questions.
Alternate flow of events	None.
Post condition	None.
Use case termination	Cancel

Table 3.4 Use Case Specification for Admin Home Page

Name of the use case	Leaderboard Page
Actor	User/Student
Precondition	Submit Quiz
Primary flow of events	Can see Leaderboard
Alternate flow of events	None.
Post condition	Can give feedback and go back to the Home Page.
Use case termination	Cancel

Table 3.5 Use Case Specification for Leaderboard

Name of the use case	Logout
Actor	User/Students
Precondition	Successful login.
Primary flow of events	None.
Alternate flow of events	Session expired.
Post condition	Users will be redirected to the login page.
Use case termination	Cancel

Table 3.6 Use Case Specification for Logout

3.3. Activity Diagram

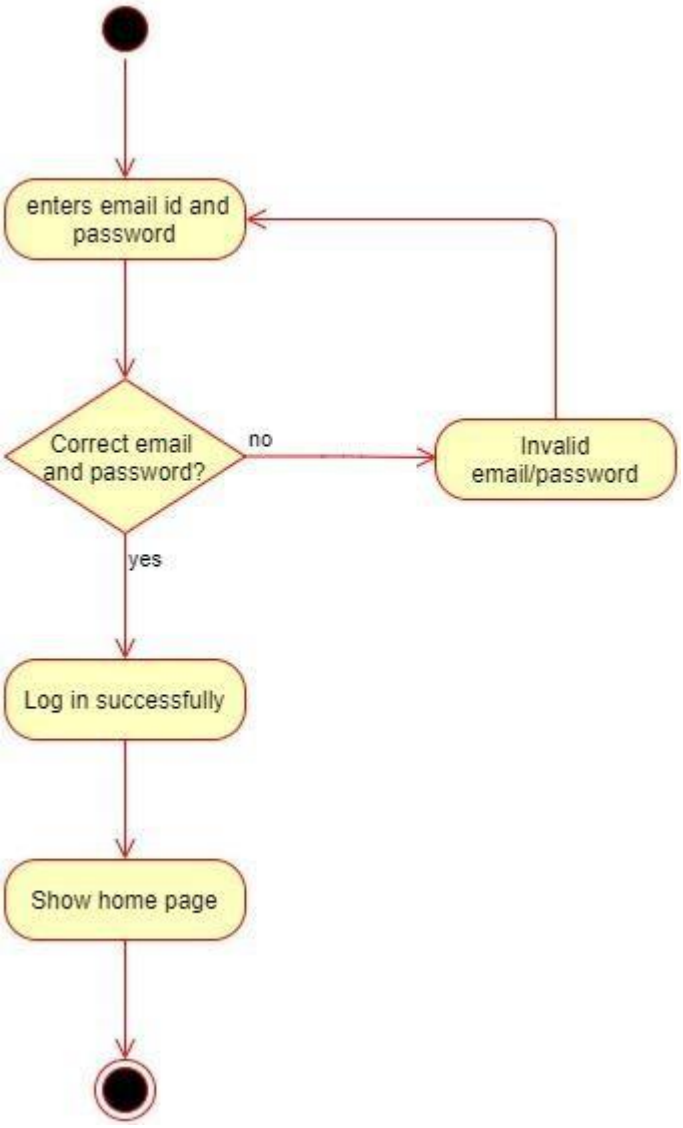


Fig. 3.2 Activity diagram for Login

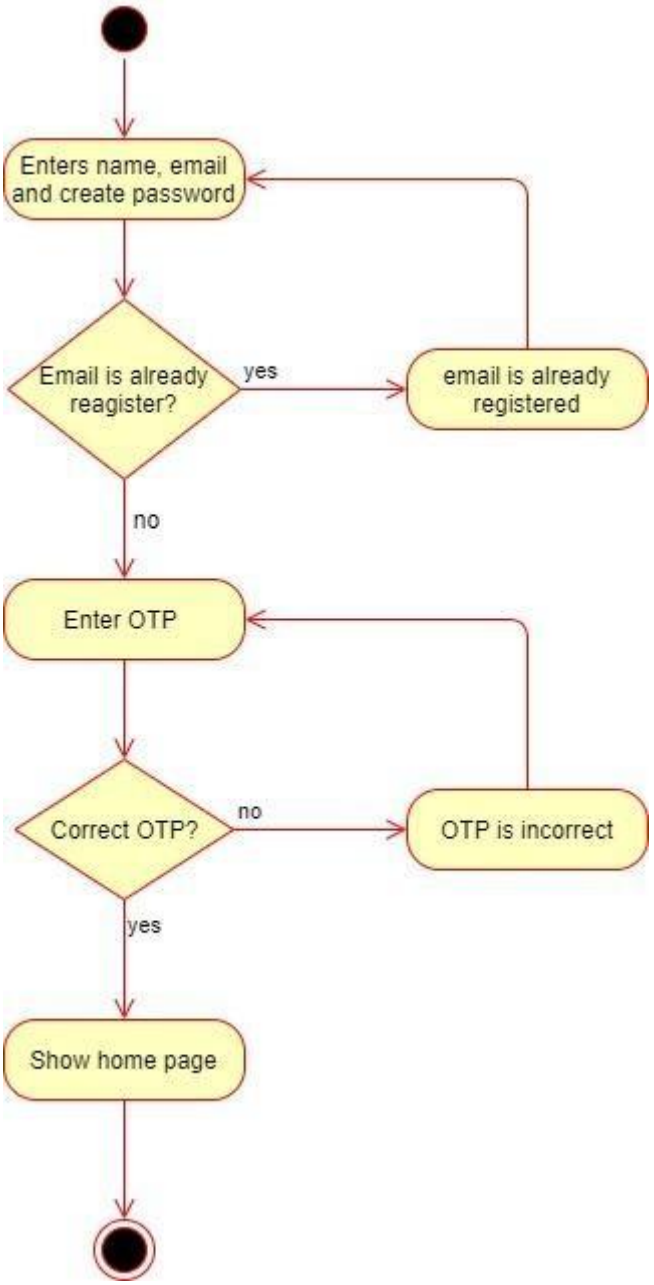


Fig.3.3 Activity Diagram for Registration

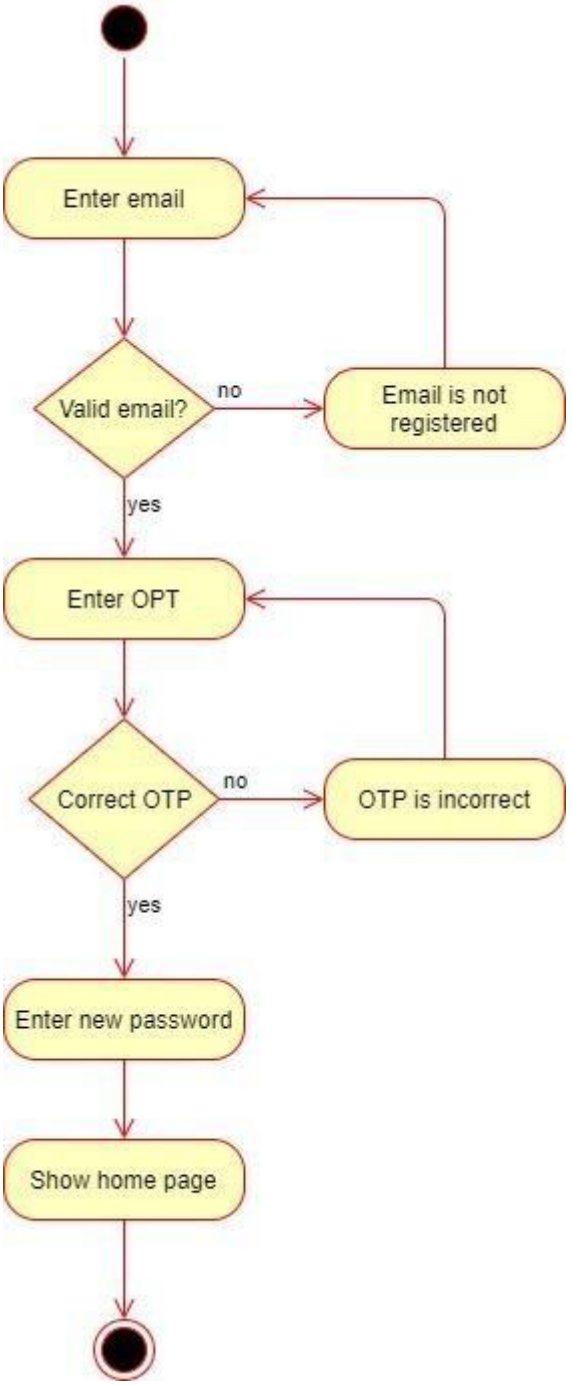


Fig. 3.4 Activity Diagram for Forget Password

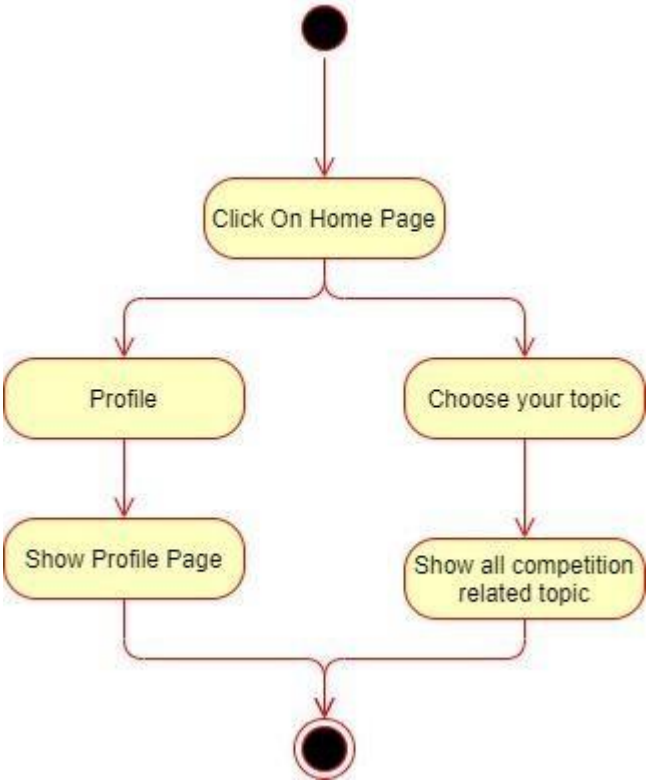


Fig. 3.5 Activity Diagram for Home Page

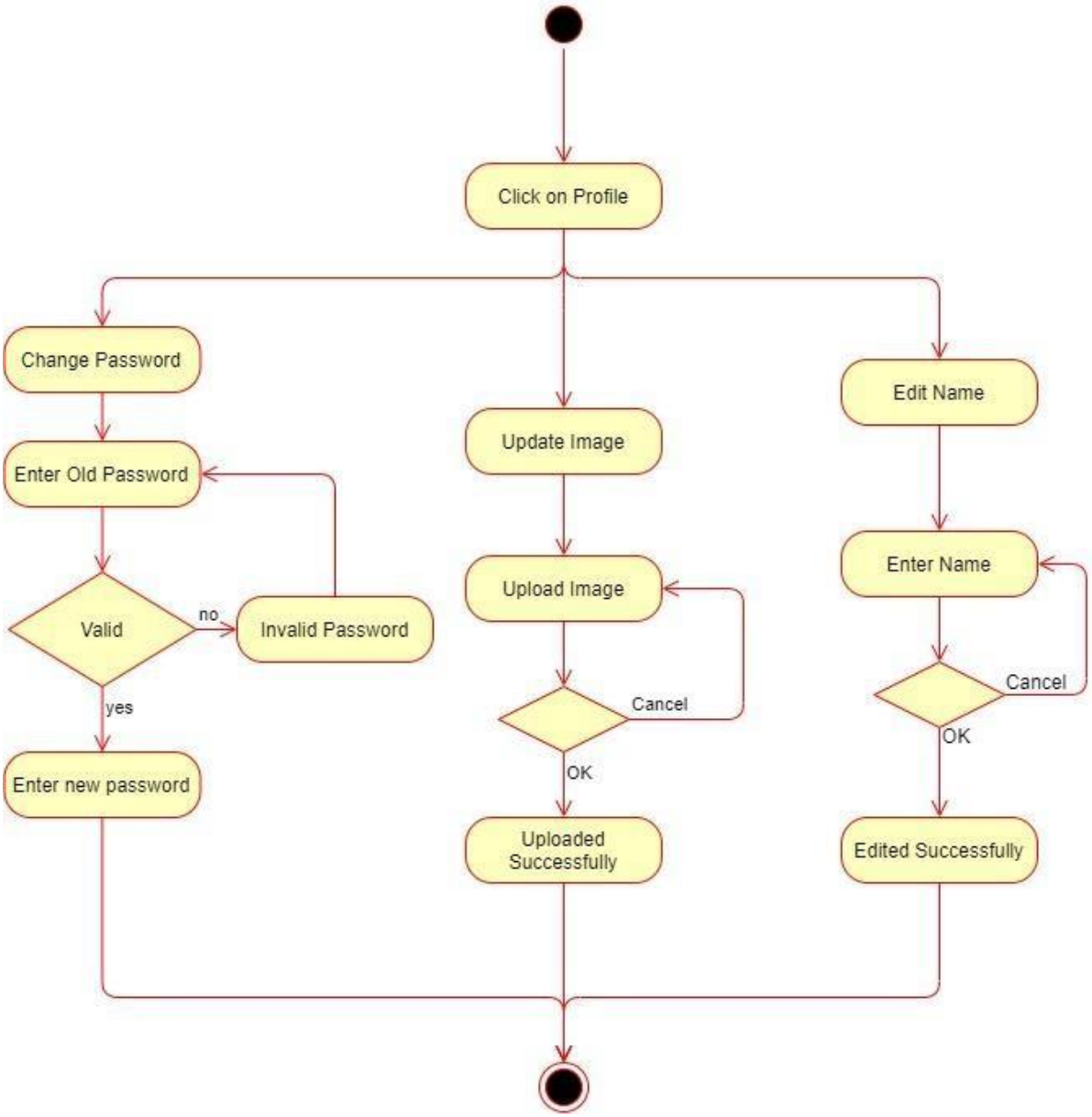


Fig.3.6 Activity Diagram for Profile

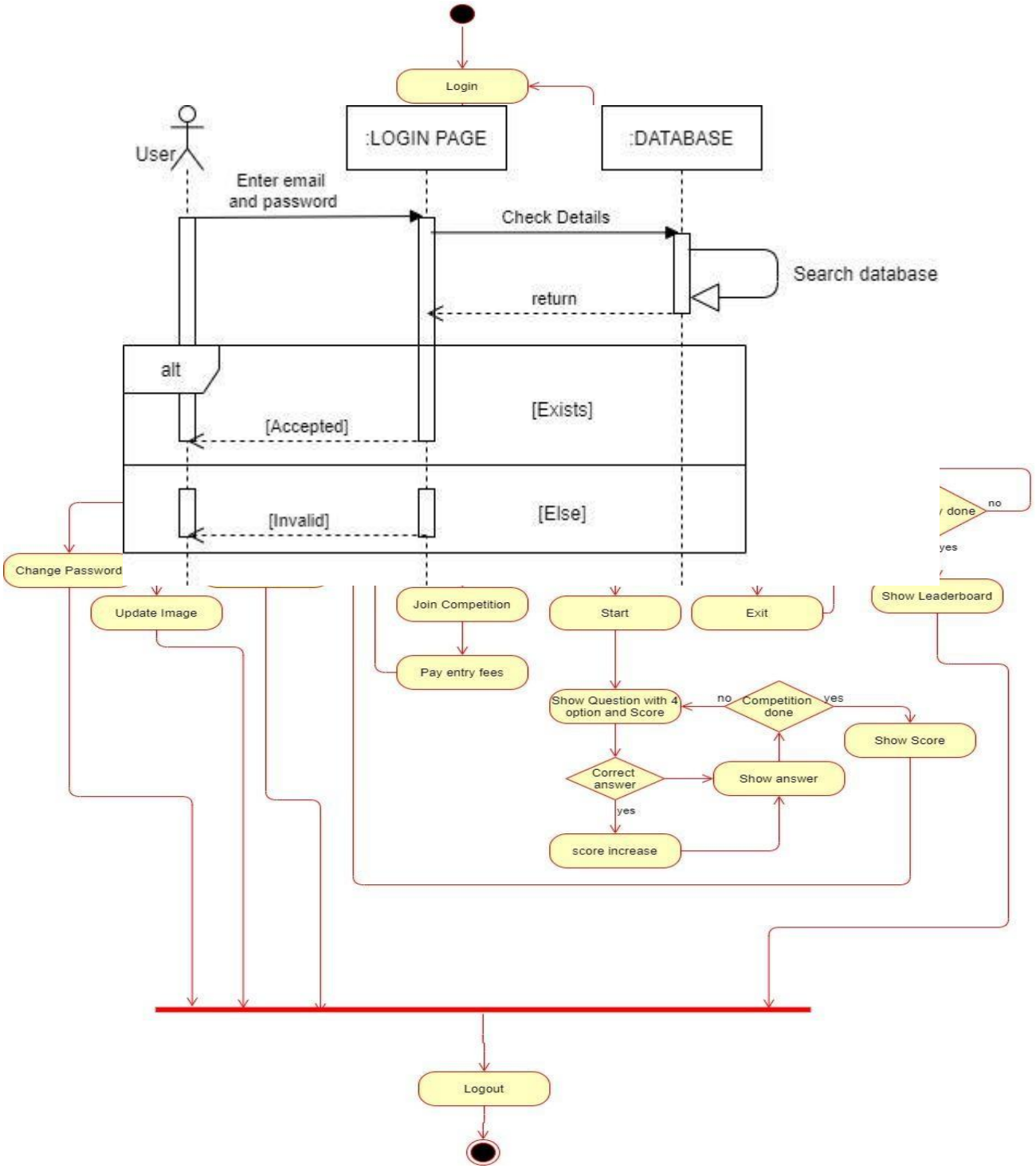


Fig.3.7 Activity Diagram for Actor

