QUIZ MANAGER

PRIYAM CHADHA & SWETALI NIMJE

INTRODUCTION

This document describes the technical specification of the project "Quiz Manager" developed using JAVA for java programming course at EPITA. It describes each application's architecture and sub-architecture their associated interfaces, database schemas, and the motivations behind the chosen design.

- Language: JAVA, SQL

- Tools:

    - Eclipse: Eclipse is a popular IDE used in Java programming.

    - H2: H2 is open-source java database used in the project

Features of the project

Quiz Manager System in in two parts based on the user of the system:

- Student

- Teacher

STUDENT:

- Register

- They can choose the quiz they want to take and attempt the quiz of a mix of MCQ and Open questions.

- The Quiz will be evaluated and grade will be displayed on the console.

TEACHER:

- Login

- CRUD Operations on Open Questions and MCQ Questions.

- Create Quiz

- Export Quiz as text file

Database Model

We have packages with services, launcher and dataModel. We are using H2 for connecting database. All the SQL queries are stored in EPITA_Java_quiz_master/sql/* directory.

DataModel	Attribute	DB structure
Student	Name, ID	- ID - NAME - GRADE - QUIZ
Question, MCQQuestion	ID, Questions, Topics, Difficulty	- ID - LABEL - DIFFICULTY - TOPIC
Answer	Question, Text	- ID - LABEL - QUESTION ID
Quiz	listMCQQuestion, listQuestion, title	- TITLE - OPENQUESTIONS

DataModel	Attribute	DB structure
		• MCQQUESTIONS
MCQCHOICE	ID, label, Question, valid	▪ ID ▪ LABEL ▪ VALID ▪ MCQQUESTION

Table1. DAO Datamodel

DATABASE QUESRIES TO CREATE TABLES :

- CREATE TABLE **ANSWER** (ID INT auto_increment PRIMARY KEY, LABEL VARCHAR(255) , QUESTIONID INT);

- CREATE TABLE **MCQANSWER** (CHOICEID INT auto_increment PRIMARY KEY, STUDENTID VARCHAR(255));

- CREATE TABLE **MCQCHOICE** (ID INT auto_increment PRIMARY KEY, LABEL VARCHAR(255), VALID BOOLEAN, MCQQUESTIONID INT);

- CREATE TABLE **MCQQUESTION** (ID INT auto_increment PRIMARY KEY, LABEL VARCHAR(255), DIFFICULTY INT, TOPICS VARCHAR(255));

- CREATE TABLE **QUESTIONS** (ID INT auto_increment PRIMARY KEY, QUESTION VARCHAR(255), DIFFICULTY INT, TOPICS VARCHAR(255));

- CREATE TABLE **STUDENT** (ID INT auto_increment PRIMARY KEY, NAME VARCHAR(255), GRADE VARCHAR(255), QUIZ VARCHAR(255));

- CREATE TABLE **QUIZ** (TITLE VARCHAR(255) PRIMARY KEY, OPENQUESTIONS VARCHAR(255), MCQQUESTIONS VARCHAR(255));

Control flow of the system:

QuizManager Class

STUDENT

main() — TEACHER → login() ↔ authenticate()

login() → menu()

Create → createOpenQuestion()

createMCQQuestion()

Update → updateOpenQuestion ()

updateMCQQuestion()

Delete → deleteOpenQuestion()

deleteMCQQuestion()

createQuiz() → createTextFile()

menu() → topicSearchOpenQuestion(), topicSearchMCQQuestion(), allOpenQuestion(), allMCQQustion()

Search

From Create, Update, Search, Delete Functions

StudentManager Class

studentmenu()

launchQuiz()

calcGrade()

JDBCDAO Classes

create()   update()   Search()

delete()   getConnection()

DATABASE

4