**JOSEPH COURBON**

**24245**

**Project Title: Online Exam Management System**

**4. DATABASE SCHEMA:**

In the Online Exam Management System, the database is structured with several tables, each serving a specific purpose. Here's a summary of the relationships between the tables:

1. User Table:

The central table storing information about system users.

* Primary Key: UserID
* Relationships:
* Foreign Key in Result Table (UserID): Relates a user to their exam results.
* Foreign Key in User\_Role Table (UserID): Associates a user with their assigned roles.

1. User\_Role Table:

Manages the roles assigned to users in the system.

* Primary Key: User\_RoleID
* Relationships:
* Foreign Key in User Table (UserID): Establishes a link between users and their assigned roles.
* Foreign Key in Role Table (RoleID): Maps user roles to predefined roles.

1. Role Table:

Defines the roles available in the system ( student, administrator).

* Primary Key: RoleID
* Relationships:

Foreign Key in User\_Role Table (RoleID): Associates roles with users.

1. Result Table:

Stores information about exam results for each user.

* Primary Key: ResultID
* Relationships:
* Foreign Key in User Table (UserID): Connects results to specific users.
* Foreign Key in Quiz Table (QuizID): Links results to a particular exam.

1. Quiz Table:

Represents the exams or quizzes created within the system.

* Primary Key: QuizID
* Relationships:
* Foreign Key in Result Table (QuizID): Connects results to a specific exam.
* Foreign Key in Category Table (CategoryID): Categorizes exams based on subject or topic.

1. Category Table:

Organizes quizzes into categories for easy navigation.

* Primary Key: CategoryID
* Relationships:

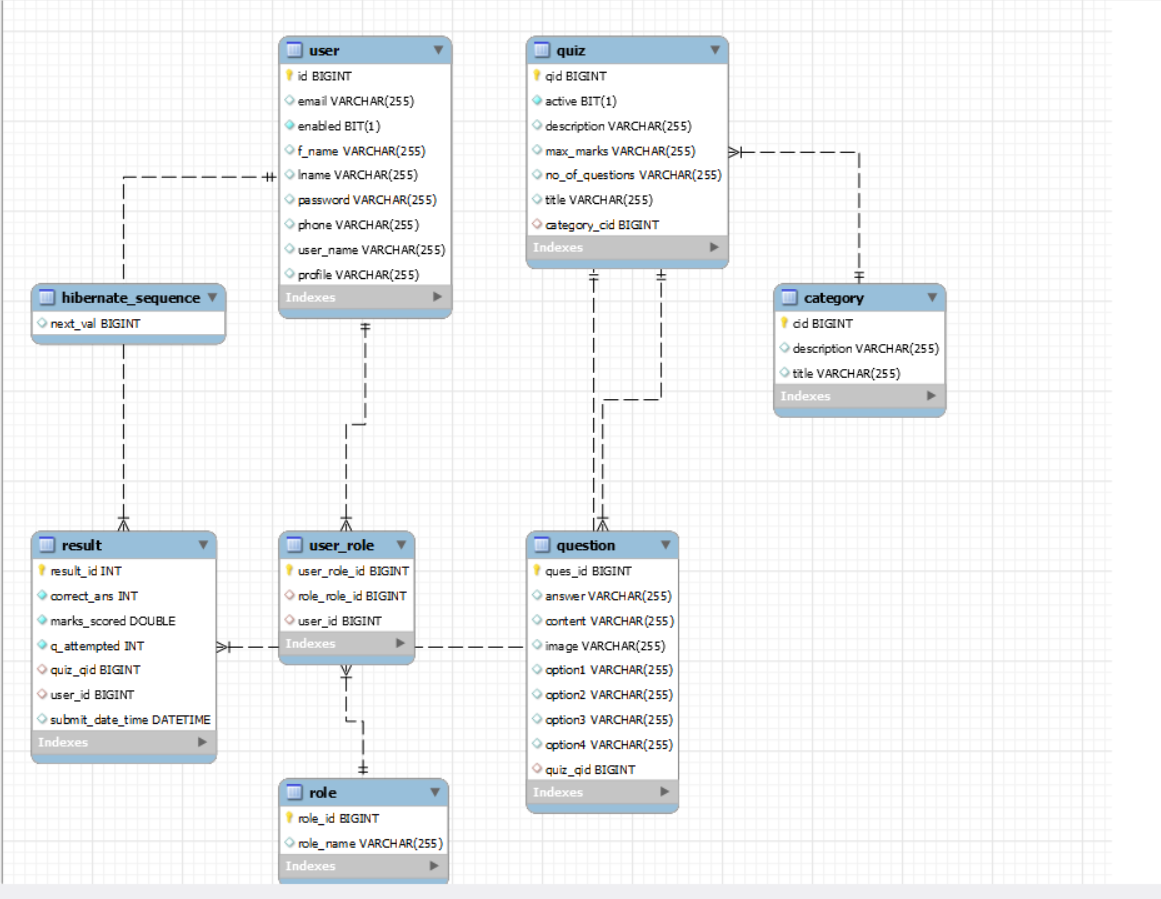
Foreign Key in Quiz Table (CategoryID): Associates quizzes with specific categories.