3.4. Sequence Diagram

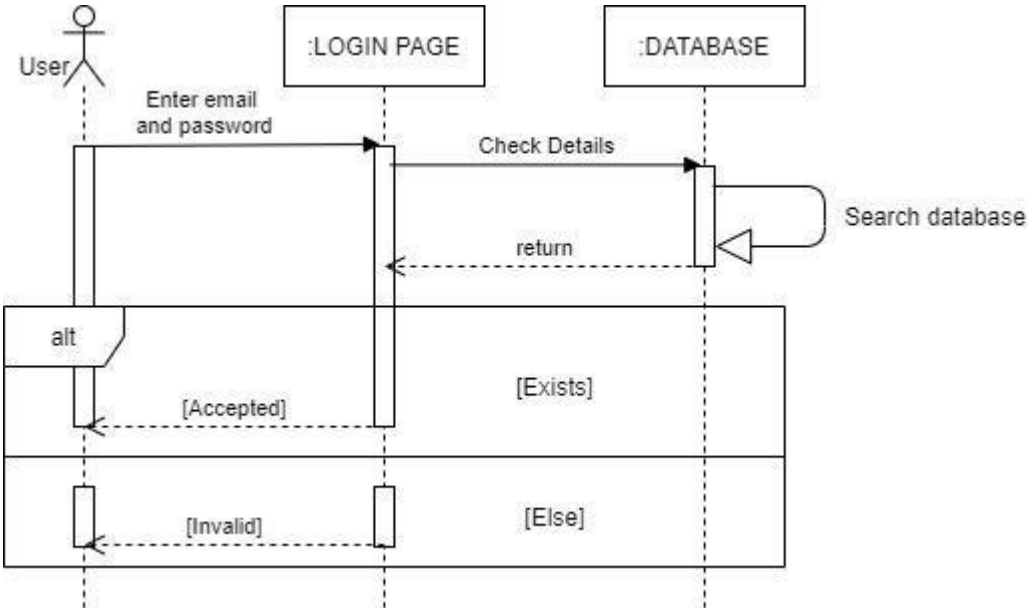


Fig: 3.8 Sequence Diagram For Login

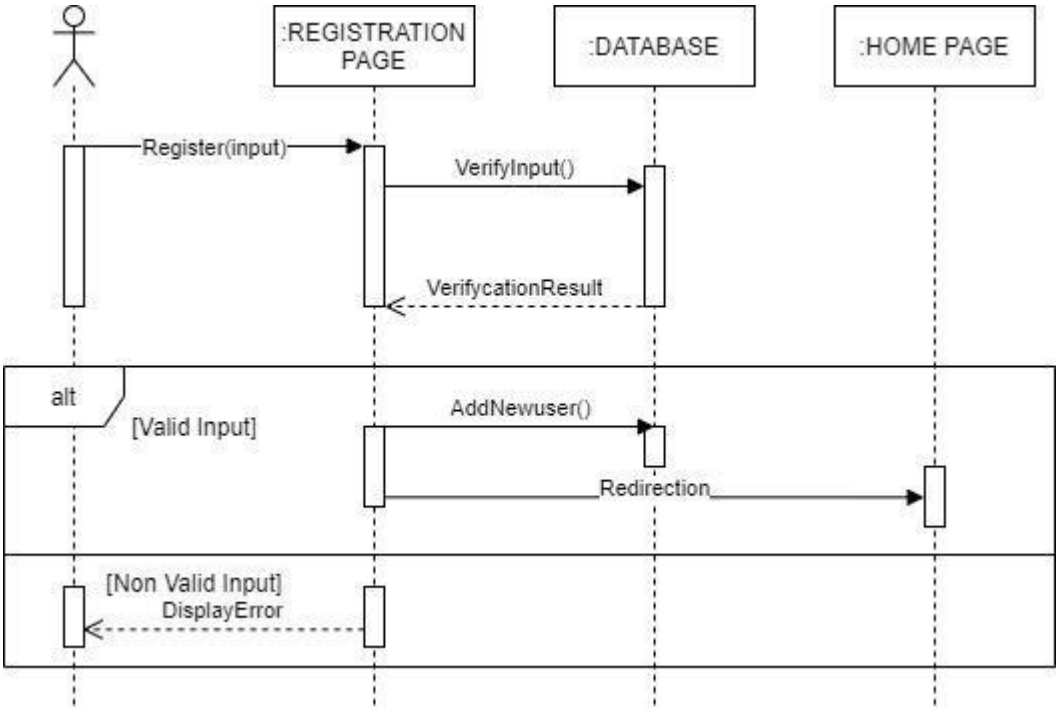


Fig: 3.9 Sequence Diagram For Registration

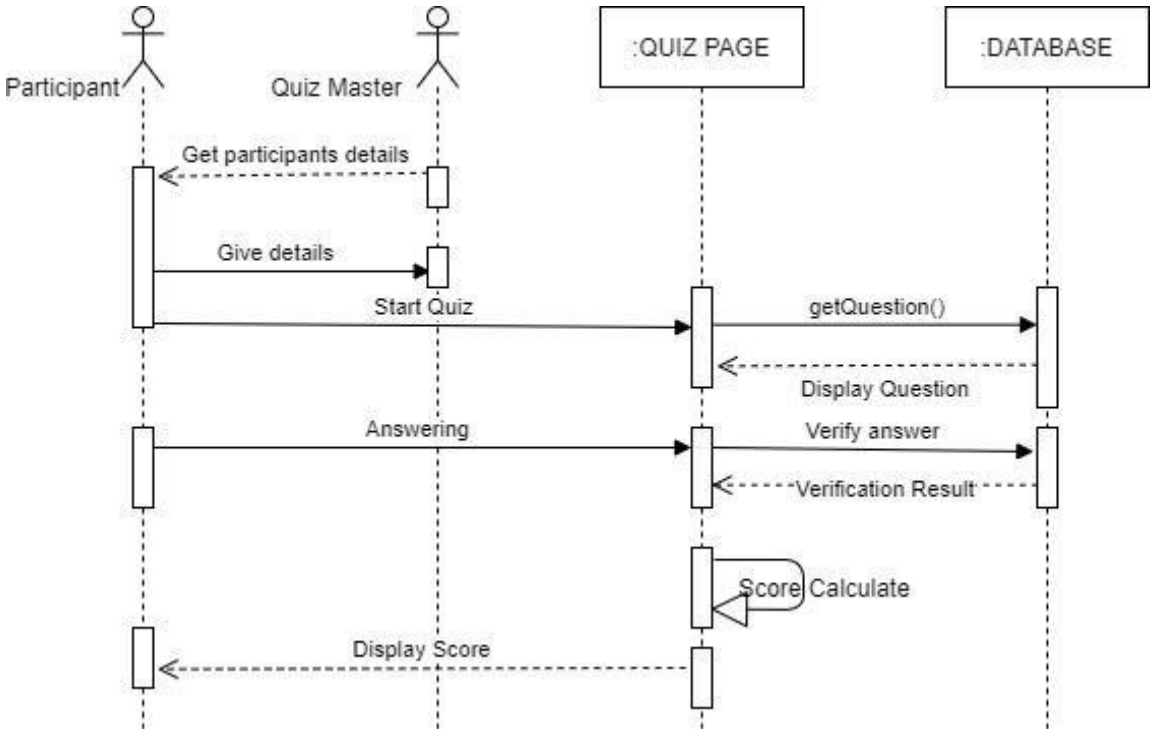


Fig: 3.10 Sequence Diagram For Quiz Play

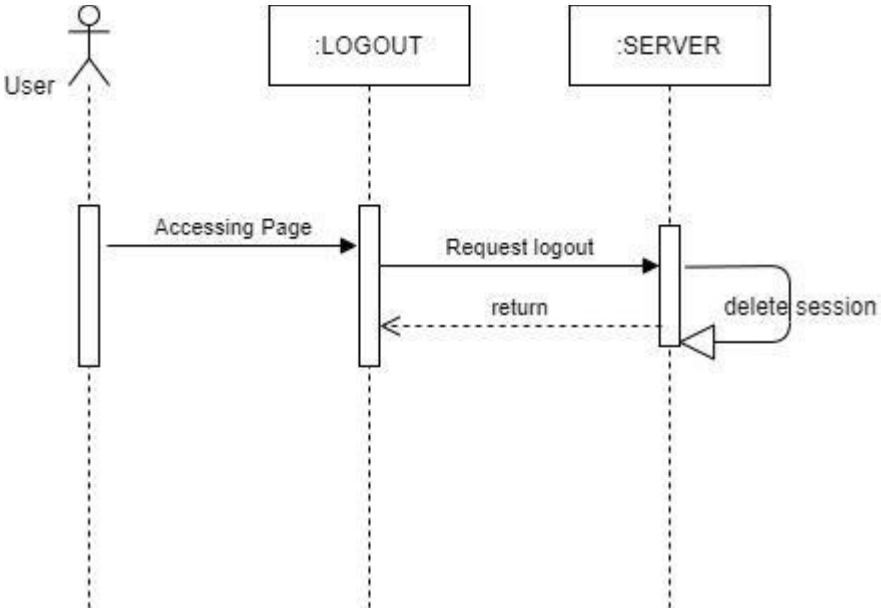


Fig: 3.11 Sequence Diagram For Logout

3.5. Class Diagram

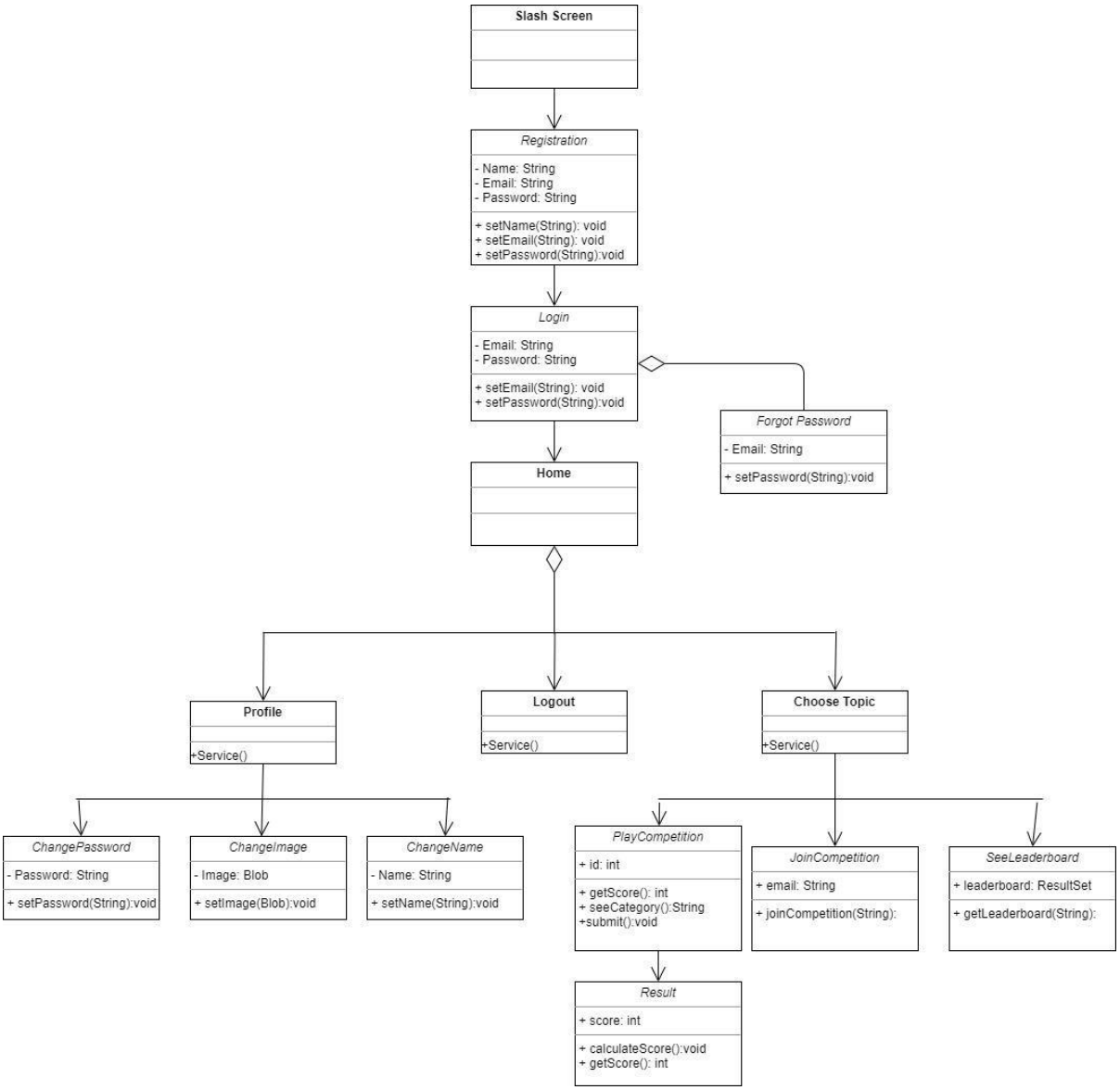


Fig: 3.12 Class diagram of Quiz app

4. DESIGN:

4.1. Architectural Design:

4.1.1. System architecture design

