

Online Quiz Portal Using REST APIs

Write up: -

Specifications: -

- Includes a RESTful Web API to perform CRUD operations on Domain objects as per requirement using Spring Boot and MySQL/Oracle database.
- The admin user has a separate API to access the admin portal, which requires authentication with the admin username and password.
- The admin user can update the profile details and change the password after using the access token generated by login.
- The admin user can add questions using the **addQuestions** API.
- The admin user can create a quiz by entering **quizid** and selecting questions with **questionid**.
- The admin user can obtain statistics on total quizzes, questions, and users by using their APIs.
- The admin user can find the users who participated in the quiz along with scores and standings.
- Users can explore various quizzes created by the admin.
- Users can create their profile using new user registration.
- Users can take the quiz and try to answer the questions.
- Users can also check if the given answers are correct, as well as their results and positions.

Online Quiz Portal Using REST APIs

Algorithm

- Create the necessary domain objects (e.g., User, Quiz, Question) with appropriate fields and relationships.
- Implement the CRUD operations for the domain objects using Spring Boot and MySQL/Oracle database. Use the appropriate annotations like **@RestController**, **@RequestMapping**, **@GetMapping**, **@PostMapping**, **@PutMapping**, **@DeleteMapping**, etc., to define the RESTful APIs for performing these operations.
- Implement the admin portal API for authentication. This API should accept the admin username and password and generate an access token upon successful authentication. You can use techniques like JSON Web Tokens (JWT) for generating and validating the access token.
- Implement the API for updating the admin user's profile details and changing the password. This API should require the access token generated during login for authentication and authorization.
- Implement the API for adding questions. This API should only be accessible to the admin user with a valid access token.
- Implement the API for creating a quiz. The admin user should provide a quiz ID and select questions using their question IDs. Store the quiz details in the database and associate the selected questions with the quiz.
- Implement the APIs for obtaining statistics on total quizzes, questions, and users. These APIs should only be accessible to the admin user with a valid access token.

Online Quiz Portal Using REST APIs

- Implement the API for retrieving the list of users who participated in a quiz along with their scores and standings. This API should only be accessible to the admin user with a valid access token.
- Implement the API for users to explore various quizzes created by the admin. This API should return a list of available quizzes that users can choose from.
- Implement the API for user registration. This API should create a new user account and generate an access token for authentication.
- Implement the API for user login. This API should accept the access token generated during registration and return a new access token for subsequent API calls.
- Implement the API for users to take a quiz. This API should accept the quiz ID and the user's answers to the questions. It should calculate the score and store the user's results in the database.
- Implement the API for users to check their quiz results. This API should return the user's score, the correct answers, and their position compared to other users.
- Test the APIs using tools like Postman to ensure they are functioning as expected.
- Deploy the application on a Tomcat server and ensure it is connected to the MySQL/Oracle database.
- Perform additional testing to validate the overall functionality and resolve any issues or bugs.

Online Quiz Portal Using REST APIs

GitHub link:-

https://github.com/bspsripathi/OnlineQuiz_Using_REST.git