ONLINE QUIZ APPLICATION

ABSTRACT:

The Online Quiz Application is a platform designed to enable users to take quizzes on various topics. It features user authentication, quiz creation, and scoring functionality. The backend serves as the backbone of the application, providing secure and efficient handling of data,

including user accounts, quizzes, and results. By utilizing Java and JDBC, the application ensures a reliable connection to the database while incorporating best practices in data handling and security.

INTRODUCTION:

The Online Quiz Application simplifies the process of conducting quizzes in an interactive and user-friendly environment. The backend architecture is designed to manage user accounts, store quiz data, and calculate scores efficiently. It enables administrators to create quizzes and users to participate, track their progress, and view their performance history. Additionally, robust security measures such as password hashing and input validation ensure data integrity and protection against unauthorized access. This documentation outlines the backend implementation, emphasizing its modular design and scalability.

TECHNOLOGIES USED:

1. **Programming Language**: Java

- Used for implementing core backend logic and managing data flow.

2. **Database**: MY SQL Workbench

- A lightweight database solution for storing user accounts, quizzes, questions, and results.

3. **Database Interaction**: JDBC

- Facilitates CRUD operations to interact seamlessly with the database.

4. **Password Security**: SHA-256

- Ensures secure storage of user passwords through hashing.

6. **Development Tools**:

- Integrated Development Environment (IDE): Eclipse

CONCLUSION:

The backend of the Online Quiz Application is a robust and secure system that effectively manages user authentication, quiz data, and results. By adhering to modular design principles, the backend can be easily extended to support additional features like a leaderboard or API integration. Security considerations, such as password hashing and SQL injection prevention, ensure data safety, making the application reliable and user-friendly.