Three-tier architecture is a client/server software architecture pattern in which the use interface (presentation), functional process logic ("business rules"), computer data storage and data access are developed and maintained as independent modules most often on separate platform.

Apart from the usual advantages of modular software with well-define interface the Three- tier architecture is intended to allow any of the three — tier to be upgraded or replace independently in response to changes in requirements or technology . For example, a change of operating system in the presentation tier would only affect the use of interface code.

Typically, the user interface runs on a desktop pc or workstations and uses a standard graphical user interface functional process logic that may consist of one or more separate modules running on workstations on application server, and RDBMS on a database server or mainframe that contains the computer data storage logic. The middle tier may be multi-tiered itself.

4.2. Database Design:

4.2.1. Normalization

Database normalization is a technique of organizing the data in the database. Normalization is a systematic approach to decomposing tables to eliminate data redundancy and undesirable characteristics like insertion, update and deletions Anomalies. This is a multi state process that puts data into tabular form by removing duplicated data from the relation tables. Normalization is used for mainly two purposes:

• Eliminating redundant data.

• Ensuring data dependence makes sense i.e. data is logically stored.

Our table in the database is in 1 NF form. A table is said to be in 1 NF if both the following conditions hold:

A relation will be 1 NF if it contains an atomic value. It states that attributes of a table cannot hold multiple values. It must hold single valued attributes. 1 NF disallows the multi-valued attribute, composite attribute, and their combinations.

