**IGIRANEZA MARIE EMMANUELLA**

**2401001756**

**DATABASE MANAGEMENT SYSTEM**

**LECTURER: BUGINGO EMMANUEL**

**UNIVERSITY OF KIGALI MUSANZE CAMPUS**

**FACULTY : BBIT**

**29 JAN 2025**

**PROJECT: ONLINE QUIZ SYSTEM**

**Online Quiz System (OQS), is a web–based quiz system that can be used by lecturers to evaluate students effectively, efficiently and perfectly.**

**Description of the ONLINE QUIZ SYSTEM**

In the context of an Online Quizzes database, the system may have the following characteristics:

* **Purpose:** Manage and store information related to online quizzes, including questions, answers, quizzes, participants, and their results.
* **Modules:**
  + **Quizzes:** Each quiz may have multiple questions with correct answers.
  + **Questions:** Each question belongs to a quiz and has one or more possible answers.
  + **Users:** Students or participants who take quizzes.
  + **Results:** Track users' scores and quiz completion.

Entity diagram

|  |
| --- |
| **User** |
| * user\_id(fk) * dusername * password email |

|  |
| --- |
| **Answer** |
| answer\_id(fk)  question\_id  answer\_text |

|  |
| --- |
| **Question** |
| * question\_id(fk) quiz\_id, question\_text |

|  |
| --- |
| **Quiz** |
| quiz\_id(fk)  title description |

|  |
| --- |
| **Result** |
| result\_id(fk) user\_id  quiz\_id  score |

 **User** (1) -> (M) **Result**

 **Quiz** (1) -> (M) **Question**

 **Question** (1) -> (M) **Answer**

 **User** (M) -> (1) **Result** -> (1) **Quiz**

For the Online Quizzes database, the ERD would involve the following entities:

* **User**: Stores user details (id, username, password, etc.)
* **Quiz**: Represents the quizzes available (id, title, description, etc.)
* **Question** : Contains who
* **Answer**: Stores multiple possible answers for each question (id, answer\_text, is\_correct, question\_id)
* **Result**: Stores users' results (id, user\_id, quiz\_id, score)

Relationships:

* A **Quiz** has many **Questions**.
* A **Question** has many **Answers**.
* A **User** can take many **Quizzes**, and each quiz will have a **Result** for the user.
* **3 Logical Data Model (LDM)**

A logical data model is a more abstract representation that defines the data entities, relationships, and constraints without focusing on how the data will be physically stored.

Here’s a brief outline of the logical schema:

* **User** (user\_id, username, password, email)
* **Quiz** (quiz\_id, title, description)
* **Question** (question\_id, quiz\_id, question\_text)
* **Answer** (answer\_id, question\_id, answer\_text, is\_correct)
* **Result** (result\_id, user\_id, quiz\_id, score)

Relationships:

* Quiz has a one-to-many relationship with Question.
* Question has a one-to-many relationship with Answer.
* User has a one-to-many relationship with Result.
* Quiz has a one-to-many relationship with Result.
* **4. Physical Data Model (PDM)**
* The physical data model takes the logical design and implements it in terms of physical storage, including data types, indexes, etc.

Here's an example of the tables:

CREATE TABLE User (

user\_id INT PRIMARY KEY AUTO\_INCREMENT,

username

VARCHAR(50) NOT NULL,

password

VARCHAR(255) NOT NULL,

email

VARCHAR(100);

CREATE TABLE Quiz (

quiz\_id INT PRIMARY KEY AUTO\_INCREMENT,

title

VARCHAR(255) NOT NULL,

description TEXT

CREATE TABLE Question (

question\_id

INT PRIMARY KEY AUTO\_INCREMENT,

quiz\_id INT,

question\_text TEXT,

FOREIGN KEY (quiz\_id) REFERENCES Quiz(quiz\_id)

);

CREATE TABLE Answer (

answer\_id

INT PRIMARY KEY AUTO\_INCREMENT,

question\_id

INT,

answer\_text TEXT,

BOOLEAN,

FOREIGN KEY (question\_id) REFERENCES Question(question\_id)

);

CREATE TABLE Result (

result\_id

INT PRIMARY KEY AUTO\_INCREMENT,

user\_id

INT,

quiz\_id INT,

score INT,

FOREIGN KEY (user\_id) REFERENCES User(user\_id),

FOREIGN KEY (quiz\_id) REFERENCES Quiz(quiz\_id)

);

**5. Data Dictionary**

A data dictionary defines each element in the database schema.

**Tables and Columns:**

* **User**
  + user\_id: Integer, primary key, auto-increment.
  + username: Varchar(50), unique, not null.
  + password: Varchar(255), not null.
  + email: Varchar(100).
* **Quiz**
  + quiz\_id: Integer, primary key, auto-increment.
  + title: In
  + description: Text.
* **Question**
  + question\_id: Integer,
  + quiz\_id: Integer
  + question\_text: Text, not null.
* **Answer**
  + answer\_id:
  + question\_id:
  + answer\_text: Text
  + is\_correct:
* **Result**
  + result\_id:
  + user\_id: Integer, foreign key to User.
  + quiz\_id: Integer,
  + score: Integer, stores the score achieved by the user.