A

MINOR PROJECT REPORT

ON

Quiz Desktop Application

IN

JAVA

Submitted in partial fulfilment of the requirement

For the award of the degree of

Bachelor Of Computer Application(2019-2022)

Submitted To : Submitted By :

B.K.N.M.U Bhavin Bhesaniya

Mayur Solanki

Under The Supervision Of

Mr. Sandeep Upadhyay Sir

**Acknowledgement**

I have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. I would like to extend my sincere thanks to all of them.

I would like to express my sincere thanks to Mr. Sandeep Upadhyay Sir, for his/her valuable guidance and support in completing my project. I would like to express my special gratitude and thanks to industry persons for giving me such attention and time.

My thanks and appreciations also go to my colleague in developing the project and people who have willingly helped me out with their abilities.

We are indebted to all the people of our collage, S.S.S.D.I.I.T, concerned, who have actively involved in marking of the project. Last but not the least we would thanks the god who has made us to do the project & also our parents without their support we would not be able to complete the project successfully

Place : Junagadh

Date : 12/10/21

Index

|  |  |  |
| --- | --- | --- |
| No. | Title | Page No. |
| 1. | Project Profile  - Introduction  - Objective  - Profile Details | 4 |
| 2. | Requirement Gathering  - System Analyse  - Identification Of Need  - Preliminary Investigation  - Questionary  - Project Size Estimation | 6 |
| 3. | Feasibility Study  - Technical Feasibility  - Economical Feasibility  - Operational Feasibility | 11 |
| 4. | Technology Used  - Tools / Environment Used  - Language and Software Description | 13 |
| 5. | Data-Flow Diagram  - 0-Level Diagram  - 1-Level Diagram | 20 |
| 6. | Gantt Chart | 23 |
| 7. | Pert Chart | 24 |
| 8. | Waterfall Model | 25 |
| 9. | Project Design  - User Interface  - Database Design / Data Dictionary  - Task Allocation | 27 |
| 10. | Screen Layout Of Project And Coding | 37 |
| 11. | Testing | 66 |
| 12. | Limitation | 69 |
| 13. | Bibliography | 70 |

Project Profile

Introduction :-

The Quiz Desktop Application is basically desktop application. This application mainly two roles Admin and Student. Admin have all the privileges like a manage student section, add, update, delete question etc. . Admin can able to show all student result, all question and all student details. Student section student register itself and take assessment to check their knowledge and show result. Student can take multiple quiz and select language which e want to check knowledge. Student can update, delete profile and forgot password. This project is being developed to automate all the school/college exam activities.

Objective :-

The main objective of this software to develop an information system to store, maintain, update and process data relating student knowledge.

Beside These Project :-

- Manages the records of every Student and attempt of every quiz.

- Well designed to usage system such as Hardware, Software and Maintenance.

- Generate different kind of necessary reports and queries.

Goal of the Project :-

- System will take all necessary backups.

- Authorized entry by password access for security reasons.

- Provide a pretty & easy GUI that Helps user about the Software.

Profile Details :-

Project Title :- Quiz Desktop Application

Front-End :- Java

Back-End :- MYSQL

Developed By :- Bhavin Bhesaniya

Mayur Solanki

Project-Guide :- Mr. Milind Anandpara

Time Duration :- 3 Month

Module :- Admin

User

Submitted To :- B.K.N.M.U.

Requirement Gathering

#### System Analyses

##### System Analyses is the collecting all the information from all the student then collect all information to obtain a clean and through understanding of the places to be developed with view to removing all ambiguities and inconsistencies from the initial student of the problem

* Questionaries
* Interview
* Observation
* Record Reviewing

#### Identification Of Need

The daily activities pointing to smooth running of a Service Centre involves recording of` large volume of data in record books. The management of student need to be managed properly. In addition, timely preparations of several reports are required. All these require managing services quickly and efficiently.

#### Preliminary Investigation

A request to receive assistance from the information system can be in made for many reasons. But in each case the requested when that request is made, the first system activity the preliminary investigation begins. Collecting the information in this phase through personal interviews.

#### Questionaries :-

Q-1. Which type of application do you want to develop?

- Windows Application

Q-2. What will the purpose of this windows application?

- Test the student knowledge and help him to grow

Q-3. Do You have any type of windows application?

- No

Q-4. Tell me functionality that you want in your application?

- 1) Two main roles admin and student

2) Admin have all rights

3) Student can only view test, asset test and show result

Q-5. You need any payment gateway method in your application?

- No

Q-6. How much time you want to complete this application?

- 2 to 3 Month

Q-7. Give title of your application?

- Fun With Learn

Q-8. Do you have any logo?

- No

Q-9. Which user can log in this application?

- Only Authorized

Q-10. What is the budget for your application?

- 12,000 to 15000

Project Cost Estimation

Line Of Code :-

#### As the name suggests, LOC count the total number of lines of source code in a project.

#### The units of LOC are:

* KLOC- Thousand lines of code
* NLOC- Non-comment lines of code
* KDSI- Thousands of delivered source instruction

Advantages :-

* Universally accepted and is used in many models like COCOMO.
* Estimation is closer to the developer’s perspective.
* Simple to use.

Disadvantages :-

* Different programming languages contain a different number of lines.
* No proper industry standard exists for this technique.
* It is difficult to estimate the size using this technique in the early stages of the project

Line Of Code In Our Project :-

We develop our project in windows system so 20 kilobytes is used in 400 lines.

Perform line of code : 70

Per Class line of code : 60

No of line of code per form : 130

Total Line Of Code :-

= Form \* Per Form Line Of Code

= 12 \* 130

= 1560

KLOG = line of code / kill byte :-

= 1560 / 20

= 78

Project Cost Estimation :-

Cocomo model :-

Total Person = 2

Total Duration = 2 Month

Project Estimation (e) = total person \* KLOC

= 2 \* (78 x 78)

= 12,168

Total Estimation :-

Total Unit Per Day = total hours day \* cost of unit per hour

= 0.5 \* 0.2

= 1.0

Total Working Day = total project day – holyday

= 90 – 15

= 75

Total Unit = per day unit \* total working day

= 75 \* 1

= 75

Total Cost Of Bill = total \* price per unit

= 75 \* 10

= 750

Total Travelling Cost = 700

Computer (Reupdate) = 500

Total Project Cost = light bill + travelling cost + project estimate cost + computer(Reupdate cost)

= 750 + 700 + 12168 + 500

= 14200 Rs.

#### Feasibility Study

##### Technical Feasibility

The main aim for technical feasibility study is to determine whether it is possible to develop the proposed system with the present technologies available and study the technical requirement.

Technical requirement :-

|  |  |
| --- | --- |
| Hardware | Software |
| - 1 GB RAM | - Windows 10, 8, 7 |
| - 500 MB Hard Disk | - NetBeans IDE Version 12 |
| - Intel i5 Processor | - JDK 11 |
| - Internet | - MySQL Server |

Economical Feasibility

The company already possesses the required hardware and software. There is no investment in hardware and software. The benefits of installing the application lie in the speedy processing of data, faster retrieval of information and increasing volume of data, and all these with greater accuracy and consistency. To sum up, the benefits are great and cost is minimal. Therefore, the project is economically feasible.

##### Operational Feasibility

The system is expected to work smoothly when developed and installed. There has been participation of management and the computer operators in planning and development of the system. There will be a slight change in the format of the reports to which management agreed. There is no disturbance in organizational structure of the company. The new application for the system will have Graphical User Interfaces (GUI). The applications with GUI are very easy to handle and operate. They need to be instructed regarding using of the application software.

Technology Used

Tools / Environment Used :-

Software :-

|  |  |
| --- | --- |
| Platform | Windows 10 Student Edition |
| IDE(Integrated Development Environment) | NetBeans 12.0 LTS |
| RDBMS (Database) | MySQL 8.0.1 |
| JDK(Java Development Kit) | 11 |

Hardware :-

|  |  |
| --- | --- |
| Processor | Intel I5 4th Generation |
| RAM | 8GB DDR4 |
| Laptop | HP 15 3047 |
| Hard Disk | 256 GB SSD |

Software Requirement Run Application :-

- NetBeans IDE

- JDK(Java Development Kit)

- MySQL Server

System Requirement :-

- Minimum Requirement :- 4GB RAM / i3 Processor / 30GB Disk Space

- Recommended :- 8GB RAM / i5 Processor / 50GB Disk Space

Language and Software Description

In this project we used java as programming language, Netbeans as IDE to develop project and MySQL as database.

JAVA :-

Java has been one of the most popular programming languages for many years.

Java is Object Oriented. However, it is not considered as pure object oriented as it provides support for primitive data types (like int, char, etc)

The Java codes are first compiled into byte code (machine independent code). Then the byte code runs on **J**ava **V**irtual **M**achine (JVM) regardless of the underlying architecture.

Java syntax is similar to C/C++. But Java does not provide low level programming functionalities like pointers. Also, Java codes are always written in the form of classes and objects.

Java is used in all kind of applications like Mobile Applications (Android is Java based), desktop applications, web applications, client server applications, enterprise applications and many more.

When compared with C++, Java codes are generally more maintainable because Java does not allow many things which may lead bad/inefficient programming if used incorrectly. For example, non-primitives are always references in Java. So we cannot pass large objects (like we can do in C++) to functions, we always pass references in Java. One more example, since there are no pointers, bad memory access is also not possible. When compared with Python, Java kind of fits between C++ and Python. The programs written in Java typically run faster than corresponding Python programs and slower than C++. Like C++, Java does static type checking, but Python does not.

History Of Java :-

Java’s history is very interesting. It is a programming language created in 1991.James Gosling, Mike Sheridan, and Patrick Naughton, a team of Sun engineers known as the **Green team**initiated the Java language in 1991.

**Sun Microsystems** released its first public implementation in 1996 as **Java 1.0**. It provides no-cost -run-times on popular platforms. Java1.0 compiler was re-written in Java by Arthur Van Hoff to strictly comply with its specifications.

With the arrival of Java 2, new versions had multiple configurations built for different types of platforms. **James**Gosling in 1995 developed Java, who is known as the Father of Java. Currently, Java is used in mobile devices, internet programming, games, e-business, etc.

Features Of Java :-

**1. Platform Independent :-**

Compiler converts source code to bytecode and then the JVM executes the bytecode generated by the compiler. This bytecode can run on any platform be it Windows, Linux, macOS which means if we compile a program on Windows, then we can run it on Linux and vice versa. Each operating system has a different JVM, but the output produced by all the OS is the same after the execution of bytecode. That is why we call java a platform-independent language.

**2. Object-Oriented Programming Language :-**

Organizing the program in the terms of collection of objects is a way of object-oriented programming, each of which represents an instance of the class.

The four main concepts of Object-Oriented programming are:

- Abstraction

- Encapsulation

- Inheritance

- Polymorphism

**3.** **Simple :-**

Java is one of the simple languages as it does not have complex features like pointers, operator overloading, multiple inheritances, Explicit memory allocation.

**4.** **Robust :-**

Java language is robust that means reliable. It is developed in such a way that it puts a lot of effort into checking errors as early as possible, that is why the java compiler is able to detect even those errors that are not easy to detect by another programming language. The main features of java that make it robust are garbage collection, Exception Handling, and memory allocation.

**5.** **Secure :-**

In java, we don’t have pointers, and so we cannot access out-of-bound arrays i.e it shows **ArrayIndexOutOfBound Exception** if we try to do so. That’s why several security flaws like stack corruption or buffer overflow is impossible to exploit in Java.

**6.** **Distributed :-**

We can create distributed applications using the java programming language. Remote Method Invocation and Enterprise Java Beans are used for creating distributed applications in java. The java programs can be easily distributed on one or more systems that are connected to each other through an internet connection.

**7.** **Multithreading :-**

Java supports multithreading. It is a Java feature that allows concurrent execution of two or more parts of a program for maximum utilization of CPU.

**8.** **Portable :-**

As we know, java code written on one machine can be run on another machine. The platform-independent feature of java in which its platform-independent bytecode can be taken to any platform for execution makes java portable.

**9. High Performance :-**

Java architecture is defined in such a way that it reduces overhead during the runtime and at some time java uses Just In Time (JIT) compiler where the compiler compiles code on-demand basics where it only compiles those methods that are called making applications to execute faster.

**10. Dynamic flexibility :-**

Java being completely object-oriented gives us the flexibility to add classes,  new methods to existing classes and even creating new classes through sub-classes. Java even supports functions written in other languages such as C, C++ which are referred to as native methods.

**11. Sandbox Execution :-**

Java programs run in a separate space that allows user to execute their applications without affecting the underlying system with help of a bytecode verifier. Bytecode verifier also provides additional security as it’s role is to check the code for any violation access.

**12. Write Once Run Anywhere :-**

As discussed above java application generates ‘.class’ file which corresponds to our applications(program) but contains code in binary format. It provides ease t architecture-neutral ease as bytecode is not dependent on any machine architecture. It is the primary reason java is used in the enterprising IT industry globally worldwide.

**13. Power of compilation and interpretation :-**

Most languages are designed with purpose either they are compiled language or they are interpreted language. But java integrates arising enormous power as Java compiler compiles the source code to bytecode and JVM  executes this bytecode to machine OS-dependent executable code.