1. Question Table:

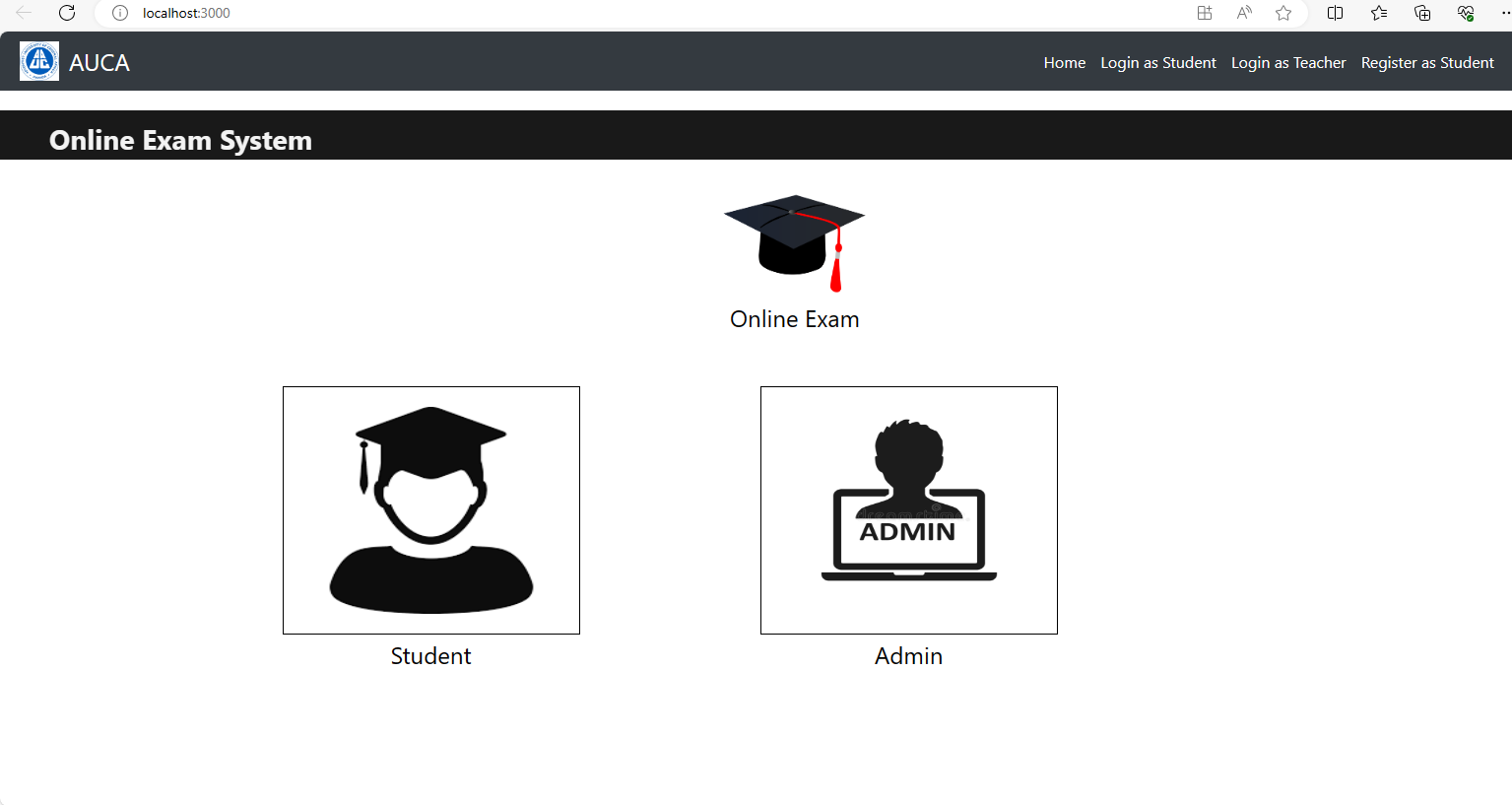
Stores details about the questions used in quizzes.

