

Full Stack Question App Project

[Repository Link to the project](#)

Introduction:

I developed this project to conduct MCQ-type question exams. The main idea of this project is to host it on localhost. Attendees should join the same localhost network to easily access the question app. No network connection is needed, so the advantage is no network problems, and this can easily handle traffic.

Overview:

The MCQ Question App is a web-based application designed to facilitate multiple-choice question exams in a localized environment. The primary objective of this project is to offer a straightforward and efficient solution for conducting exams without the need for an internet connection, thus eliminating potential network-related issues.

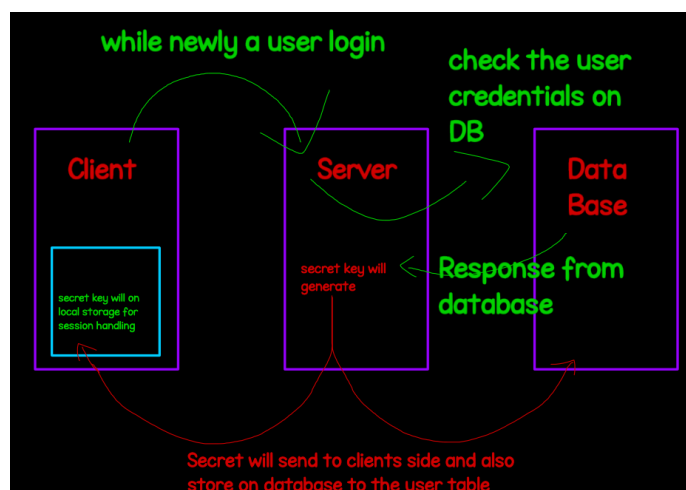
Key Features:

- **Local Hosting**
- **Session Handling using a secret key**
- **Update the answer in the database for every change made to an answer**
- **Previously stored answers will automatically load when a user refreshes or relogs in.**

Technologies Used:

- **Frontend:** using the React framework, providing a dynamic and responsive user interface for seamless interaction.
- **Backend:** Powered by Node.js and Express.js, facilitating the server-side logic and data management.
- **Database:** Utilizes a database system MySQL to store exam questions, user information, answer for the questions and exam results securely.

How Session Handling Works:



When a new user logs in, the credentials are sent to the server, which verifies them by comparing them with the database. If the credentials are correct, the server creates a secret key containing 16 characters. This secret key is then sent to both the client-side and the database. The secret key is stored in the corresponding user table in the database and also stored in the client-side local storage. Subsequently, every time the page is refreshed or any server-side operation is performed, data communication occurs with the help of the secret key.

Advantages:

- **Enhanced Security**
- **Improved User Experience**
- **Scalability**

Usage:

To use the MCQ Question App, users simply need to set up the application on a local server within their network. Participants can then access the app via their web browsers and join the exam session using the provided credentials.

API Documentation:

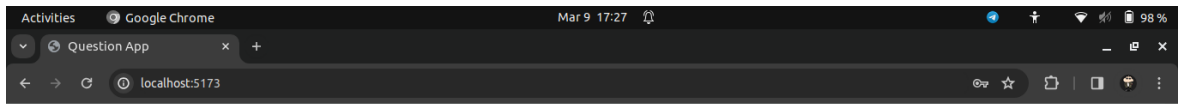
Routes:

- METHOD:GET -> serverAddress/answers : Responsible for sending stored answer as JS array format or JSON
- METHOD:POST -> serverAddress/answers : Responsible to storing the updated answer from client to database
- METHOD:GET -> serverAddress/questions : Responsible for sending stored questions as JS array format or JSON
- METHOD:POST -> serverAddress/auth : This mehtod is responsible for verifying the user-entered credentials with the database and performing login operations

Sample Images :

```
mysql> show tables;
+-----+
| Tables_in_question_app |
+-----+
| answers                 |
| questions               |
| users                   |
+-----+
```

```
mysql> select * from questions;
+-----+-----+-----+-----+
| question | options | question_no | question_type |
+-----+-----+-----+-----+
| What is transistor | ["Food","Computer","Electronic Component","Animal"] | 1 | text |
| What is Binary | ["Insect","Country","Maths Problem","Computer Number System"] | 2 | text |
| Calculate the expression 2+2 | ["9", "2/4+7-9", "((8*8-50)/2-6+1)*2", "-4-(-4*-1)"] | 3 | text |
| Who is the Humans ? | ["ChatGPT", "Gemini", "Tamil", "LED light"] | 4 | text |
| Python if statement syntax ? | ["include <IF> 'Condition${}' ","if condition :", "_IF_ ${condition} : reusult","py.if.condition & \"statement\""] | 5 | text |
+-----+-----+-----+-----+
```



1. What is transistor

- ☒ Food
- ☐ Computer
- ☐ Electronic Component
- ☐ Animal

2. What is Binary

- ☒ Insect
- ☐ Country
- ☐ Maths Problem
- ☐ Computer Number System

3. Calculate the expression 2+2

- ☐ 9
- ☐ 2/4*7-9
- ☒ ((8*8-50)/2-6+1)*2
- ☐ -4-(-4*-1)

4. Who is the Humans ?

- ☒ ChatGPT
- ☐ Gemini
- ☐ LED light