Netbeans IDE :-

**NetBeans** is an open-source  IDE(integrated development environment) for Java. NetBeans allows applications to be developed from a set of modular software called *modules*. NetBeans runs on Windows, Mac and Linux. In addition to Java development, it has extensions for other languages like php, c++, html and java script. Applications based on NetBeans, including the NetBeans IDE, can be extended by [third party developers](https://en.wikipedia.org/wiki/Third_party_developer).

MySQL Database :-

MySQL is a relational database management system based on the Structured Query Language, which is the popular language for accessing and managing the records in the database. MySQL is open-source and free software under the GNU license. It is supported by **Oracle Company**.

**Database Concept:-**

A database system is basically a computer based record keeping system. The collection of data, usually referred to as the database contains information about one particular enterprise . It maintains any information that may be necessary to the decision –making processes involved in the management of the organization.

A database may also may be defined as a collection of interrelated data store together to serve multiple applications; a common and controlled approach is used in adding new data and in modifying and retrieving existing data within the database. The data is structured so as to provide a foundation for future application development.

**Data definition Language (DDL):-**

A database is specified by asset of definition, which is expressed by a specified language called a data definition language (DDL). The result of the compilation of DDL statement is a set of table, which is stored in a special file called the data dictionary (or directory).

A data dictionary is a file that contains metadata, i.e. “data about data”. This file is consulted before actual data is read or modified in the database system. The storage structure and accessing methods are specified by a set of definition of special type of DDL called a data storage & definition language.

**Data Manipulation Language (DML):-**

A data manipulation language (DML) is a language that enables user to access or manipulate data as organized by the appropriate data model.

There are basically two types:

1. Procedural DMLs requires the user to specify what data is needed and how to get it.
2. Database manager- It provides the interface between the low level data stored in the database and the application program and queries submitted to the system.
3. Query processor- It translates statement in query language into low level instruction that database manager can understands.
4. DML pre-complier- It converts the DML statement embedded in an application program to normal procedure calls in the host language.
5. DDL compiler- It converts DDL statements to set of tables containing metadata, or “data about data”.

**AN OVERVIEW OF DATABASE TECHNOLOGY :-**

**Relational Model:-**

A relational model consists of a collection of tables each of which is assigned a unique name. a row in table represents a relationship among a set of values. Since a table is a collection of such relationship there is close correspondence between the concept of the table and the mathematical concept of relation, from which the relational data model takes its name. The relational model was propounded by E.F.Codd of the IBM and has since been acknowledged as a very important concept in DBMS. The relational model has established itself as primary data model for commercial data processing application.

**Domain : -**

A domain is a pool of values from which the actual values appearing in a given column are drawn. A domain is said be atomic if element of the domain are considered to be invisible units. For each attribute, there is a set of permitted values called the domain of that attribute. For all relation r, the domain of all attributes of be atomic. A domain is atomic if elements of the domain are considered to be invisible units.

**Entity :-**

An entity is an object that exists and is distinguishable from other objects. An entity set is a set of entities of the same type. An entity is represented by a set of attributes. Attributes- An attribute is a function, which maps from an entity set into a domain. For each attribute there is a set of permitted values, called the domain of that attribute.

**Keys :-**

It is important to be able to specify how entities and relationships are distinguishable. Super key- A super key is asset of one or more attributes which; taken collectively, allow us to identify uniquely an entity in the entity set. Candidate key- A super key may contain extraneous attributes. If K is a super key, then so is any superset o K. we are often interested in super keys for which no proper subset is a super key. Such a minimal super keys are called candidate key.

**Primary key :-**

A primary is a set of one or more attributes that can uniquely identified tuples with in the relational. It denotes a candidate key that is chosen by the database designer as the primary means of identifying entities with an entity set.

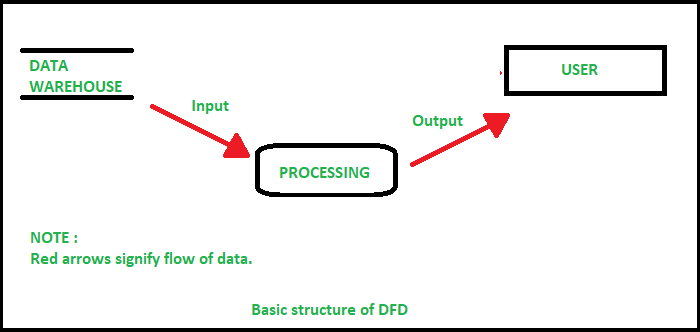
Relational algebra- The relational algebra is a collection of operations on relations .Each operation takes places one or more relations as its operands and procedure another relation as its result. It consists of a set of operations that take one or two fundamental operational operations in the relational algebra are select, project, Cartesian product, rename union and set difference.

**Relational Commercial Language:**

SQL (Structured Query Language):- SQL is a language that enables you to create and operate on relational database, which are set of related information stored in tables. SQL uses the combination of relational algebra and relational calculus constructs. The original version was developed at IBM’s San Jose Research Laboratory. This language originally called sequel, was implemented as part of the system project in early 1970s.

Data Flow Diagram

**DFD** is the abbreviation for **Data Flow Diagram**. The flow of data of a system or a process is represented by DFD. It also gives insight into the inputs and outputs of each entity and the process itself. DFD does not have control flow and no loops or decision rules are present. Specific operations depending on the type of data can be explained by a flowchart. Data Flow Diagram can be represented in several ways. The DFD belongs to structured-analysis modelling tools. Data Flow diagrams are very popular because they help us to visualize the major steps and data involved in software-system processes.



0 Level Diagram :-

It is also known as a context diagram. It’s designed to be an abstraction view, showing the system as a single process with its relationship to external entities. It represents the entire system as a single bubble with input and output data indicated by incoming/outgoing arrows.

Admin

User

1 - Level Diagram :-

In 1-level DFD, the context diagram is decomposed into multiple bubbles/processes. In this level, we highlight the main functions of the system and breakdown the high-level process of 0-level DFD into subprocesses

(User Module)

User

Login

Quiz Home

Feed Back

Profile

Register Quiz Question Feedback

Marks

(Admin Module)

Admin

Login

User Info

Update / Delete Question

Add Question

Write Update Read Read

Delete

Quiz Question Quiz Question Register

Quiz Marks

Quiz Question

Gantt Chart

A Gantt chart, commonly used in project management, is one of the most popular and useful ways of showing activities (tasks or events) displayed against time. On the left of the chart is a list of the activities and along the top is a suitable time scale. Each activity is represented by a bar; the position and length of the bar reflects the start date, duration and end date of the activity.

Gantt charts can be used in managing projects of all sizes and types. These may include building infrastructure like dams, bridges, and highways. They may also include software development and other technologies. Project management tools, such as Microsoft Visio, Project, SharePoint, and [Excel,](https://www.investopedia.com/articles/personal-finance/032415/importance-excel-business.asp) or specialized software, such as Gantt or Match ware, can help in designing Gantt charts.

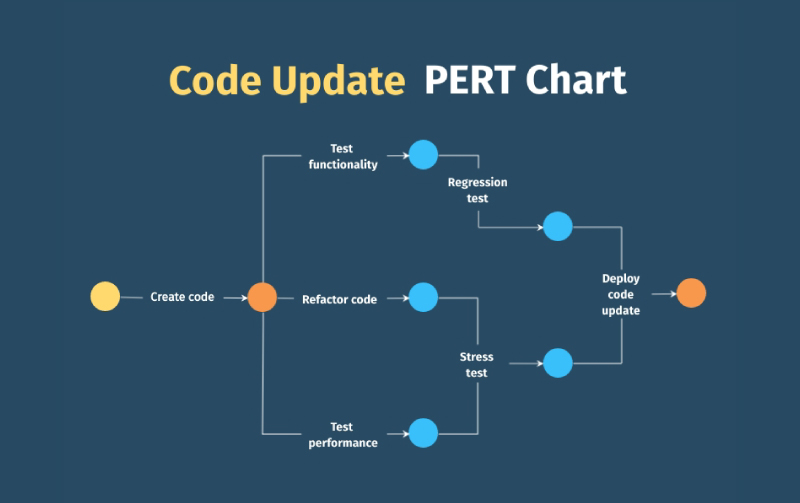
|  |  |
| --- | --- |
| Development Phase | Days |
| Investigation | 17/7/21 |
| System Analyses | 1/8/21 |
| System Design | 15/8/21 |
| Code | 30/8/21 |
| Testing | 15/9/21 |
| Documentation | 25/9/21 |

Pert Chart

A PERT chart is project management tool that provides a graphical representation of a project's timeline. The Program Evaluation Review Technique (PERT) breaks down the individual tasks of a project for analysis. PERT charts are considered preferable to Gannt Chart because they identify task dependencies, but they're often more difficult to interpret.

A PERT chart uses circles or rectangles called nodes to represent project events or milestones. These nodes are linked by vectors or lines that represent various tasks. Dependent tasks are items that must be performed in a specific manner. For example, if an arrow is drawn from Task No. 1 to Task No. 2 on a PERT chart, Task No. 1 must be completed before work on Task No. 2 begins.

Items at the same stage of production but on different task lines within a project are referred to as parallel tasks. They're independent of each other, but they're planned to occur at the same time.



Waterfall Model

The Waterfall Model was the first Process Model to be introduced. It is also referred to as a **linear-sequential life cycle model**. It is very simple to understand and use. In a waterfall model, each phase must be completed before the next phase can begin and there is no overlapping in the phases.

Waterfall approach was first SDLC Model to be used widely in Software Engineering to ensure success of the project. In "The Waterfall" approach, the whole process of software development is divided into separate phases. In this Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially.



The sequential phases in Waterfall model are –

* **Requirement Gathering and analysis** :-

All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.

* **System Design** −The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.
* **Implementation** − With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase. Each unit is developed and tested for its functionality, which is referred to as Unit Testing.
* **Integration and Testing** − All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.
* **Deployment of system** − Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market.
* **Maintenance** − There are some issues which come up in the client environment. To fix those issues, patches are released. Also to enhance the product some better versions are released. Maintenance is done to deliver these changes in the customer environment.

Advantages :-

* Easy to understand, arrange task and use
* Works well for smaller projects where requirements are very well understood.
* Clearly defined stages, Well understood milestones.
* Process and results are well documented.

Disadvantages :-

* No working software is produced until late during the life cycle.
* High amounts of risk and uncertainty.
* Not a good model for complex and object-oriented projects.
* Poor model for long and ongoing projects.
* It is difficult to measure progress within stages

Project Design

User Interface :-

The user interface is perhaps the most important part of an application; it's certainly the most visible. To users, the interface is the application; they probably aren't aware of the code that is executing behind the scenes. No matter how much time and effort you put into writing and optimizing your code, the usability of your application depends on the interface.

User ultimately determines the usability of any application. Interface design is an iterative process; rarely is the first pass at designing an interface for your application going to yield a perfect interface. By getting users involved early in the design process, I have created a better, more usable interface with less effort. Perhaps the most important principle of interface design is one of simplicity.

The best place to start when designing a user interface is to look at some of the best-selling applications from Microsoft or other companies; after all, they probably didn't get to be best sellers because of poor interfaces. One of the main advantages of the Windows operating system is that it presents a common interface across all applications. A user that knows how to use one *Windows-based application* should be able to easily learn any other. Unfortunately, applications that stray too far from the established interface guidelines aren't as easily learned.

In this application, you’ll find many things in common, such as Forms, MDI Interface, Menus, toolbars, ToolTips, context-sensitive menus, Colours, Images, Icons, Fonts, tabbed dialogs, and a simple screen display based application. It's no coincidence that **NETBEANS**  provides the capabilities for adding all of these to your own applications.

I don't need to be an artist to create a great user interface — most of the principles of user interface design are the same as the basic design principles taught in any elementary art class.

Although NetBeans makes it easy to create a user interface by simply dragging controls onto a form, a little planning up front can make a big difference in the usability of your application. You might consider drawing your form designs on paper first, determining which controls are needed, the relative importance of the different elements, and the relationships between controls. We can also borrow from your own experience as a user of software. Think about some of the applications that you have used; what works, what doesn't, and how you would fix it. Remember, however, that your personal likes and dislikes may not match those of your users; you'll need to validate your ideas with them.

Designing Database / Data Dictionary :-

First, We have created the appropriate User Profiles either Graphically orthrough Command, and then Run the concern tables after logging in individually these user profiles.

Create Database :-

A database is an organized collection of structured information, or data, typically stored electronically in a computer system. A database is usually controlled by a [database management system (DBMS)](https://www.oracle.com/in/database/what-is-database/#WhatIsDBMS). Together, the data and the DBMS, along with the applications that are associated with them, are referred to as a database system, often shortened to just database.