4.3. Interface Design

4.3.1. Screenshots

Fig: 4.1 Login GUI

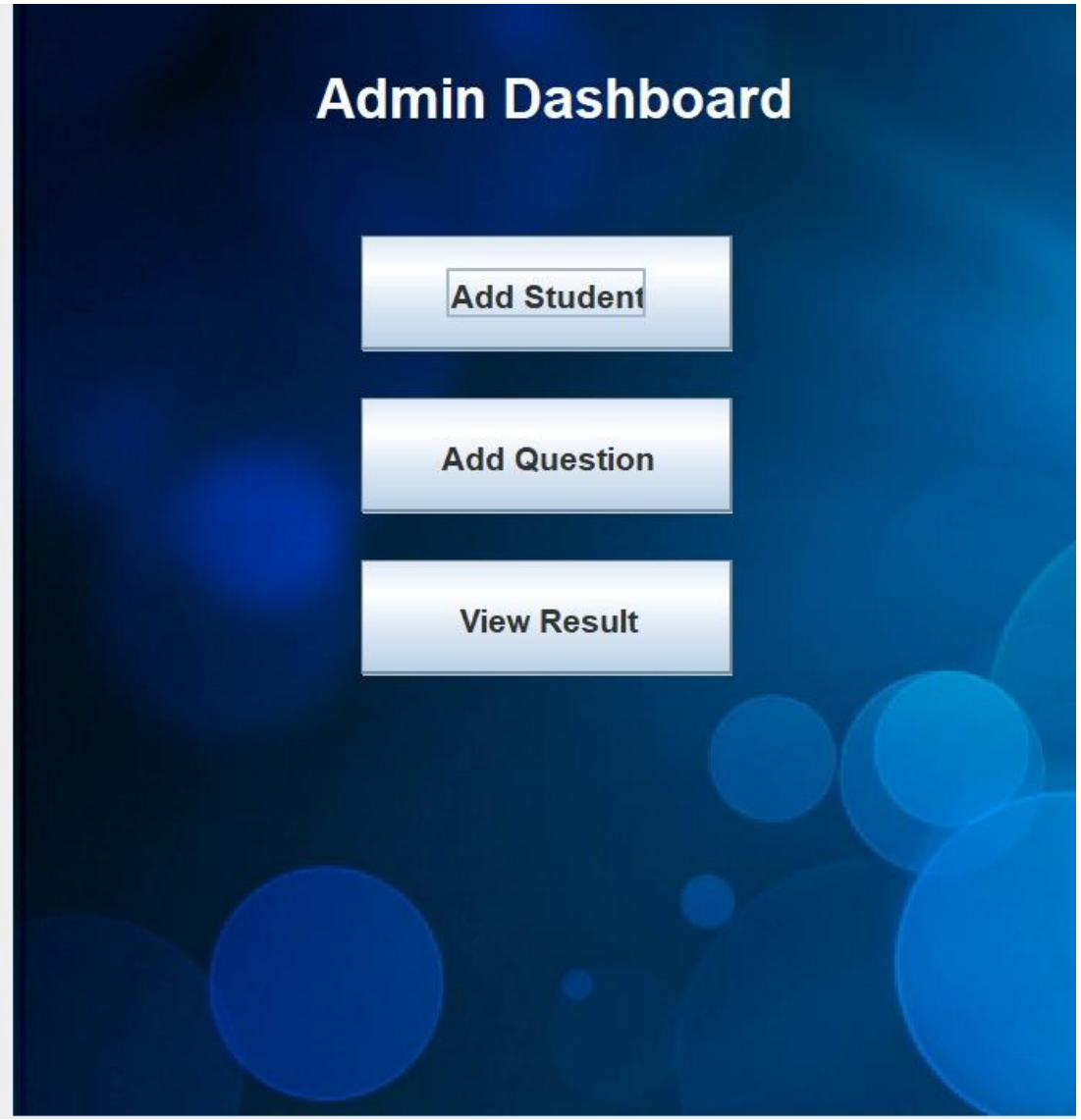


Fig: 4.2 Admin Dashboard GUI

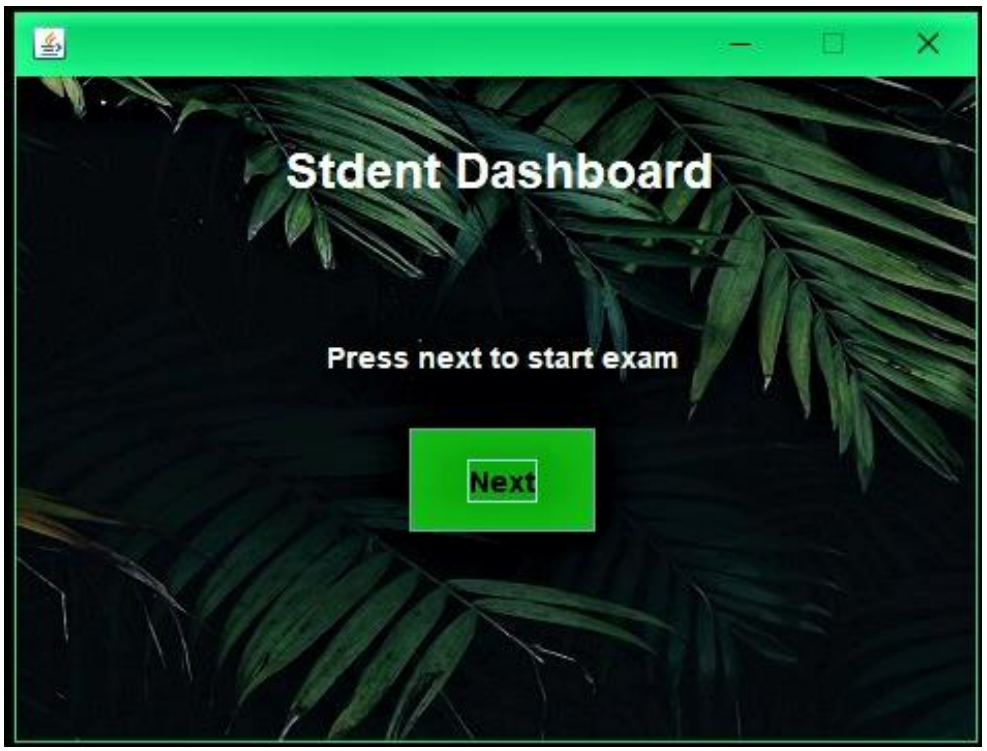
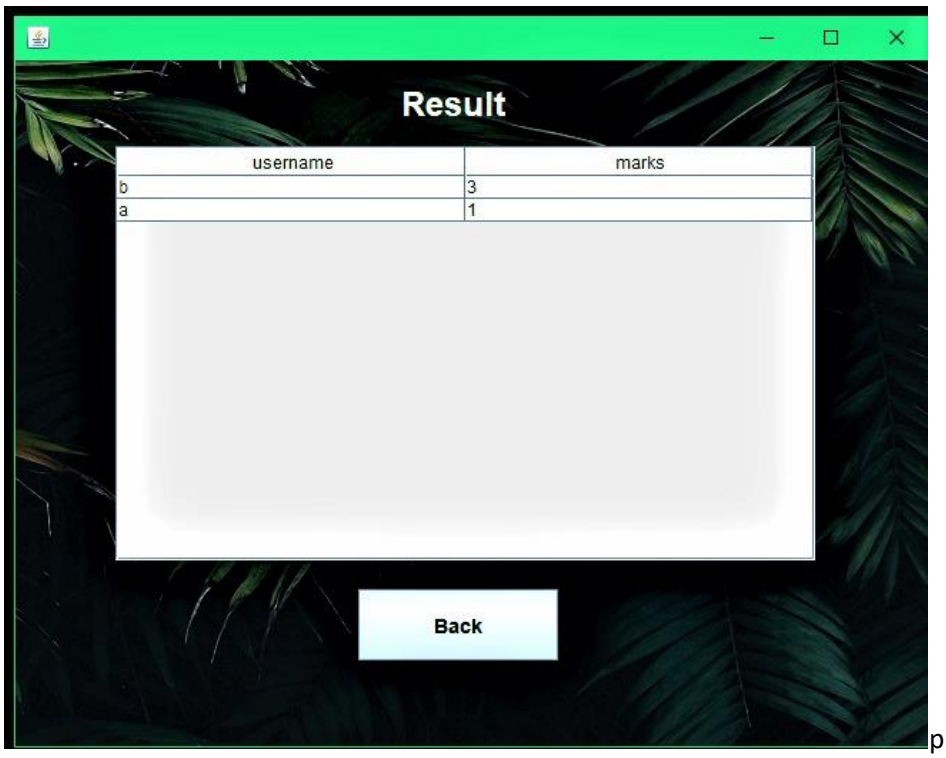


Fig: 4.3 Student Dashboard GUI



pb

Fig: 4.4 Result GUI

5. Implementation:

5.1. Language and database system used for the implementation

- For the frontend development of the a fore mentioned project:
  HTML 5,CSS , JavaScript are used.

- For the backend development of the a fore mentioned project:

• Java i.e., the Swing is used.

5.2. Features of language and database used for the project

• **HTML:** Widely used for creating web page & pioneer in web application development

• Single & integrated environment, Analysis Services, ReportingServices Supports, Administrative Tasks.

5.3. Description of third party tools used

• An IDE, called Netbeans using Apache Tomcat server.

• For the designing phase, UML Diagram maker from Draw. io.
• For CSS , bootstrap.

6. Future scope and limitations:

Scope:-

The Scope of this project is very broad in terms of gaining knowledge and sharing knowledge among the world.

Few points are:-

• Can be used anywhere any time as it is a web based application.

• This application will be used in educational institutions as well as in the corporate world.

Limitation:-

• We want to implement a concept where the user itself can add questions for others.

• We want to implement a concept where the user can solve any query related to each question at that time only.

• We also want to implement a concept where every user can see the profile of another user.

7. Conclusion:

The ‘Quiz Mukabla’ project is a web-based application to conduct quiz competitions online on the web in an efficient manner. The main objective of this project is to efficiently evaluate a candidate via a fully automated system that not only saves time but also gives fast results. The administrator can conduct, reschedule, terminate, re-conduct quizzes according to their convenience. This software can also be used in educational institutions and the corporate world. Can be used anywhere, anytime as it is a web-based application; the user location doesn’t matter. This system allows users to sign up or register and take part in any available quiz competition. Any number of participants can take part in any quiz competition at a time but they must be logged in first.

The administrator doesn’t have to be present while the quiz is being conducted.

The application shows the participants’ score to the participant in real time. Participants will be able to see their standing in the leaderboard after they completethe quiz. Participants can also see quizzes they participated in the past and their scores/standings in the same. This software provides a competitive platform, where a student not only judges their knowledge/skill but also they can improve the same and earn cash at the same time.

8. References:

- **https://www.W3schools.com/**
- **https://stackoverflow.com/**
- **https://www.draw.io/**

- Roger S. Pressman, and Bruce R. Maxim, *Software Engineering: A practitioner's approach, Eighth Edition*. New York, US: McGraw Hill Education, 2015

- Sierra, Kathy and Bert Bates, 2003, Head First Java, O'Reilly.

- https://docs.oracle.com/javase/tutorial