LAPORAN PERANCANGAN REST API QUIZ

**TEMA :**

**QUESLESS (QUESTION LESSONS)**



Disusun Oleh :

**KELOMPOK 2 PEMROGRAMAN BERBASIS PLATFORM**

Dosen Pengajar : **Alun Sujjada, S.Kom, M.T**

Widy Putri Pratiwi ( 20230040027 )

Abdul Azis Burhanudin ( 20230040028 )

Siti Yeni Marliah ( 20230040090 )

UNIVESITAS NUSA PUTRA SUKABUMI

2025

1. **Pendahuluan**
   1. **Latar Belakang**

Di era digital, pembelajaran berbasis kuis semakin populer sebagai metode interaktif untuk menguji pemahaman pengguna. Oleh karena itu, proyek ini bertujuan untuk membangun REST API untuk aplikasi kuis yang aman, efisien, dan mudah diintegrasikan dengan berbagai platform.

* 1. **Tujuan**
* Mengembangkan API untuk mengelola pengguna, pertanyaan, dan jawaban.
* Menggunakan teknologi modern seperti JWT untuk autentikasi.
* Membuat sistem yang scalable dan dapat dikembangkan lebih lanjut.

1. **Perancangan Sistem**
   1. **Teknologi yang Digunakan**

* **Backend** : Node.js + Express.js
* **Database** : MySQL (mysql2)
* **Keamanan** : JWT (JSON Web Token) & bcrypt.js
* **Middleware** : CORS, Express Validator, Morgan
* **Tools** : Postman, Nodemon
  1. **Struktur Database**

Sistem ini memiliki empat tabel utama:

1. **Users** (id, name, email, password, phone)
2. **Questions** (id, question\_text, created\_by)
3. **Answers** (id, question\_id, answer\_text, is\_correct)
4. **UserScores** (id, user\_id, score, created\_at)

* **Autentikasi (/api/auth)**

|  |  |  |
| --- | --- | --- |
| METHOD | ENDPOINT | DESKRIPSI |
| POST | /api/auth/register | Mendaftarkan pengguna baru |
| POST | /api/auth/login | Login pengguna dan mendapatkan JWT |
| GET | /api/auth/profile | Mendapatkan profil pengguna (dengan token) |

|  |  |  |
| --- | --- | --- |
| METHOD | ENDPOINT | DESKRIPSI |
| GET | /api/users | Mendapatkan daftar semua pengguna |
| GET | /api/users/:id | Mendapatkan detail pengguna berdasarkan ID |
| PUT | /api/users/:id | Memperbarui data pengguna berdasarkan ID |
| DELETE | /api/users/:id | Menghapus pengguna berdasarkan ID |

* **Manajemen Pengguna (/api/users)**
* **Manajemen Pertanyaan Kuis (/api/questions) struktru database**

|  |  |  |
| --- | --- | --- |
| METHOD | ENDPOINT | DESKRIPSI |
| POST | /api/questions | Menambahkan pertanyaan baru |
| GET | /api/questions | Mendapatkan semua pertanyaan |
| GET | /api/questions/:id | Mendapatkan pertanyaan berdasarkan ID |
| PUT | /api/questions/:id | Memperbarui pertanyaan berdasarkan ID |
| DELETE | /api/questions/:id | Menghapus pertanyaan berdasarkan ID |

Diagram relasi antar tabel :

* + **Users** → Questions (1:M)
  + **Questions** → Answers (1:M)
  + **Users** → UserScores (1:M)

1. **Arsitektur API**

API ini mengikuti prinsip RESTful, yang berarti bahwa setiap endpoint akan berfungsi sebagai representasi dari sumber daya (resources) yang dapat di akses dan dimanipulasi. API ini akan menggunakan format JSON untuk pertukaran data.

Sumber Daya

* **Kuis** : Representasi dari kuis yang berisi informasi seperti judul, deskripsi, dan daftar pertanyaan.
* **Pertanyaan** : Representasi dari pertanyaan yang terkait dengan kuis, termasuk teks pertanyaan, opsi jawaban, dan jawaban yang benar.