**Syntax :-** create database finalquizapp;

Create Registration Table :-

In this table user information such as their name, email and password stored

Figure 1.1.1 :-

**Syntax :-**

create table register

(

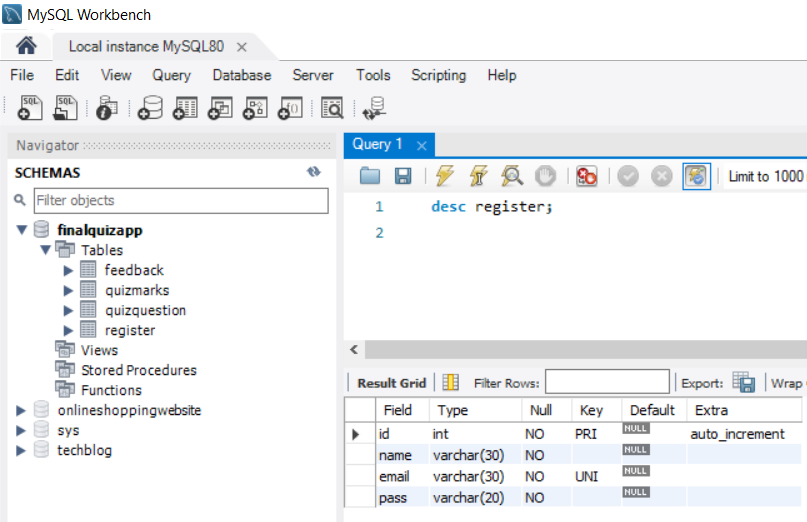
id int primary key auto increment,

name varchar(30) not null,

email varchar (30) not null unique,

pass varchar(20) not null

);



Create QuizQuestion Table :-

In this table we stored question such as id, question, option1, option2, option3, option4, answer and language. tiny text datatype store short information in string. Figure 1.1.2 :-

**Syntax :-**

create table quizquestion

(

id int primary key auto increment,

question varchar(120) not null,

opt1 varchar (120) not null,

opt2 varchar (120) not null,

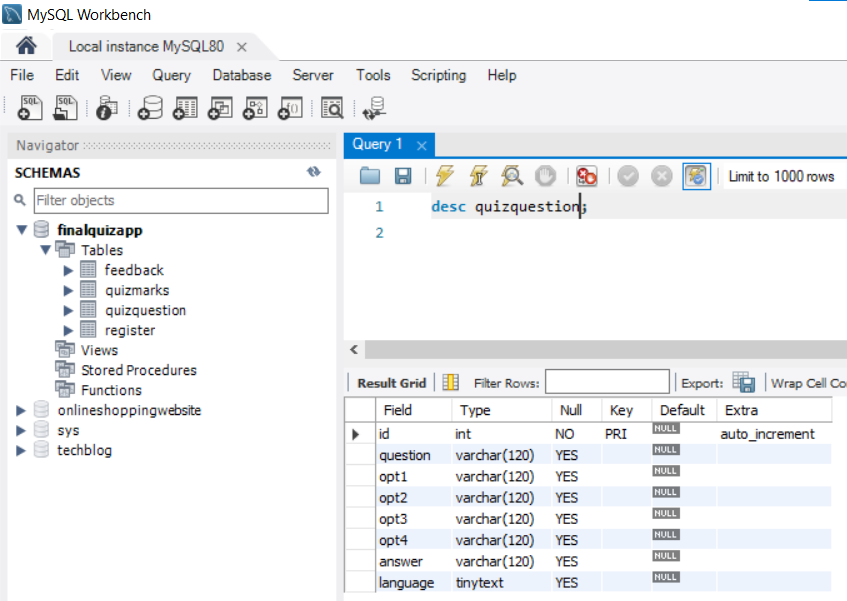
opt3 varchar (120) not null,

opt4 varchar (120) not null,

answer varchar (120) not null,

language tinytext not null

);



Create QuizMarks Table :-

This table store user id with user quiz marks. Every user attempt quiz marks are stored here with the language.

Figure 1.1.3

**Syntax :-**

CREATE TABLE quizmarks

(

id int NOT NULL PRIMARY KEY AUTO\_INCREMENT,

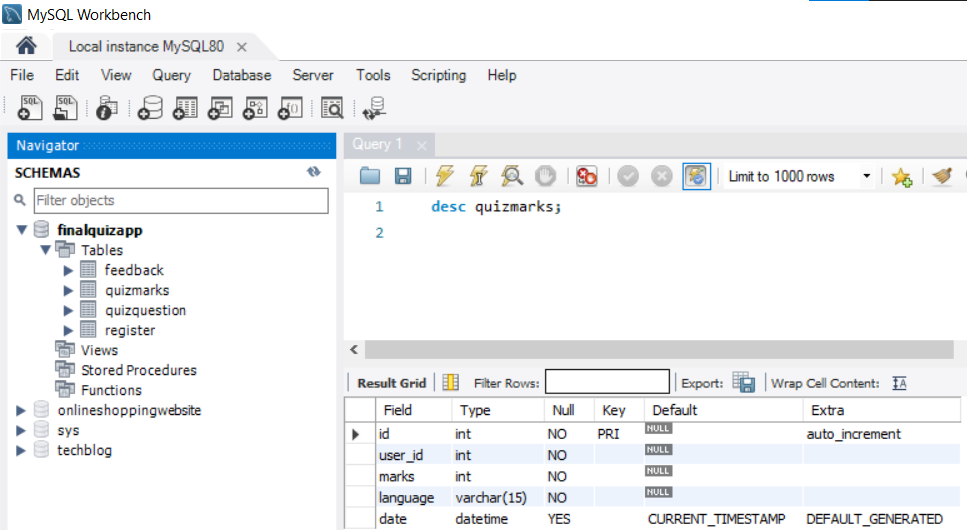
user\_id int NOT NULL,

marks int NOT NULL,

language varchar(15) NOT NULL,

date datetime DEFAULT CURRENT\_TIMESTAMP

);



Create Feedback Table :-

This table use to store user feedback. Feedback is useful to improve our application.

Figure :- 1.1.4

**Syntax :-**

CREATE TABLE feedback

(

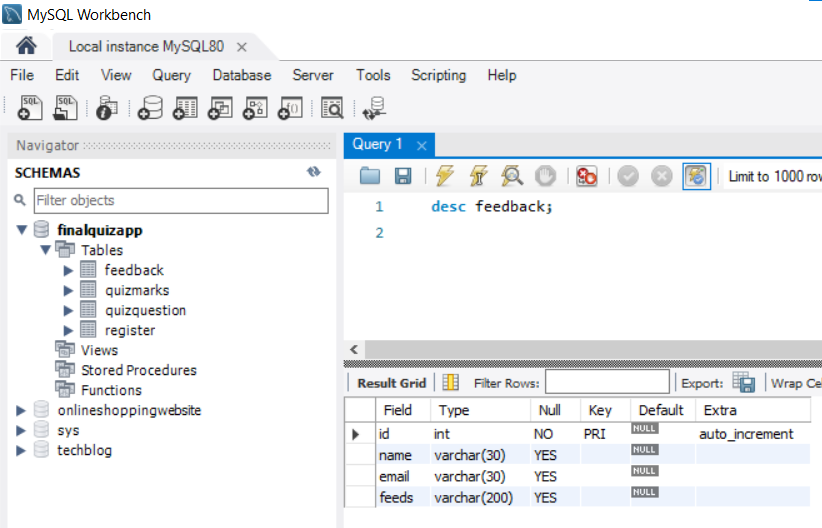
id int NOT NULL PRIMARY KEY AUTO\_INCREMENT,

name varchar(20) DEFAULT NULL,

email varchar(30) DEFAULT NULL,

feeds varchar(200) DEFAULT NULL,

);



Connecting With Database :-

We use JDBC api to connect java application with database. JDBC stands for Java Database Connectivity. JDBC is a Java API to connect and execute the query with the database. It is a part of JavaSE (Java Standard Edition). JDBC API uses JDBC drivers to connect with the database. There are four types of JDBC drivers:

* JDBC-ODBC Bridge Driver,
* Native Driver,
* Network Protocol Driver, and
* Thin Driver

##### Diagram :-



**Step 1 :-**

Download Mysql database :- <https://dev.mysql.com/downloads/installer/> Than Install it.

**Step 2 :-**

Open Netbeans ide and project. We Develop Project in maven so open project files folder and open pom.xml and write following code :-

<dependency>

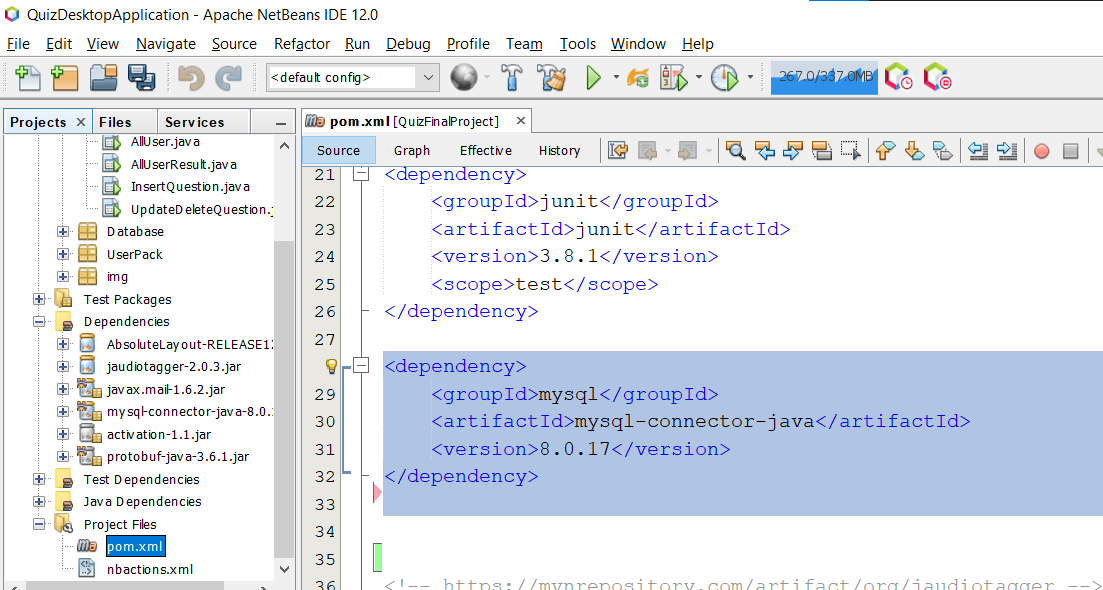
<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<version>8.0.17</version>

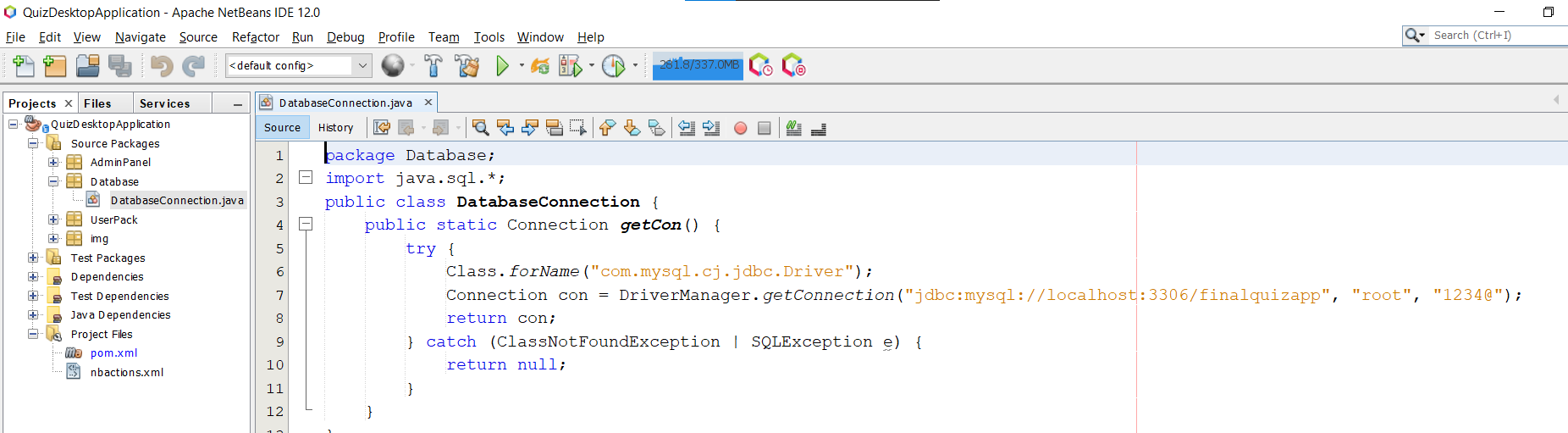
</dependency>

This Code add mysql required driver for communication. You can see in Figure-2.1.1



**Step 3 :-**

Create Package Database and file which show in figure 2.1.2 . This is use for establish connection with the database. This file return connection object which is used for entire application.



Backup Database For Application :-

To ensure that your data is safeguarded, I define a solid backup strategy. You can backup all database as well as One can also backup all data including entire application according to needs on any of the given media: Floppy, Hard Disk, CD, DVD or even on a Zip Disk or a thumb storage drive or on a local network, etc. This software also takes automatic backups.