* Primary Key: QuestionID
* Relationships:

Foreign Key in Quiz Table (QuizID): Relates questions to a specific quiz.



**5. USER DOCUMENTATION**



To access the application, visit [localhost 3000] and log in using the provided credentials or complete the registration process.

Upon logging in, users will be greeted by a comprehensive dashboard displaying key information such as upcoming exams, recent results, and important announcements.

When ready to take an exam, simply navigate to the "Exams" section, choose the relevant quiz, and follow the on-screen instructions for answering questions. Remember to save or submit responses as guided by the system.

To review completed exams, visit the "Results" section where detailed information, including scores and feedback, can be accessed.

Manage your profile effectively by updating personal information, including password changes, in the dedicated "Profile" or "Account" section.

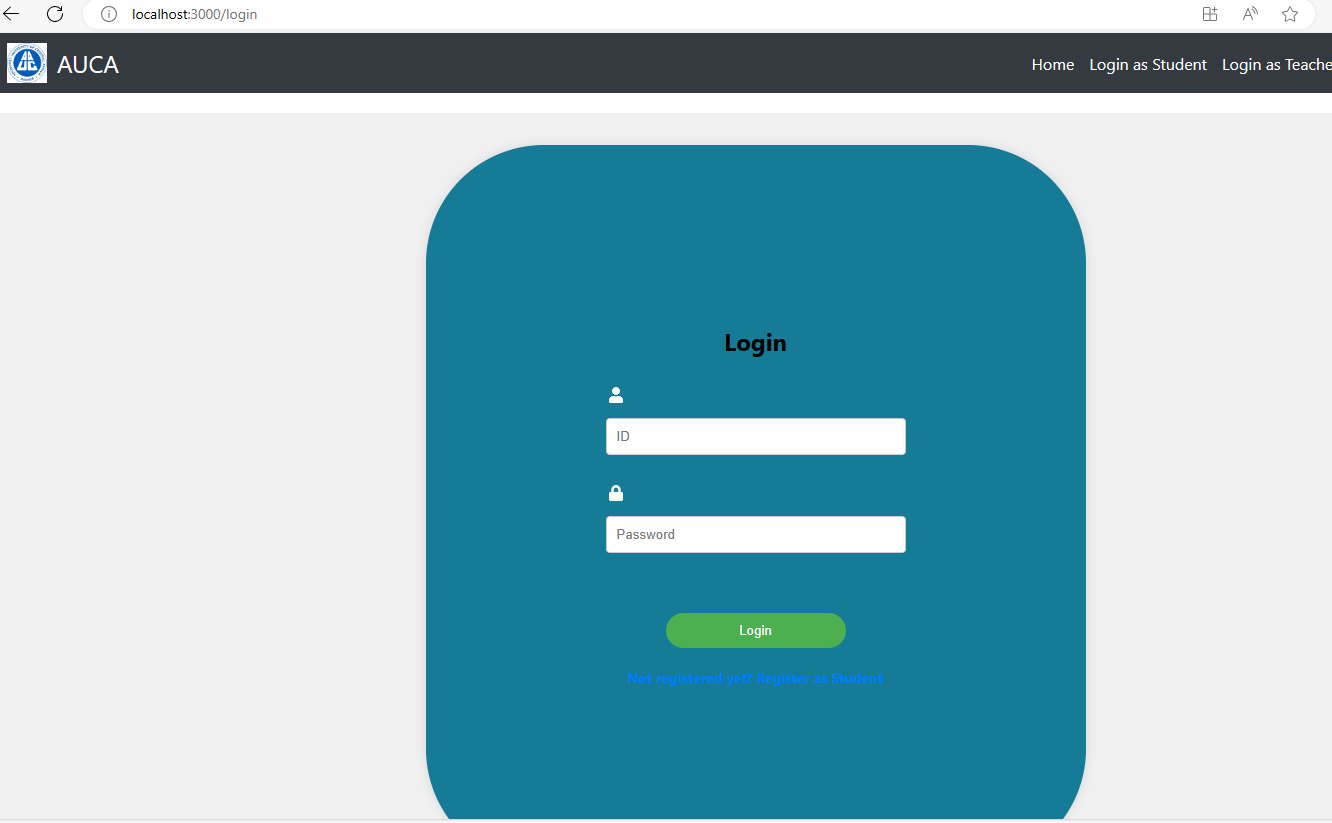
Explore exams organized by subject or topic effortlessly in the "Categories" section, allowing you to choose from a selection of available quizzes.

