Simple Spring Boot Quiz Application

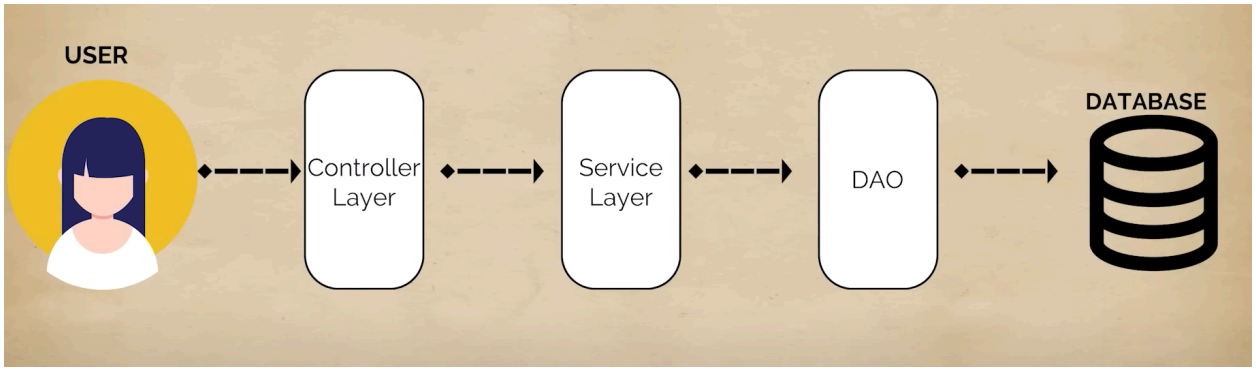
Steps:

1. Create a spring boot project with Maven in **start.spring.io**

   Required Dependencies:
   1.Spring Web
   2. Postgresql driver
   3. Spring Data JPA
   4.Lombok

2. Configure the database connection in **QuizApp/src/mainresources/application.properties**
   Provide the SQL driver class, database name, username, password.

3. To start building the project with Spring Boot you need to know about the process behind the scene.



In total we have 3 layers between user and the database,

- The first layer is the control layer, where users submit the request. The controller can only accept the request.
- The second layer is the service layer. It processes the request which is coming from the control layer and performs some business logic here.
- The third layer is DAO which is responsible for fetching data from the database
  .

4. Next, you need to know about Spring MVC, i.e., Model View Controller.Spring MVC (Model-View-Controller) is a framework within the Spring Framework that is used for building web applications. It follows the MVC design pattern, which separates the application into three main components: Model, View, and Controller. This separation helps in developing loosely coupled, maintainable, and testable web applications.
5. Before building the project, create a table with some set of questions in your database, whether it is Postgres, MySQL or Oracle. (I used postgresql).

6. Create an annotated entity class in src/main/java/com.example.<name>.model. This class represents your table in the database.
7. Declare the variables with appropriate data types which represent columns of your database.
8. Mention the required annotations.
9. After creating the Model you need the controller which accepts the user requests, so you need the controller package.
10. In the controller package create the QuestionController class with **@RESTCONROLLER** annotation.

    What is a Rest Controller ?

    A **@RestController** in Spring is a specialized version of the **@Controller** annotation. It's used to create RESTful web services in a Spring application. When you annotate a class with **@RestController**, you're telling Spring that the class will handle web requests and that its methods will return data directly in the form of JSON or XML, rather than rendering a view (like a JSP page).

11. Create the methods for accepting the CRUD operation requests from users. CRUD(Create Read Update Delete).
12. Now after getting the user request, you actually need a directory or package which will process the user requests. So, this is where the service layer comes into picture.
13. Create a service directory which stores all the service class files.
14. Now the service class has a responsibility to process the user requests. It will process all the requests of the user with the help of the DAO layer.
15. DAO(Data Access Object) is the layer which is used to fetch data from the database.
16. By using all these steps you can build a simple Quiz web application.

Sample Screenshots & WorkFlow

1. Fetch all questions from database.

localhost:8081/question/allQuestions


```
[
  {
    "id": 1,
    "question_title": "What is a class in Java?",
    "option1": "A function",
    "option2": "An object",
    "option3": "A data structure",
    "option4": "A loop",
    "right_answer": "An object",
    "difficulty_level": "Easy",
    "category": "java"
  },
  {
    "id": 3,
    "question_title": "What is a list in Python?",
    "option1": "A type of loop",
    "option2": "A built-in function",
    "option3": "A data structure",
    "option4": "An object",
    "right_answer": "A data structure",
    "difficulty_level": "Easy",
    "category": "python"
  },
  {
    "id": 4,
    "question_title": "Which data structure uses First-In-First-Out (FIFO) order?",
    "option1": "Stack",
    "option2": "Queue",
    "option3": "Array",
    "option4": "LinkedList",
    "right_answer": "Queue",
    "difficulty_level": "Medium",
    "category": "python"
  },
```

2. It will give the questions based on the category or difficulty level,since I have created the quiz application on java and python.

localhost:8081/question/category/java


```
[
  {
    "id": 1,
    "question_title": "What is a class in Java?",
    "option1": "A function",
    "option2": "An object",
    "option3": "A data structure",
    "option4": "A loop",
    "right_answer": "An object",
    "difficulty_level": "Easy",
    "category": "java"
  },
  {
    "id": 5,
    "question_title": "What is a constructor?",
    "option1": "A member of a class",
    "option2": "A loop in Python",
    "option3": "A data type",
    "option4": "A special method",
    "right_answer": "A special method",
    "difficulty_level": "Medium",
    "category": "java"
  },
  {
    "id": 7,
    "question_title": "In Java, what is used to create an instance of a class?",
    "option1": "Class",
    "option2": "Method",
    "option3": "Object",
    "option4": "Constructor",
    "right_answer": "Constructor",
    "difficulty_level": "Easy",
    "category": "java"
  },
```


3.You can also add or delete questions using PostMapping or DeleteMapping.
4. Next I created the QuizController class which allows you to create a quiz with parameters like category,number of questions,and title of the quiz.
5. This can be done using ManyToMany relation.
6. This application can also allow you to take a quiz and calculate your quiz score.

Note: This is only a web application, it doesn't have any UI.

I built this project for learning and understanding purposes with the help of Navin sir, he's my tutor. I've been following his content for the past 2 years.