Today's hardware and software components are many times more reliable than those of the not-so-distant past; however, they will never be perfect, and there is always the chance of having some type of failure. It is therefore prudent to always have some type of "safety net" in place at all times, so that if a situation arises that results in the loss or damage of online data, production can be restored quickly with minimal or no loss of data. When people think of situations that could result in the loss of data, they typically think of disk crashes.

Some of these problems include :-

* Disk subsystem failure
* Systems software failure
* Accidental or malicious use of deletion statements
* Accidental or malicious use of updating statements
* Destructive viruses
* Natural disasters (fire, flood, earthquake, and so on)
* Theft

Task Allocation :-

Task allocation is the way of working process organization when responsibilities and workloads related to one task are distributed among different individuals and organizational units that perform own parts or portions of common work. We often resort to allocating tasks, because actually it is the background for collaboration at tasks. Of course how to allocate tasks correctly depends on multiple factors, especially on specificity of the task, but how to manage task allocation is quite another matter. Allocating tasks can be supported by special tools that can work for any organization and any kind of task.

If a team member didn’t know what exactly their role was, would you be able to clearly  allocate a  task for them? Correct allocations of the tasks also means ensuring that the involved team members are clear on their roles. The more certain they are of their roles and responsibilities, the greater the efficiency and effectiveness in assignment and review of work. This is especially important in the context of letting everyone contribute as well. I divide my task with my project partner Mayur.

Task Sub-Task Allocated To

User Module StratProject.java Mayur

Homepage.java Mayur

Register.java Mayur

Login.java Mayur

Otpconfirm.java Mayur

Forgotpass.java Mayur

Exampage.java Bhavin

QuizHome.java Bhavin

Admin Module InsertQuestion.java Bhavin

Updatedeletequestion.java Bhavin

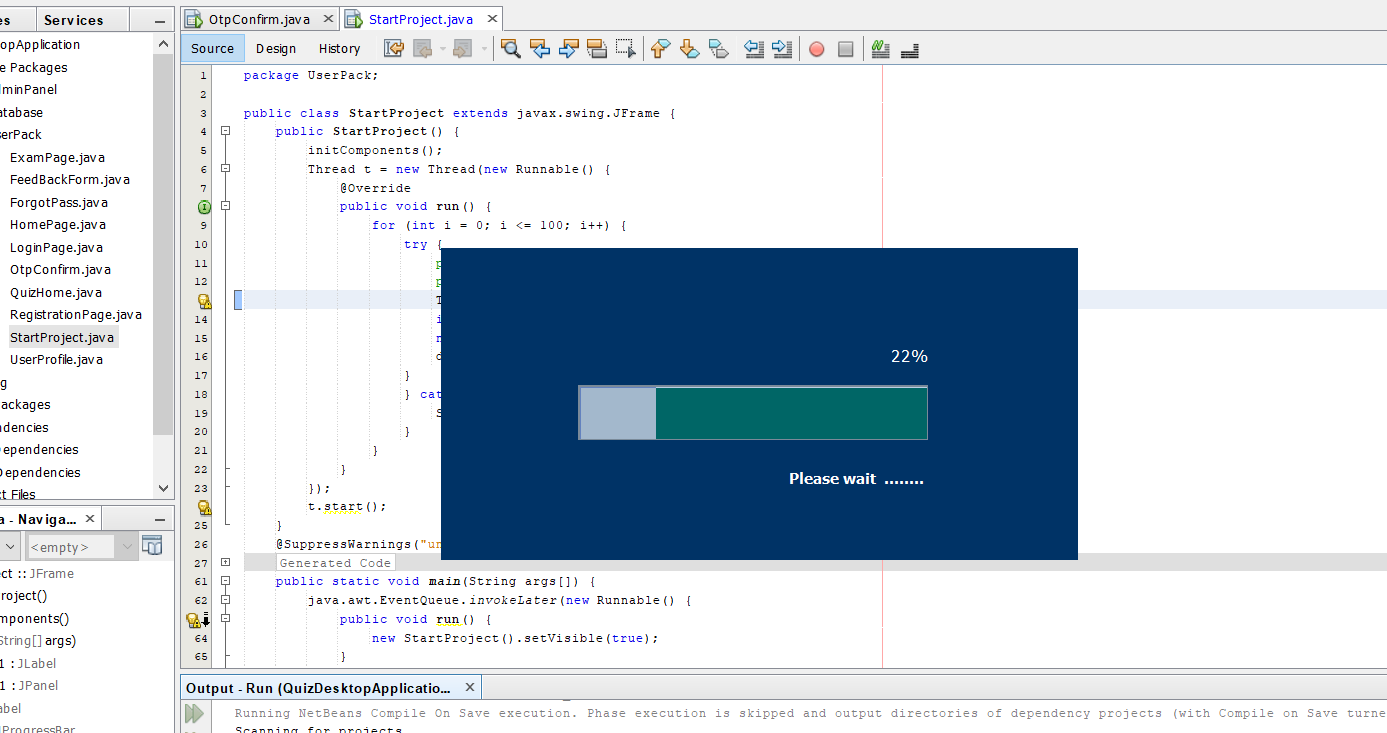
AllUserInfo.java Bhavin

Database Module Databaseconnection.java Bhavin

Screen And Source Code Of The Project

**Screen :- 1 StartProject.java**

**-** Here our project is started. U can see the progress bar using thread.



Code :-

Thread t = new Thread(new Runnable() {

@Override

public void run() {

for (int i = 0; i <= 100; i++) {

try {

pro.setValue(i);

p.setText(Integer.toString(i)+"%");

Thread.sleep(80);

if (i == 99)

{

new HomePage().setVisible(true);

}

}

catch (InterruptedException ex)

{

System.out.print(ex);

}

}

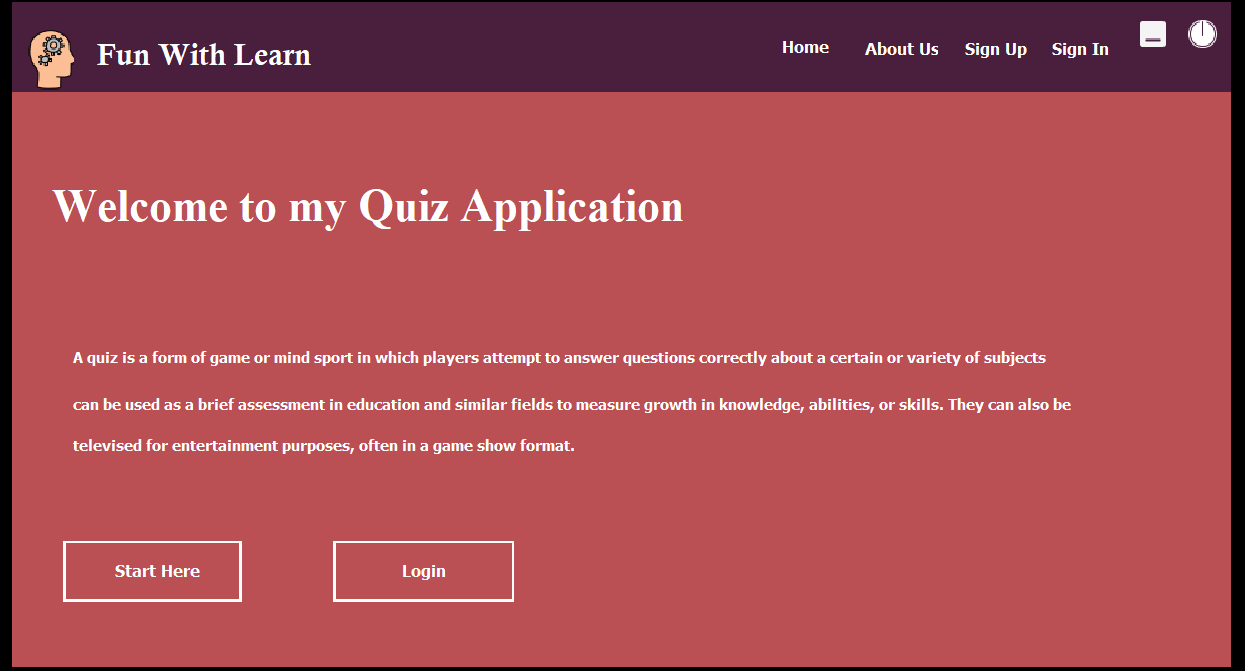
}

});

t.start();

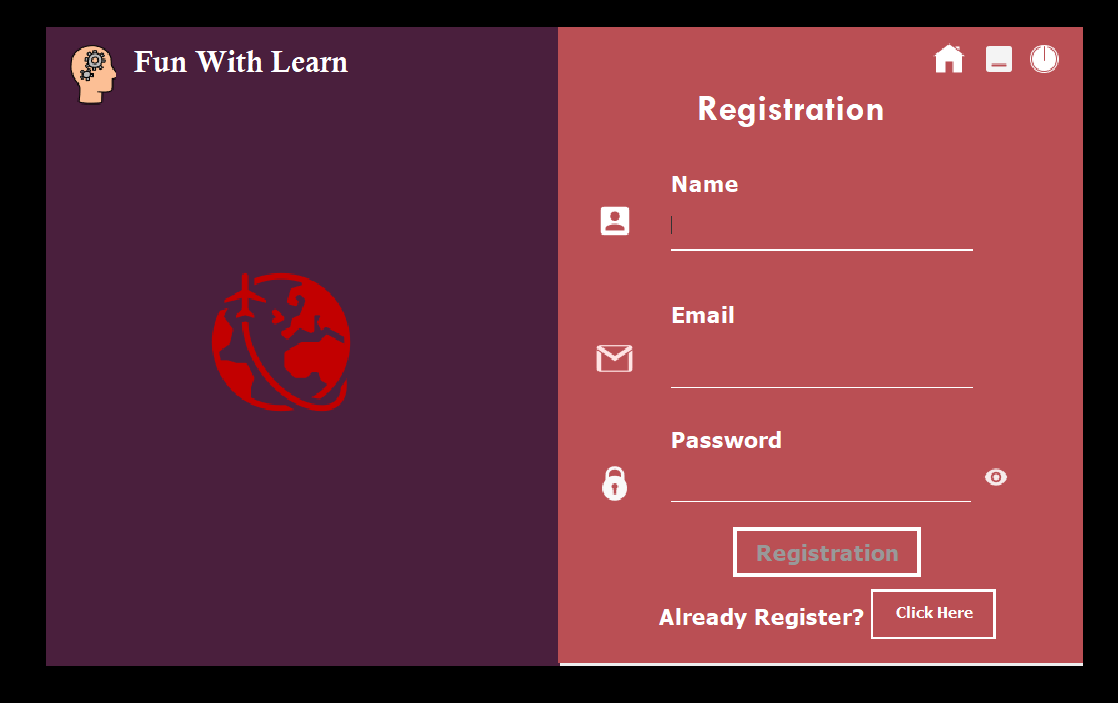
**Screen :- 2 HomePage.java**

In Screen 2 u can see Home, About us, Sign up, Sign in at header bar with minimize and close button. In left side u can quiz title and logo. In the home body section u can see some description about quiz. In about Us section u can connect with our social link. Whenever u click on signup it go to the register screen and if u already register than click on sign in button which move to the login page.

****

**Screen :- 3 Registration.java**

Here user can register with name, email and password which stored in register table database.

