

**Project Report**

**on**

Quiz Manager Application

Submitted by

AmirAli MASHAYEKHI

(amirali313@gamil.com)

Rinku Maria Thomas

(rinkumaria1894@gmail.com)

Contents

1.Subject Description 3

[1.1]Overview: 3

[1.2]BackGround: 3

[1.3] Motivation 3

[1.4] Objective 3

[1.5] Development Environment 3

2.Subject analysis 4

[2.1] Features 4

[2.2] Application Feasibility 4

[2.3] Data Description 4

[2.4] Expected Results 5

[2.5] Scope and limitations 6

3.Conception 7

[3.1] Data Structures 7

[3.1] Global Application Flow 8

4.Console operation Description 9

4.Configuration 10

References 11

# 1.SUBJECT DESCRIPTION

### **[1.1]Overview**:

This report contains all the information about the Quiz Manager Java fundamental project.

[1.2] Background**:**

The goal of this project is to make a fully functional Quiz Manager application using JAVA.

### [1.3] Motivation:

This report and its subject, namely Quiz Manager, has been greatly influenced by our will to learn the concepts of the programming language **JAVA**.

What we encountered is that

* Java is Easy to write and more readable and eye catching.
* Most of the concepts like OOPs are drawn from C++ thus making Java learning simpler.

### [1.4] Objective

The main objective of this project is to run a Quiz Manager application which can fetch a quiz(set of questions) for a student and grade it. Admin user(teacher) can do CRUD(Create, Read, Update, Delete) manipulations on the questions.

### [1.5] Development Environment

**PLATFORM USED :** MacOS Mojave

**LANGUAGE USED :** Java SE 11.02

**IDE :** IntelliJ Idea

**DATABASE :** h2 SQL database

# 2.SUBJECT ANALYSIS

## [2.1] Features

* Platform Independent
* Easy to use
* Robust
* User friendly
* Facile to modify
* Clean separation of classes

## [2.2] Application Feasibility

* This current application is a prototype of a system that can be employed for a quiz manager system for testing the working of the quiz
* Most of the components used such as the development platform, servers, and databases are open source.

## [2.3] Data Description

The data description and data access objects are clearly specified below.

The Schema for **QUESTION** table is **<ID, QUESTION, DIFFICULTY, TOPICID, CHOICE1, CHOICE2, CHOICE3, CHOICE4, ANSWER>**

**ID :** INT (PRIMARY KEY, AUTO\_INCREMENT)

**QUESTION :** VARCHAR(255)(UNIQUE)

**DIFFICULTY :** INT

**TOPICID :** INT (FOREIGN KEY REFERENCES TOPICS(ID))

**CHOICE[1-4] :** VARCHAR(255)

**ANSWER :** INT

The Schema for **TOPICS** table is **<ID, TOPIC1, TOPIC2>**

**ID :** INT(PRIMARY KEY, AUTO\_INCREMENT)

**TOPIC1, TOPIC2** **:** VARCHAR(255)

**DAO**

1. **QuestionDAO**

* **create(question)  :** This takes a new question data from teacher and stores it in the database.
* **update(question)  :** This takes a question data from teacher and updates the question in the database.
* **delete(question)  :** This takes a question id from teacher and deletes that question from the database.
* **showAllQuestions()  :** This displays all the questions to the user.
* **getQuestions(topics, difficulty) :** This displays a set of questions to the student based of the topics selected and difficulty.

1. **MCQAnswerDAO**

* **create(question)  :** This takes the answer data from teacher and stores it in the database.
* **update(question)  :** This takes the answer data from teacher and updates the question in the database.

## [2.4] Expected Results

## The end result of the application can be looked as a highly sophisticated, user friendly and secure tool created for Quiz Management.

## This tool is capable of authenticating the student and teacher, display a quiz based on some criteria, create a new question, display all questions, updating an existing question, and deleting a question from the database.

## The tool needs to communicate with the database and return with the results in quick time.

## [2.5] Scope and limitations

**Scope:**

* Can be used in schools and universities to increase the examination and grading efficiency.
* Will help reduce the paper waste.
* Once setup can be used for many times and can be updated when needed.

**Limitations:**

* Need Requested Software requirement
* Lack of Exe file

# 

**3. CONCEPTION**

**[3.1] Data Structures**

The data structures provided by the Java utility package are very powerful and perform a wide range of functions. These data structures consist of the following interface and classes : Enumeration, BitSet, Vector, Stack, Dictionary, Hashtable, Properties.

However the data structure used in this project is **LIST** ,which is a collection of elements .List is used in the project to store the collection of identities.

**String** is also a widely used data structure in this project

**[3.3] Global Application Flow**

YES

NO

Student

Teacher

NO

**CONTINUE?**

**DB OPERATION**

**Delete**

**Update**

**Create**

**getQuestions**

**ShowAll**

**AUTHENTICATE**

**MENU**

# 3.CONSOLE OPERATIONS DESCRIPTIONS

Console Operations Implemented in this Systems are

Each Operations are explained below

* **loginStudent(), authenticateStudent():**Authenticates student login.
* **loginTeacher(), authenticateTeacher():**Authenticates teacher login.
* **handleStudentQuiz:**

This console operation will take topics and difficulty from the student.

* **takeQuiz():**

This console operation will display set of questions based on topics and difficulty. It was also calculate grade for MCQ Questions.

* **handleTeacherRequest():**

## This console operations will display a menu to do CRUD operations to the teacher.

**5. CONFIGURATION**

* url : jdbc:h2:~/test
* username : sa
* password :
* Database : h2

We have used a global configuration which reads these properties from a file in the file system. We have used a standard format called **properties** for this purpose.

# References

* <https://thomas-broussard.fr/>
* <https://www.tutorialspoint.com/java/java_data_structures.htm>
* <https://www.tutorialspoint.com/>
* [https:](http://www.srikanthtechnologies.com/asrikanth.aspx)//www.w3school.com/
* [https:](http://www.srikanthtechnologies.com/asrikanth.aspx)//www.javatpoint.com/