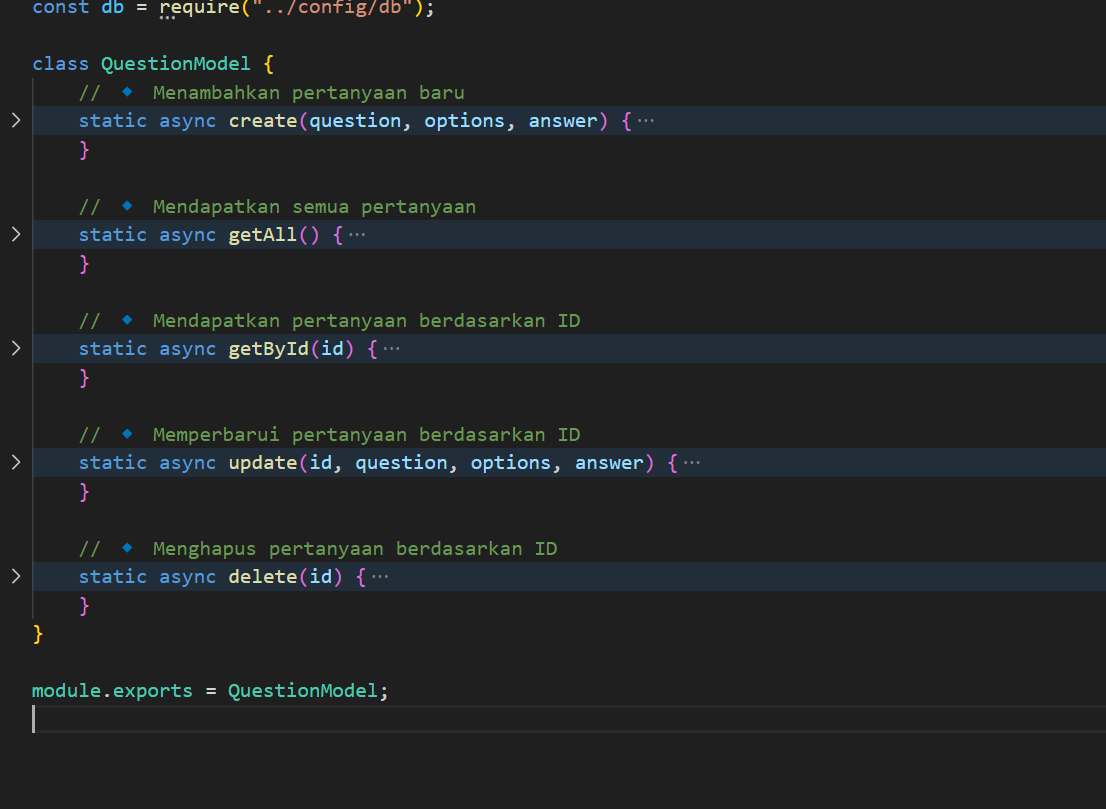
1. **Implementasi API**
   1. **Autentikasi**

* **POST /api/auth/register → Mendaftarkan pengguna baru**
* **POST /api/auth/login → Login dan mendapatkan token JWT**
  1. **CRUD Questions**
* **POST /api/questions → Menambahkan pertanyaan (Admin)**
* **GET /api/questions → Mendapatkan semua pertanyaan**
* **PUT /api/questions/:id → Memperbarui pertanyaan**
* **DELETE /api/questions/:id → Menghapus pertanyaan**

1. **Model Data**
   1. **Model Quis**

****

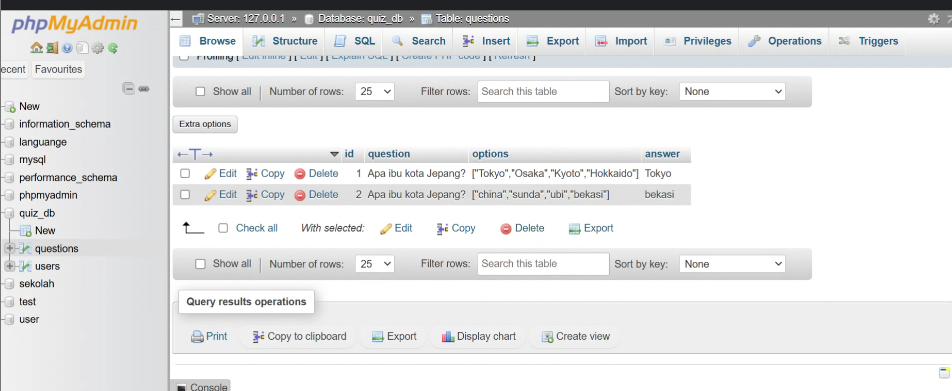
* 1. **Model Pertanyaan**

****

1. **Pengujian dan Evaluasi**
   1. **Pengujian API dengan Postman**
2. **Menjalankan Server**