****

Code :-

//This Method generate random no for otp

public String generateOTP() {

int randomPin = (int) (Math.random() \* 9000) + 1000;

otp = String.valueOf(randomPin);

return otp;

}

//User click on register button otp send to the user email using this method.

public static void sendEmail(String message, String subject, String to, String from) {

Properties properties = System.getProperties();

properties.put("mail.smtp.host", "smtp.gmail.com");

properties.put("mail.smtp.port", "465");

properties.put("mail.smtp.ssl.enable", "true");

properties.put("mail.smtp.auth", "true");

Session session = Session.getDefaultInstance(properties, new javax.mail.Authenticator() {

@Override

protected PasswordAuthentication getPasswordAuthentication() {

return new PasswordAuthentication("bhavin.otp2021@gmail.com", "Bh@vin12Op");

}

});

session.setDebug(true);

MimeMessage m = new MimeMessage(session);

try

{

m.setFrom(from);

m.addRecipient(Message.RecipientType.TO, new InternetAddress(to));

m.setSubject(subject);

m.setContent(message, "text/html");

Transport.send(m);

}

catch (MessagingException e) {

System.out.println(e);

}

}

//This pattern matching not allowed number or special character in name

String PATTERN = "^[a-zA-Z]{0,30}$";

Pattern patt = Pattern.compile(PATTERN);

Matcher match = patt.matcher(EnterName.getText());

if (!match.matches()) {

ErrorShow.setText("Name format incorrect");

EnterEmail.setEditable(false);

EnterPassword.setEditable(false);

RegistrationBtn.setEnabled(false);

}

else {

ErrorShow.setText(null);

EnterEmail.setEditable(true);

}

//For Validation On Email Pattern

String PATTERN = "^(^[a-zA-Z0-9\_.+-]+@[a-zA-Z0-9-]+\\.[a-zA-Z0-9-.]+$)";

Pattern patt = Pattern.compile(PATTERN);

Matcher match = patt.matcher(EnterEmail.getText());

if (!match.matches()) {

ErrorShow.setText("Email format incorrect");

EnterPassword.setEditable(false);

} else {

ErrorShow.setText(null);

EnterPassword.setEditable(true);

}

//For Password Validation (minimum 8 character with special character, capital and number)

StringPATTERN="^(?=.\*[0-9])(?=.\*[a-z])(?=.\*[A-Z])(?=.\*[@#$%^&+=])(?=\\S+$).{8,}$";

Pattern patt = Pattern.compile(PATTERN);

Matcher match = patt.matcher(EnterPassword.getText());

if (!match.matches()) {

ErrorShow.setText("Not Ful fill Password Condition");

RegistrationBtn.setEnabled(false);

} else {

ErrorShow.setText(null);

RegistrationBtn.setEnabled(true);

}

//Whenever User Click on Registration button this action performed

String name = EnterName.getText();

String uemail = EnterEmail.getText();

String password = EnterPassword.getText();

otp = generateOTP();

String message = otp;

String subject = "OTP Confirmation For Quiz Desktop Application";

to = uemail;

String from = "bhavin.otp2021@gmail.com";

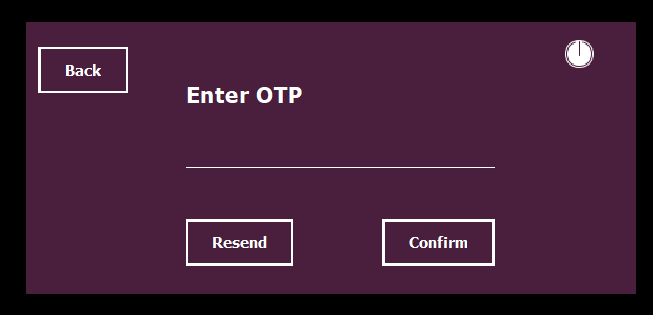
sendEmail(message, subject, to, from);

JOptionPane.showMessageDialog(null, "Plz Confirm With Otp for Login");

new OtpConfirm(otp, to, name, password).setVisible(true);

dispose();

**Screen :- 5 OtpConfirm.java**



//This Constructor get value from parameter for user data that store in database and use to resend mail

public OtpConfirm(String otp, String to, String uname, String passwd) {

initComponents();

receiveotp = otp;

receivemail = to;

name = uname;

password = passwd;

}

//Whenever user press on confirm button condition check if fulfil than data stored in the database otherwise show error message

String a = EnterOtp.getText();

if (a.equals("")) {

JOptionPane.showMessageDialog(null, "please enter otp");

}

else {

if (receiveotp.equals(a)) {

try {

Connection con = DatabaseConnection.getCon();

PreparedStatement pst = con.prepareStatement("insert into register(name,email,pass) values ('" + name + "','" + receivemail + "','" + password + "')");

Statement st = con.createStatement();

ResultSet rst = st.executeQuery("select \* from register where email ='" + receivemail + "'");

if (rst.equals(receivemail)) {

JOptionPane.showMessageDialog(null, "You have Already Register with this email id");

new RegistrationPage().setVisible(true);

dispose();

}

else {

pst.executeUpdate();

ResultSet rs = st.executeQuery("select id from register where email ='" + receivemail + "'");

rs.next();

int user\_id = rs.getInt(1);

new QuizHome(user\_id).setVisible(true);

dispose();

}

} catch (SQLException e) {

JOptionPane.showMessageDialog(null, e);

}

}

else {

JOptionPane.showMessageDialog(null, "Please Enter Right Otp");

}

}

//In case otp not arrive or any other issue You can resend it again

RegistrationPage obj = new RegistrationPage();

receiveotp = obj.generateOTP();

String message = receiveotp;

String subject = "OTP";

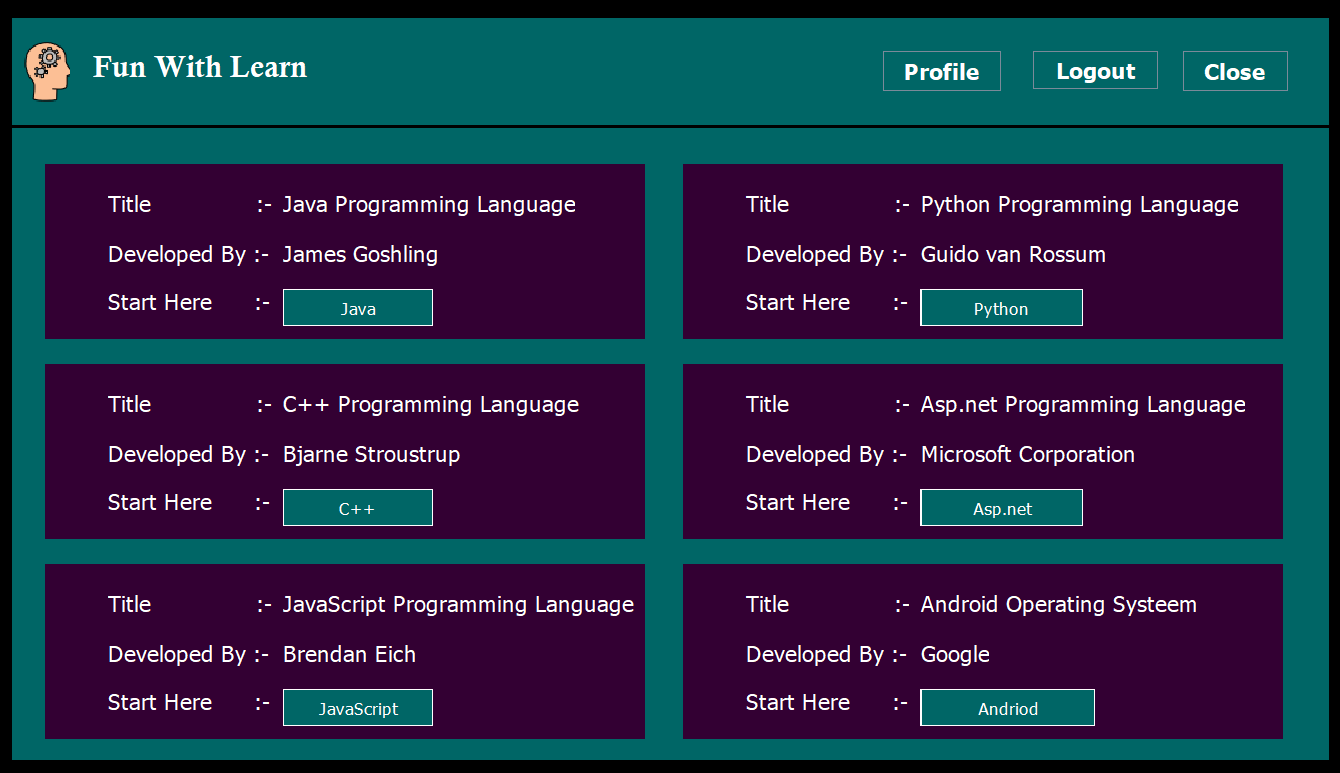
String to = receivemail;

String from = "bhavin.otp2021@gmail.com";

obj.sendEmail(message, subject, to, from);

**Screen :- 6 QuizHomePage.java**

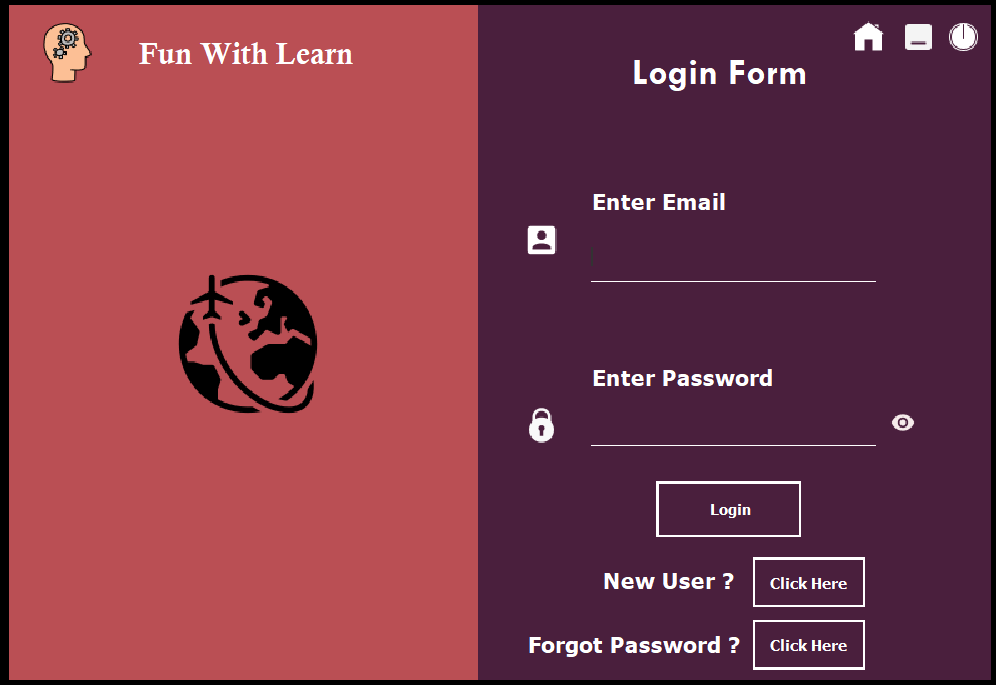
//This is the main screen after user register in the application

****

**Screen :- 7 LoginPage.java**

If User click on logout it move to the following screen

If User Already register than it will write their email and password to login again



Code :-

String umail = EnterEmail.getText();

String passwd = EnterPassword.getText();

try {

Connection con = DatabaseConnection.getCon();

Statement st = con.createStatement();

ResultSet rs = st.executeQuery("select \* from register where email ='" + umail + "' and pass = '" + passwd + "'");

if (umail.equals("") & passwd.equals("")) {

JOptionPane.showMessageDialog(null, "Please Enter Detail");

} else {

if (rs.next()) {

int user\_id = rs.getInt(1);

JOptionPane.showMessageDialog(null, "Successfully login ");

new QuizHome(user\_id).setVisible(true);

dispose();

} else {

if ((umail.equals("b") & passwd.equals("admin")) || (umail.equals("m") & passwd.equals("admin"))) {

new AdminHome().setVisible(true);

dispose();

} else {

JOptionPane.showMessageDialog(null, "Please enter valid username and password ");

}

}

}

}

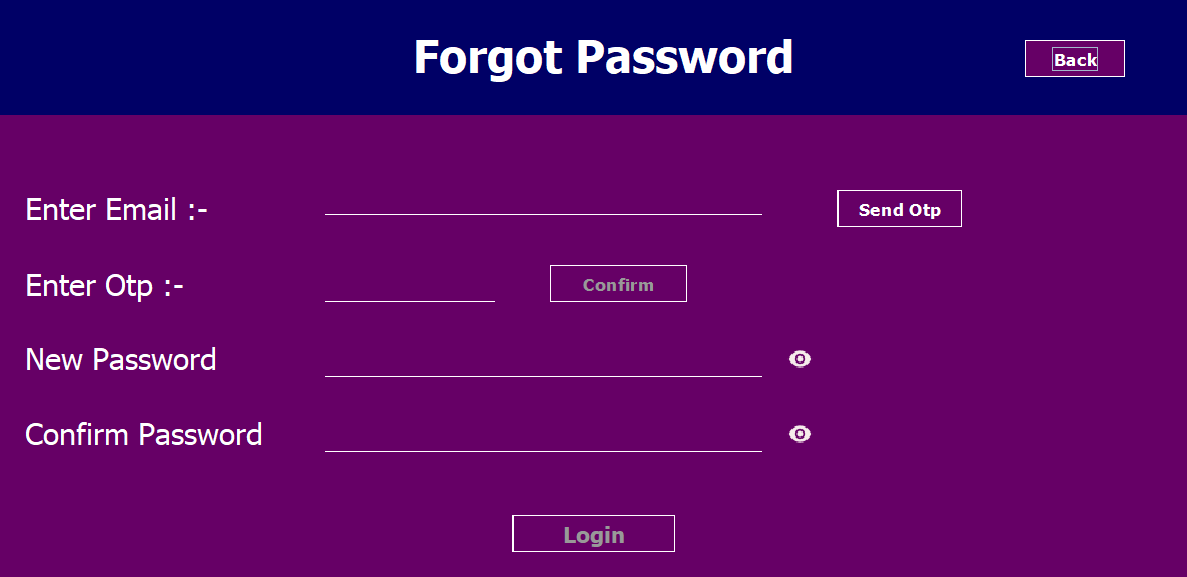
catch (HeadlessException | ClassNotFoundException | SQLException e) {

JOptionPane.showConfirmDialog(this, e);

}

**Screen :- 8 Forgotpassword.java**

//If user need to forgot password than it can reset from here



Code :-

//Check user true email id before click on send otp and then click on confirm otp so they can write password

emailId = MailField.getText();

try {

Connection con = DatabaseConnection.getCon();

Statement st = con.createStatement();