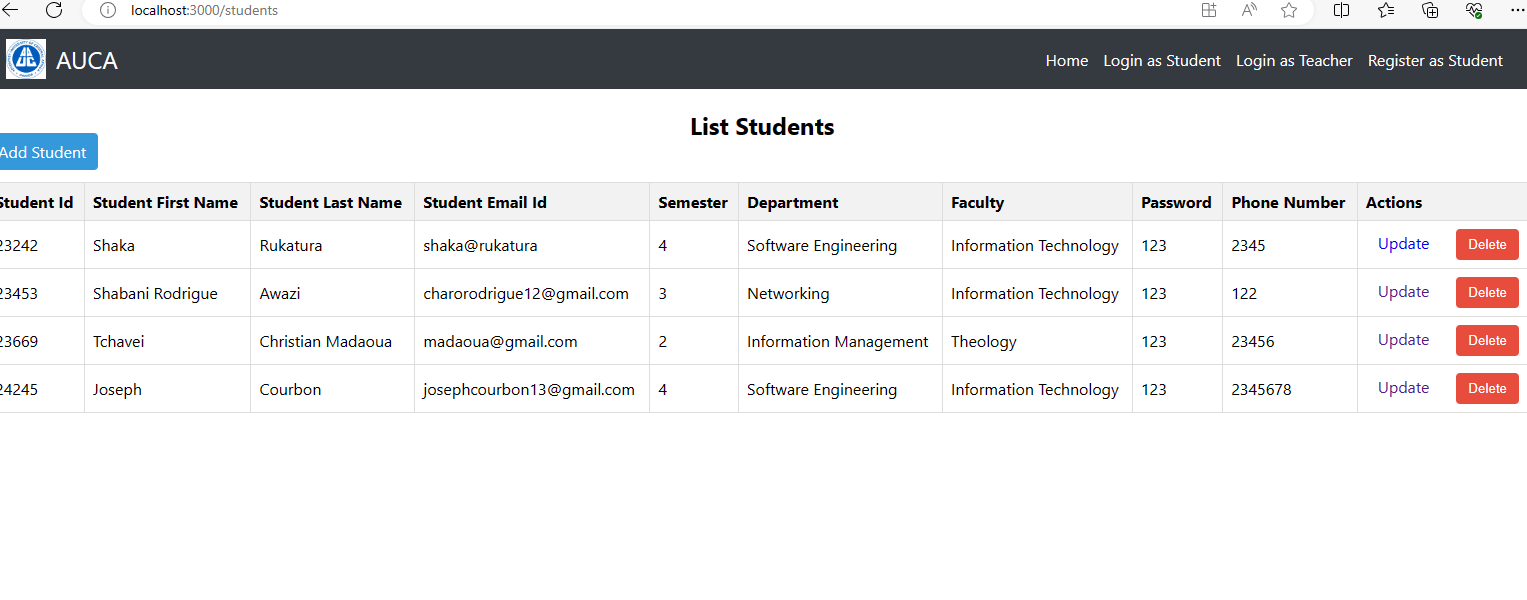
Stay informed about important events such as upcoming exams, announcements, and result notifications through the system's notification feature.

User roles, such as student, instructor, or administrator, come with varying permissions. Access role-specific features based on your assigned privileges.

For additional assistance or information, find detailed guides or FAQs in the "Help" or "Support" section. If further help is needed, reach out to support through the provided contact details.

Always remember to log out after completing tasks by using the "Logout" button located in the top-right corner.





**6. TECHNICAL DOCUMENTATION :**

The Online Exam Management System employs a three-tier architecture, with a frontend developed using React and a backend implemented in Spring Boot MVC. The system relies on a MySQL database for data storage.

**Frontend:**

Framework: React

Languages: JavaScript, CSS

Key Features: Responsive UI, dynamic content rendering, and seamless user interactions.

**Backend**:

Framework: Spring Boot MVC

Languages: Java

Components: RESTful API for communication between frontend and backend.

Security Measures: Implementing authentication and authorization using Spring Security.

**Database:**

Database Management System: MySQL

Schema: Structured to store user data, exam details, and results.

ORM: Hibernate for efficient database interaction.