ResultSet rs = st.executeQuery("select \* from register where email = '" + emailId + "'");

if (emaild.equals("")) {

JOptionPane.showMessageDialog(null, "Please enter email id");

} else {

if (rs.next()) {

RegistrationPage obj = new RegistrationPage();

otpw = obj.generateOTP();

String message = otpw;

String subject = "OTP Confirmation For QuizApp";

String to = emailId;

String from = "bhavin.otp2021@gmail.com";

obj.sendEmail(message, subject, to, from);

OtpConfirmBtn.setEnabled(true);

} else {

JOptionPane.showMessageDialog(null, "Please enter true email id which u login earlier");

}

}

} catch (HeadlessException | SQLException e) {

JOptionPane.showMessageDialog(null, e);

}

//Update users’ password

String passId = PassField.getText();

try {

Connection con = DatabaseConnection.getCon();

Statement st = con.createStatement();

st.executeUpdate("update register set pass = '" + passId + "' where email = '" + emailId + "'");

JOptionPane.showMessageDialog(null, "Successfully login");

new LoginPage().setVisible(true);

dispose();

}

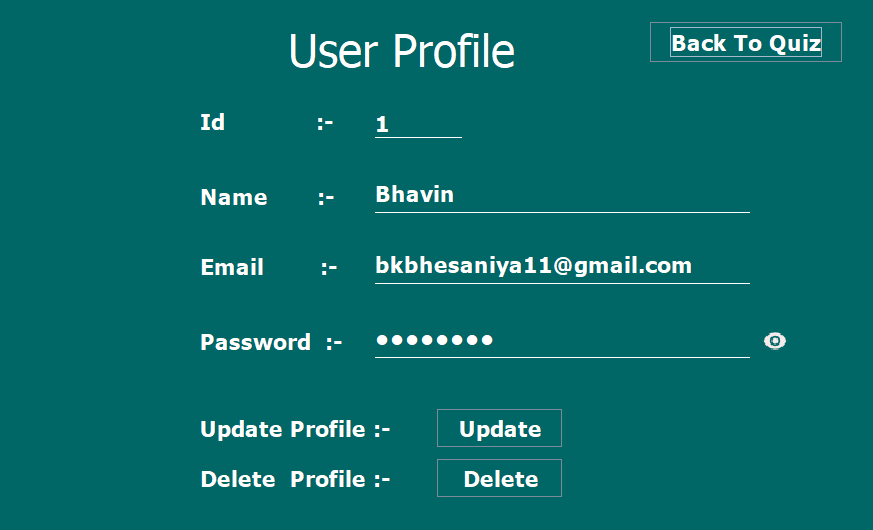
Catch (HeadlessException | SQLException e) {

JOptionPane.showMessageDialog(null, e);

}

**Screen :- 9 UserProfile.java**

//BACK to the screen 6 when user click on the profile button this screen open here user can update name and password and also able to delete his profile.



Code :- Read value from database when screen load

try {

Connection con = DatabaseConnection.getCon();

Statement st = con.createStatement();

ResultSet rs = st.executeQuery("select \* from register where id = '" + uId + "' ");

if (rs.next()) {

IdField.setText(rs.getString(1));

NameField.setText(rs.getString(2));

EmailField.setText(rs.getString(3));

EnterPassword.setText(rs.getString(4));

} }

catch (SQLException e) {

JOptionPane.showMessageDialog(null, e); }

//Update Profile Code

String id = IdField.getText();

String name = NameField.getText();

String email = EmailField.getText();

String passwd = EnterPassword.getText();

try {

Connection con = DatabaseConnection.getCon();

PreparedStatement pst = con.prepareStatement("update register set name=?,email=?,pass=? where id=?");

pst.setString(1, name);

pst.setString(2, email);

pst.setString(3, passwd);

pst.setString(4, id);

pst.executeUpdate();

JOptionPane.showMessageDialog(null, "Successfully updated");

dispose();

new UserProfile(uId).setVisible(true);

} catch (HeadlessException | SQLException e) {

JOptionPane.showMessageDialog(null, e);

}

//Delete Profile Code

String id = IdField.getText();

try {

Connection con = DatabaseConnection.getCon();

PreparedStatement pst = con.prepareStatement("delete from register where id = '" + id + "'"); pst.executeUpdate();

JOptionPane.showMessageDialog(null, "Successfully Deleted");

dispose();

new RegistrationPage().setVisible(true); } catch (SQLException e) {

JOptionPane.showMessageDialog(null, e);

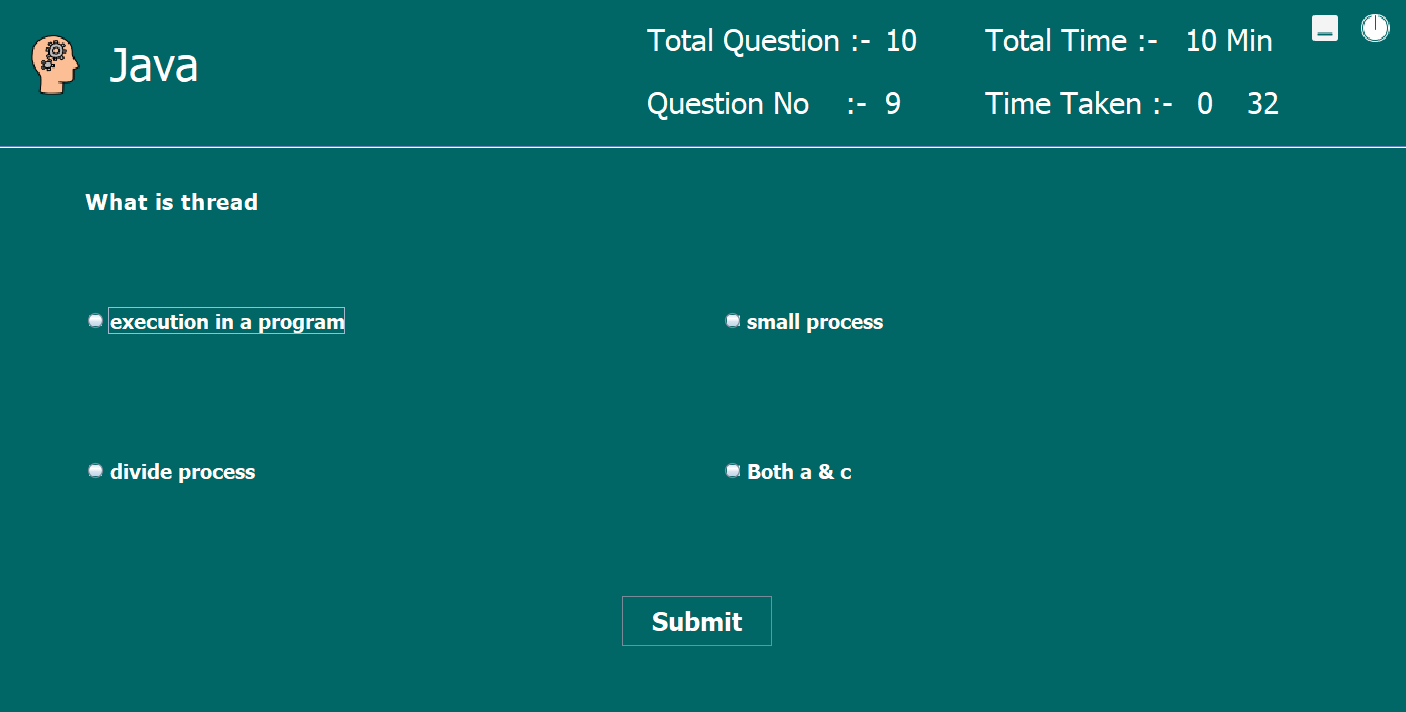
}

**Screen :– 10 ExamPage.java**

//After select an exam based on click button this code fire and go to the exam page

new ExamPage(JavascriptBtn.getText(), userId).setVisible(true);

//This is the exam page select any one option and press next after reach 10 question submit button visible.

****

Code :- For Open Question

public void question() {

try {

Connection con = DatabaseConnection.getCon();

Statement st = con.createStatement();

//Fetch Question From Database

ResultSet rsl = st.executeQuery("select \* from quizquestion where id > " + questionId + " AND language = '" + selectedLanguage + "' LIMIT 1");

while (rsl.next()) {

counter++;

questionId = Integer.parseInt(rsl.getString(1));

QuestionNoUpdate.setText(String.valueOf(counter)); //Set Question Number

QuestionLabel.setText(rsl.getString(2));

Opt1RadioBtn.setText(rsl.getString(3));

Opt2RadioBtn.setText(rsl.getString(4));

Opt3RadioBtn.setText(rsl.getString(5));

Opt4RadioBtn.setText(rsl.getString(6));

answer = rsl.getString(7);

}

} catch (SQLException e) {

JOptionPane.showMessageDialog(null, e);

}

}

Code :- For checking answer and move to the next question

public void answerCheck() {

String studentAnswer;

if (Opt1RadioBtn.isSelected()) {

studentAnswer = Opt1RadioBtn.getText();

} else if (Opt2RadioBtn.isSelected()) {

studentAnswer = Opt2RadioBtn.getText();

} else if (Opt3RadioBtn.isSelected()) {

studentAnswer = Opt3RadioBtn.getText();

} else {

studentAnswer = Opt4RadioBtn.getText();

} //Checking Marks

if (studentAnswer.equals(answer)) {

marks = marks + 1;

}

//Clear Radio Button

Opt1RadioBtn.setSelected(false);

Opt2RadioBtn.setSelected(false);

Opt3RadioBtn.setSelected(false);

Opt4RadioBtn.setSelected(false);

//At last question next button hide and show Submit button

if (counter == 10) {

NextQuestionBtn.setVisible(false);

SubmitBtn.setVisible(true);

}

}

Code :- When click on submit button this code is executed

public void submit() {

answerCheck();

try {

Connection con = DatabaseConnection.getCon();

PreparedStatement pst = con.prepareStatement("insert into quizmarks (user\_id,marks,language) values(" + userId + "," + marks + ",'" + selectedLanguage + "')");

pst.executeUpdate();

String marks1 = String.valueOf(marks);

JOptionPane.showMessageDialog(null, "Your Total Marks Is : " + marks1, "Select", JOptionPane.INFORMATION\_MESSAGE);

dispose();

new QuizHome(userId).setVisible(true);

} catch (SQLException e) {

JOptionPane.showMessageDialog(null, e);

} }

Code :- Timer Logic

time = new Timer(1000, new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

SecondLabel.setText(String.valueOf(sec));

MinuteLabel.setText(String.valueOf(min));

if (sec == 60) {

sec = 0;

min++;

if (min == 5) {

time.stop();

answerCheck();

submit();

}

}

sec++;

}

});

time.start();

Code :- Select only one button

if (Opt1RadioBtn.isSelected()) {

Opt2RadioBtn.setSelected(false);

Opt3RadioBtn.setSelected(false);

Opt4RadioBtn.setSelected(false);

}

**Screen :– 11 FeedbackForm.java**

**//** This page take feedback form user



Code :-

try {

Connection con = DatabaseConnection.getCon();

Statement st = con.createStatement();

ResultSet rs = st.executeQuery("select \* from register where id = '" + userId + "' ");

rs.next();

nm = rs.getString(2);

email = rs.getString(3);

String feedtxt = jTextArea1.getText();

PreparedStatement pst = con.prepareStatement("insert into feedback(name,email,feeds) values ('" + nm + "','" + email + "','" + feedtxt + "')");

pst.executeUpdate();

JOptionPane.showMessageDialog(null, "Thanks You For Your Valueable FeedBack");

System.exit(0);

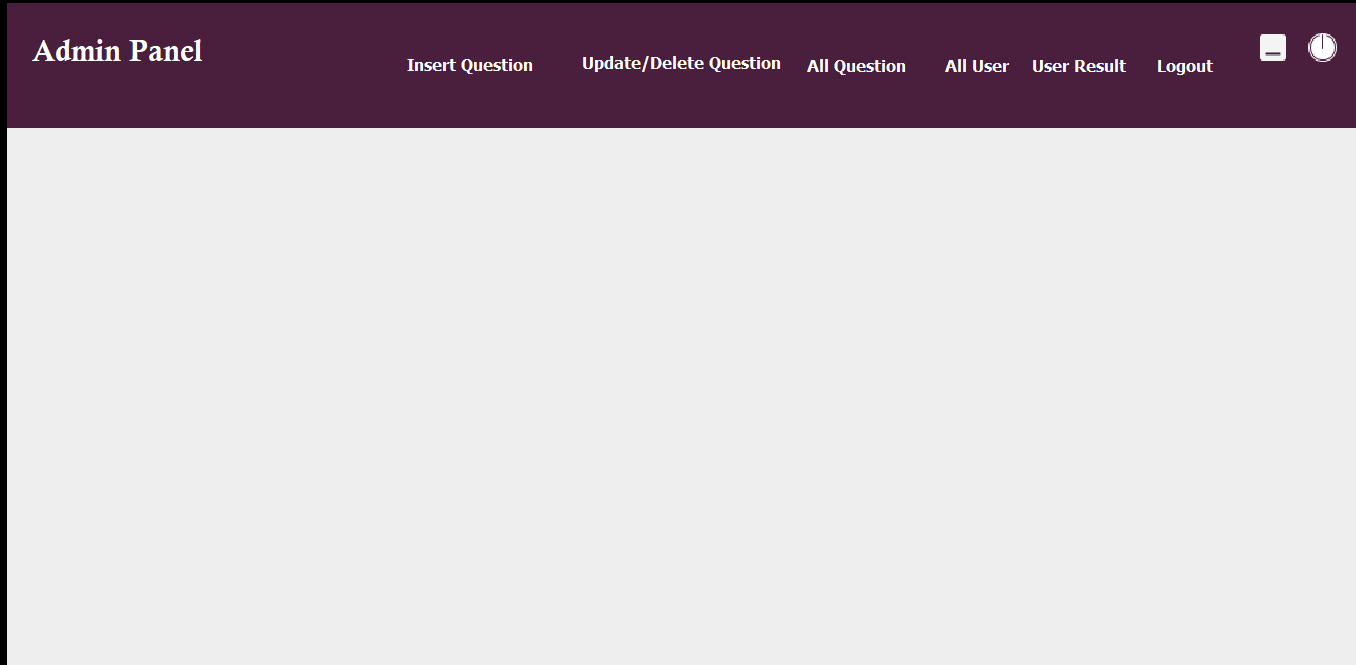
} catch (SQLException e) {

JOptionPane.showMessageDialog(null, e);

}

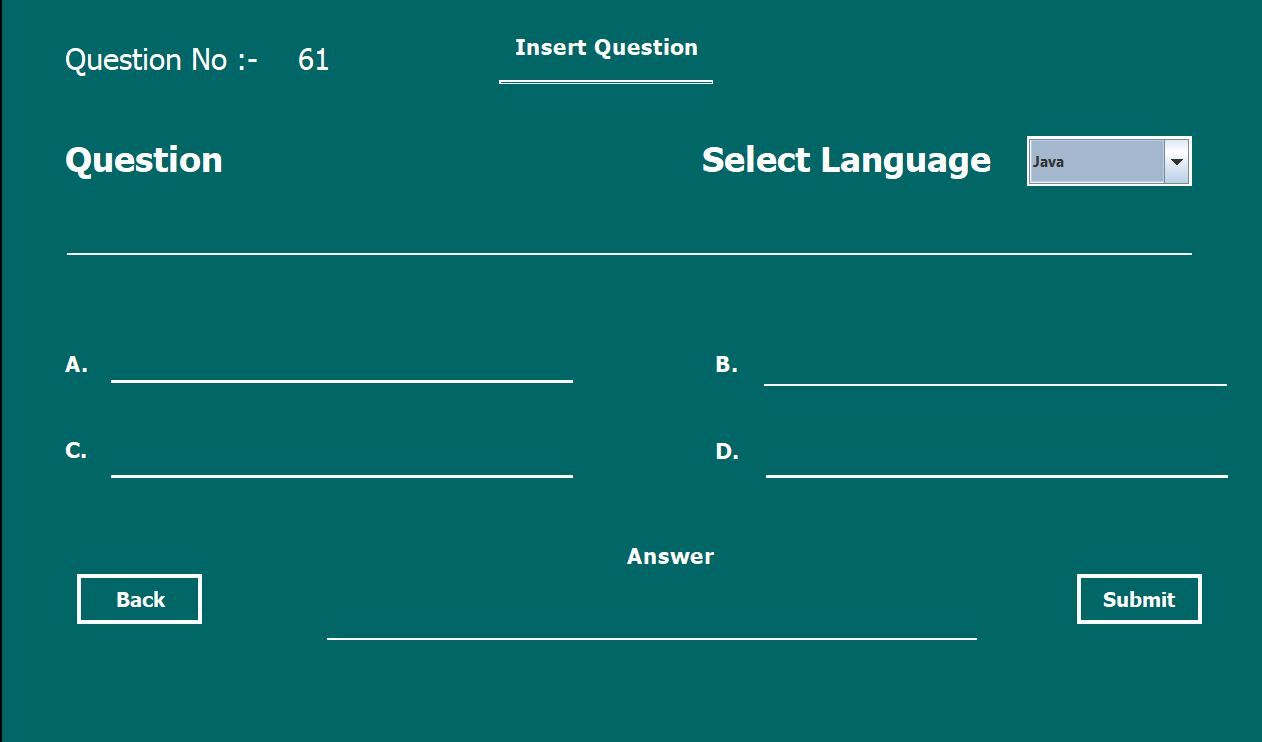
**Screen :- 12 AdminHome.java**

//Admin Side -> Go From -> Login Page to -> Admin Home Here Admin Panel u can choose your option what you want



**Screen :- 13 InsertQuestion.java**

- Here you can add question with 4 option and select language after click on submit button



Code :- Show u question no when frames open

try {

Connection con = DatabaseConnection.getCon();

Statement stmt = con.createStatement();

ResultSet rs = stmt.executeQuery("select count(id) from quizquestion");

if (rs.next()) {

int id = rs.getInt(1);

id = id + 1;

String str = String.valueOf(id);

QuestionIdLabel.setText(str);

} else {

QuestionIdLabel.setText("1");

}

} catch (SQLException e)

{

JOptionPane.showMessageDialog(null, e);

}

Code :- Add Question in the database

String question = NewQuestionField.getText();

String opt1 = Option1.getText();

String opt2 = Option2.getText();

String opt3 = Option3.getText();

String opt4 = Option4.getText();

String answer = RightAnswer.getText();

String lang = String.valueOf(ComboBox.getSelectedItem());

try {

Connection con = DatabaseConnection.getCon();

PreparedStatement ps = con.prepareStatement("insert into quizquestion (question,opt1,opt2,opt3,opt4,answer,language) values ('" + question + "','" + opt1 + "','" + opt2 + "','" + opt3 + "','" + opt4 + "','" + answer + "','" + lang + "')");

if ("".equals(question) | "".equals(opt1) | "".equals(opt2) | "".equals(opt3) | "".equals(opt4) | "".equals(answer)) {

JOptionPane.showMessageDialog(null, "Please Enter All Details");

} else {

ps.executeUpdate();

JOptionPane.showMessageDialog(null, "Successfully updated");

dispose();

new InsertQuestion().setVisible(true);

}

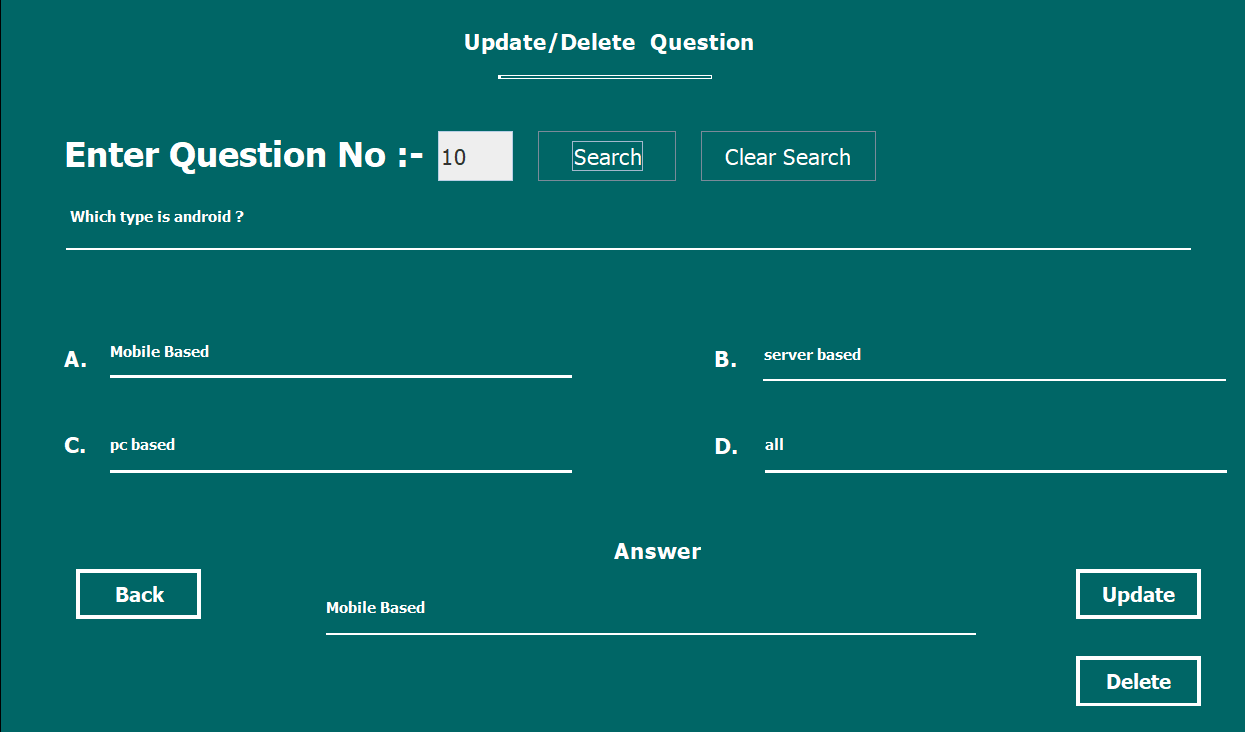
} catch (HeadlessException | SecurityException | SQLException e) {

JOptionPane.showMessageDialog(null, e);

}

**Screen :- 14 Update/Delete Question**

**//** In this frame u can update and delete question using search id.

****

Code :- Search value in textbox if found data display other wise message show

String id = SearchField.getText();

try {

Connection con = DatabaseConnection.getCon();

Statement st = con.createStatement();

ResultSet rs = st.executeQuery("select \* from quizquestion where id = '" + id + "'");

if (rs.next()) {

NewQuestion.setText (rs.getString(2));

Option1.setText(rs.getString(3));

Option2.setText(rs.getString(4));

Option3.setText(rs.getString(5));

Option4.setText(rs.getString(6));

RightAnswer.setText(rs.getString(7));

SearchField.setEditable(false);

UpdateBtn.setEnabled(true);

DeleteBtn.setEnabled(true);

ClearSearchBtn.setEnabled(true);

} else {

JOptionPane.showMessageDialog(null, "Question id not exist");

}

} catch (SQLException e) {

JOptionPane.showMessageDialog(null, e);

}

Code :- Update Question Code

String id = SearchField.getText();

String question = NewQuestion.getText();

String opt1 = Option1.getText();

String opt2 = Option2.getText();

String opt3 = Option3.getText();

String opt4 = Option4.getText();

String answer = RightAnswer.getText();

try {

Connection con = DatabaseConnection.getCon();

PreparedStatement pst = con.prepareStatement("update quizquestion set question=?,opt1=?,opt2=?,opt3=?,opt4=?,answer=? where id=?");

pst.setString(1, question);

pst.setString(2, opt1);

pst.setString(3, opt2);

pst.setString(4, opt3);

pst.setString(5, opt4);

pst.setString(6, answer);

pst.setString(7, id);

pst.executeUpdate();

JOptionPane.showMessageDialog(null, "Successfully updated");

dispose();

new UpdateDeleteQuestion().setVisible(true);

} catch (HeadlessException | SecurityException | SQLException e) {

JOptionPane.showMessageDialog(null, e);

}

Code :- Delete Question Code

String id = SearchField.getText();

try {

Connection con = DatabaseConnection.getCon();

PreparedStatement pst = con.prepareStatement("delete from quizquestion where id = ?");

pst.setString(1, id);

pst.executeUpdate();

JOptionPane.showMessageDialog(null, "Successfully Deleted");

dispose();

new UpdateDeleteQuestion().setVisible(true);

} catch (SQLException e) {

JOptionPane.showMessageDialog(null, e);

}

Code :- Clear Search Button

NewQuestion.setText("");

Option1.setText("");

Option2.setText("");

Option3.setText("");

Option4.setText("");

RightAnswer.setText("");

SearchField.setText("");

SearchField.setEditable(true);

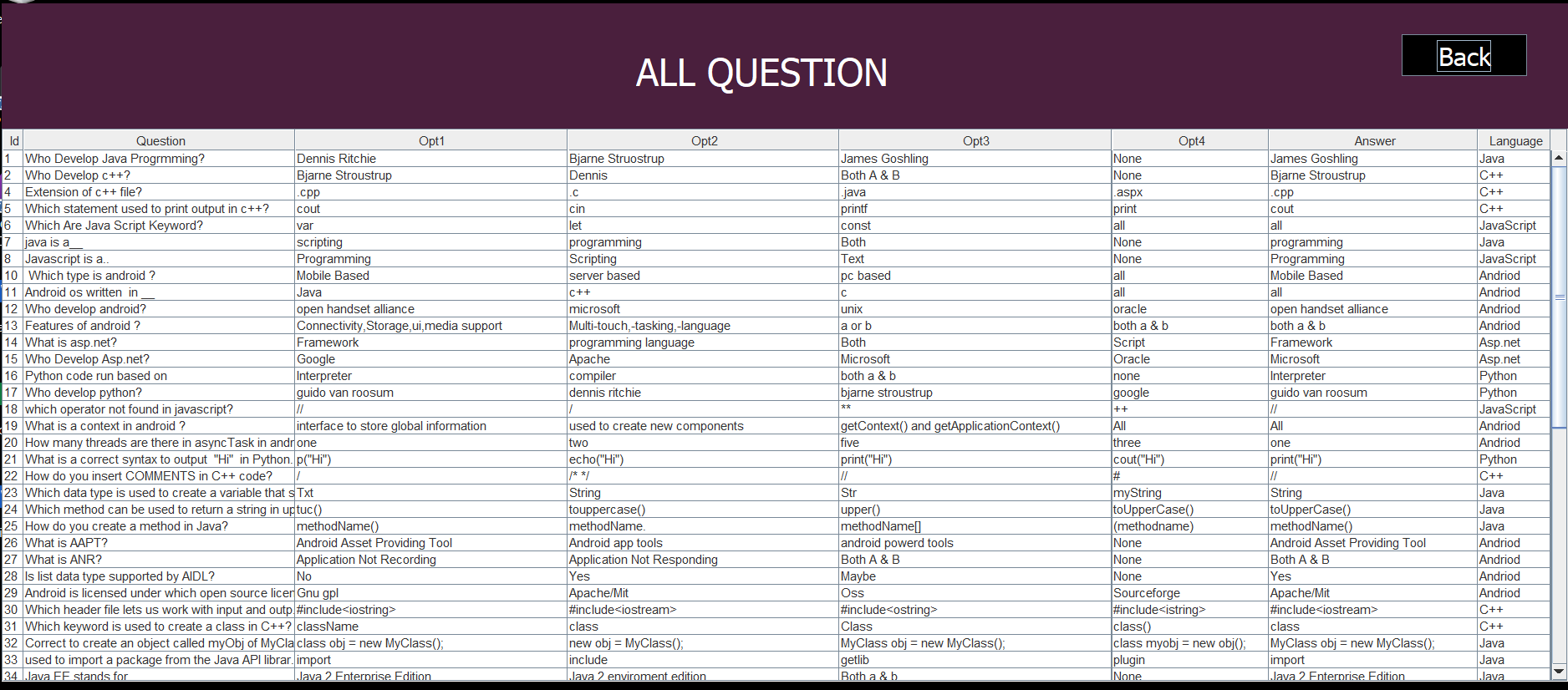
UpdateBtn.setEnabled(false);

DeleteBtn.setEnabled(false);

ClearSearchBtn.setEnabled(false);

**Screen :- 15 AllQuestion.java**

This Frame Show All Question In The Table Format

****

Code :-

try {

Connection con = DatabaseConnection.getCon();

Statement st = con.createStatement();

ResultSet rs = st.executeQuery("select \* from quizquestion");

while (rs.next()) {

String id = String.valueOf(rs.getInt("id"));

String Question = rs.getString("question");

String opt1 = rs.getString("opt1");

String opt2 = rs.getString("opt2");

String opt3 = rs.getString("opt3");

String opt4 = rs.getString("opt4");

String answer = rs.getString("answer");

String lang = rs.getString("language");

String tbData[] = {id, Question,opt1,opt2,opt3,opt4,answer,lang};

DefaultTableModel tblmodel = (DefaultTableModel) AllQuestionTable.getModel();

tblmodel.addRow(tbData);

} } catch (SQLException e) {

JOptionPane.showMessageDialog(rootPane, e);

}

**Screen :- 16 ALLUser.java**

// In this frame u can see all user register in table****

try {

Connection con = DatabaseConnection.getCon();

Statement st = con.createStatement();

ResultSet rs = st.executeQuery("select \* from register");

while (rs.next()) {

String id = String.valueOf(rs.getInt("id"));

String name = rs.getString("name");

String email = rs.getString("email");

String pass = rs.getString("pass");

String tbData[] = {id, name, email, pass};

DefaultTableModel tblmodel = (DefaultTableModel) jTable1.getModel();

tblmodel.addRow(tbData);

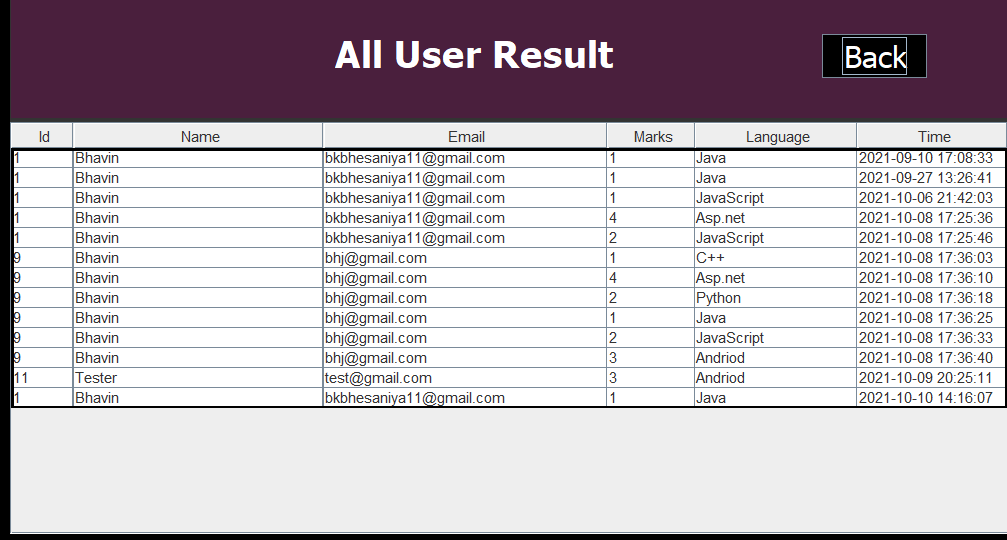
} } catch (SQLException e) {

JOptionPane.showMessageDialog(rootPane, e);

}

**Screen :- 17 AllUserResult.java**

**//** All User Result show in this frame



try {

Connection con = DatabaseConnection.getCon();

Statement st = con.createStatement();

ResultSet rs = st.executeQuery("select register.id, register.name, register.email,quizmarks.id, quizmarks.user\_id,quizmarks.marks,quizmarks. date,quizmarks.language from register inner join quizmarks on register.id=quizmarks.user\_id ");

while (rs.next()) {

String id = String.valueOf(rs.getInt("id"));

String name = rs.getString("name");

String email = rs.getString("email");

String marks = rs.getString("marks");

String lang = rs.getString("language");

String date = rs.getString("date");

String tbData[] = {id, name, email, marks, lang, date};

DefaultTableModel tblmodel = (DefaultTableModel) ResultTable.getModel();

tblmodel.addRow(tbData);

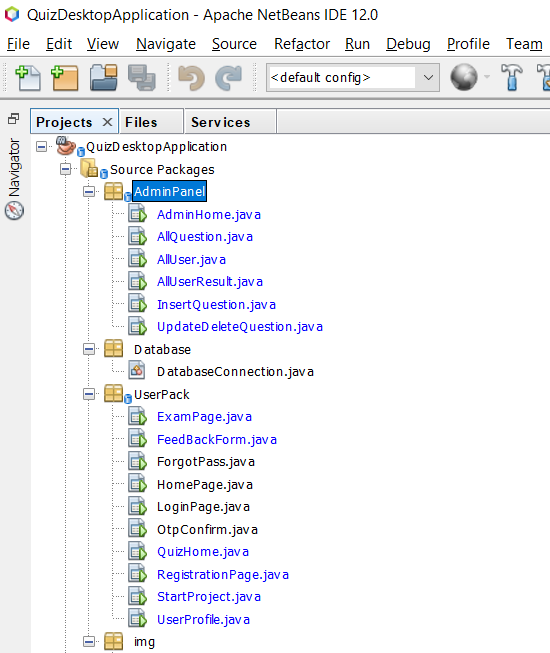
} } catch (SQLException e) {

JOptionPane.showMessageDialog(rootPane, e);

}

**Screen :- 18 Complete File Structure**

//U Can See Complete File Structure In This Application

****

Testing

Testing process contains many types of phases testing can also be divided into various types. Testing is a process to find the error and to combine it with the development of the software.

Error may be occurs when the objects are improperly specified and declared, failed, fault, test case and suite are connected with the phases process.

**BLACK-BOX TESTING :-**

**Black Box Testing**is a software testing method in which the functionalities of software applications are tested without having knowledge of internal code structure, implementation details and internal paths. Black Box Testing mainly focuses on input and output of software applications and it is entirely based on software requirements and specifications. It is also known as Behavioural Testing.

**TYPES OF BLACK-BOX TESTING :-**

* **Functional testing** – This black box testing type is related to the functional requirements of a system; it is done by software testers.
* **Non-functional testing**– This type of black box testing is not related to testing of specific functionality, but non-functional requirements such as performance, scalability, usability.
* **Regression testing**– [Regression Testing](https://www.guru99.com/regression-testing.html) is done after code fixes, upgrades or any other system maintenance to check the new code has not affected the existing code.



**WHITE-BOX TESTING :-**

**White Box Testing** is software testing technique in which internal structure, design and coding of software are tested to verify flow of input-output and to improve design, usability and security. In white box testing, code is visible to testers so it is also called Clear box testing, Open box testing, Transparent box testing, Code-based testing and Glass box testing.

**TYPES OF WHITE-BOX TESTING :-**

* **Unit Testing :-**It is often the first type of testing done on an application.[Unit Testing](https://www.guru99.com/unit-testing-guide.html) is performed on each unit or block of code as it is developed. Unit Testing is essentially done by the programmer. As a software developer, you develop a few lines of code, a single function or an object and test it to make sure it works before continuing Unit Testing helps identify a majority of bugs, early in the software development lifecycle. Bugs identified in this stage are cheaper and easy to fix.
* **Testing for Memory Leaks** :-  Memory leaks are leading causes of slower running applications. A QA specialist who is experienced at detecting memory leaks is essential in cases where you have a slow running software application.
* **White Box Penetration testing :**-  In this testing, the tester/developer has full information of the application’s source code, detailed network information, IP addresses involved and all server information the application runs on.  The aim is to attack the code from several angles to expose security threats
* **White Box Mutation Testing** :- Mutation testing is often used to discover the best coding techniques to use for expanding a software solution.

**Test case :-**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Suit Id – 1**  **Test Id - 1** | | | **Description :- This Test case on admin panel** | | |
| **No.** | **Task** | **Expected Output** | **Actual Result** | **Pass / Fail** | **Remark** |
| **1.** | **Insert Question** | **Insert Successfully** | **Insert Successfully** | **Pass** |  |
| **2.** | **Update Question** | **Update Successfully** | **Update Successfully** | **Pass** |  |
| **3.** | **Delete Question** | **Delete Successfully** | **Delete Successfully** | **Pass** |  |
| **4.** | **User Info** | **User result**  **, info show** | **User result**  **, info show** | **Pass** |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test Suit Id – 1**  **Test Id - 2** | | | **Description :- This Test case on User Module to check validation on user and store data** | | |
| **No.** | **Task** | **Expected Output** | **Actual Result** | **Pass / Fail** | **Remark** |
| **1.** | **Registration User with Validation** | **Register Successfully** | **Register Successfully** | **Pass** |  |
| **2.** | **Quiz Exam Page** | **Exam Complete**  **Successfully** | **Exam Complete Successfully** | **Pass** |  |
| **3.** | **Feedback Form** | **Thanks for your Feedback** | **Thanks for your Feedback** | **Pass** |  |

Limitation

1. Time: The project’s completion, or final due date for deliverables. Time constraints can be negotiated but can never be overcome completely.
2. Cost: The budget of the project delineates how much can be spent on certain things. Maybe everyone in the office would work better with their own helicopter or personal chef, but successful project management can’t be about having the best of everything. Instead, it’s about doing the best you can with what you have.
3. Scope: What is expected of the project, as outlined in the project plan. This can be somewhat negotiated, but in the end, if you’ve agreed to do something, you’re expected to get it done.
4. Quality: These are the limitations placed on the project deliverable by the client. For example, if they specifically want a landing page to have a load speed of under two seconds, then not achieving that will be a failure.
5. Benefits: This constraint relates to the expected outcomes or *benefits* from a project. Think of a new advertising campaign that’s actually turning people off your product in droves, though the expected benefits were *more*customers.
6. Risk: The risk tolerance of the project usually can’t be overcome. The risks that are identified and ranked on the project’s risk register therefore need to be constantly monitored to ensure they don’t exceed the risk tolerance threshold of the project’s stakeholders.

Bibliography

There are some study materials which is used by me to develop my software system,

Which are as follows :-

**Books :-**

- Pl / Sql (By Ivan N. Bayross),BPB Publication, New Delhi

- ntroduction to System Analysis and Design, Third Edition, Prentice-Hall of India Pvt, NewDelhi

- Java the complete reference, eleventh edition, Mc Graw Hill

**Reference Site :-**

- www.tutorialspoint.com

- www.w3schools.com

- www.javatpoint.com

- [www.geeksforgeeks.com](http://www.geeksforgeeks.com)

=> END OF PROJECT REPORT <=