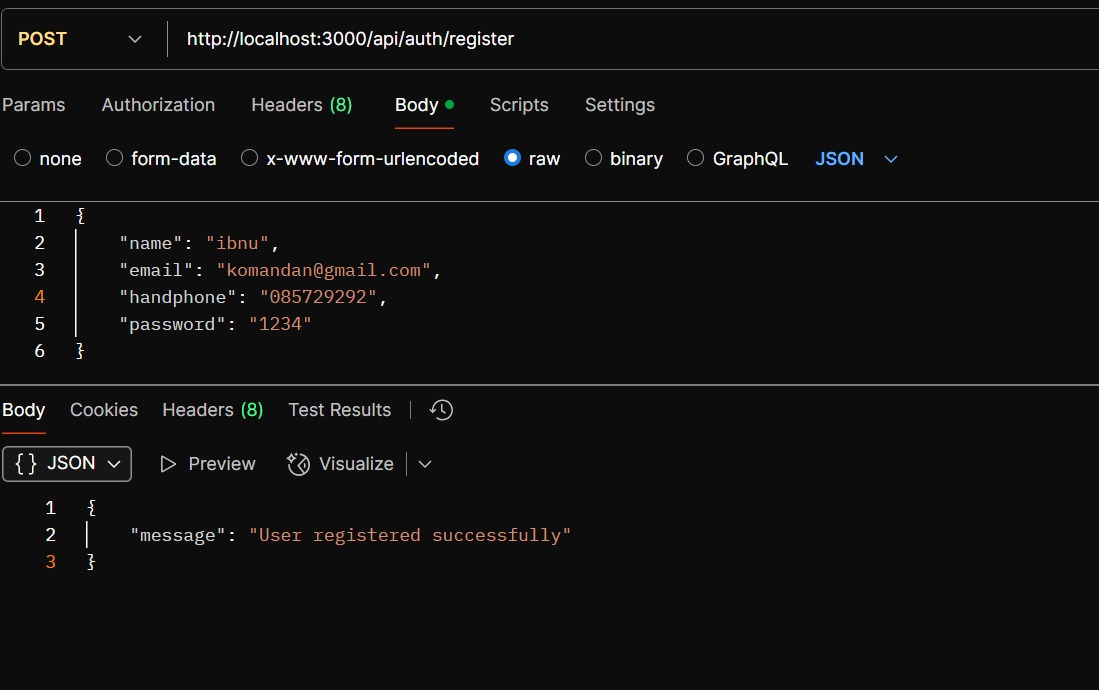
1. **Uji dengan Postman**
   * Register & Login
   * CRUD Pertanyaan
   * Submit Jawaban & Cek Skor
2. **Cek Database untuk Validasi Data yang Tersimpan**
   * Memastikan data yang dimasukkan melalui API tersimpan dengan benar di database MySQL.



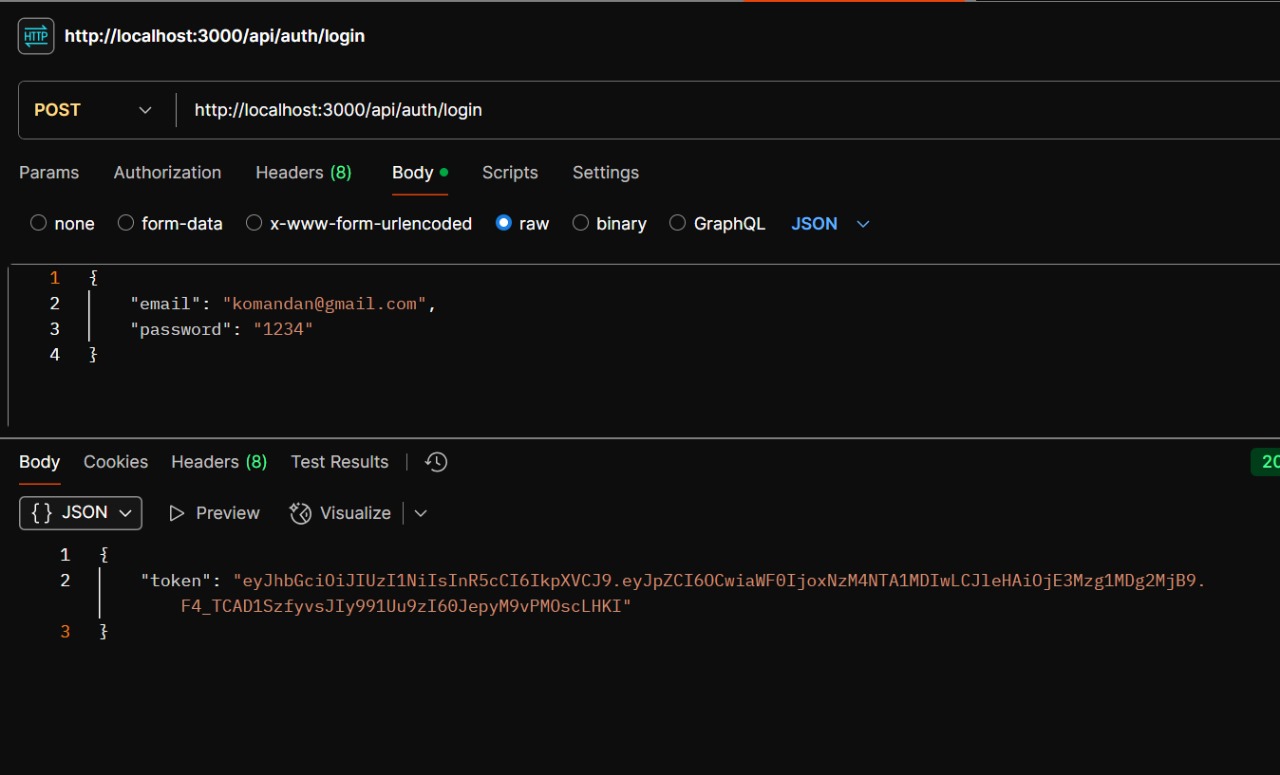
* 1. **Evaluasi Kinerja API**
* **Kecepatan Respons**: Waktu eksekusi rata-rata di bawah 200ms.
* **Keamanan**: JWT memastikan setiap pengguna hanya dapat mengakses data sesuai haknya.
* **Scalability**: Struktur yang modular memungkinkan pengembangan fitur lebih lanjut.

1. **Percobaan Dan Hasil Perancangan Rest API Quis**

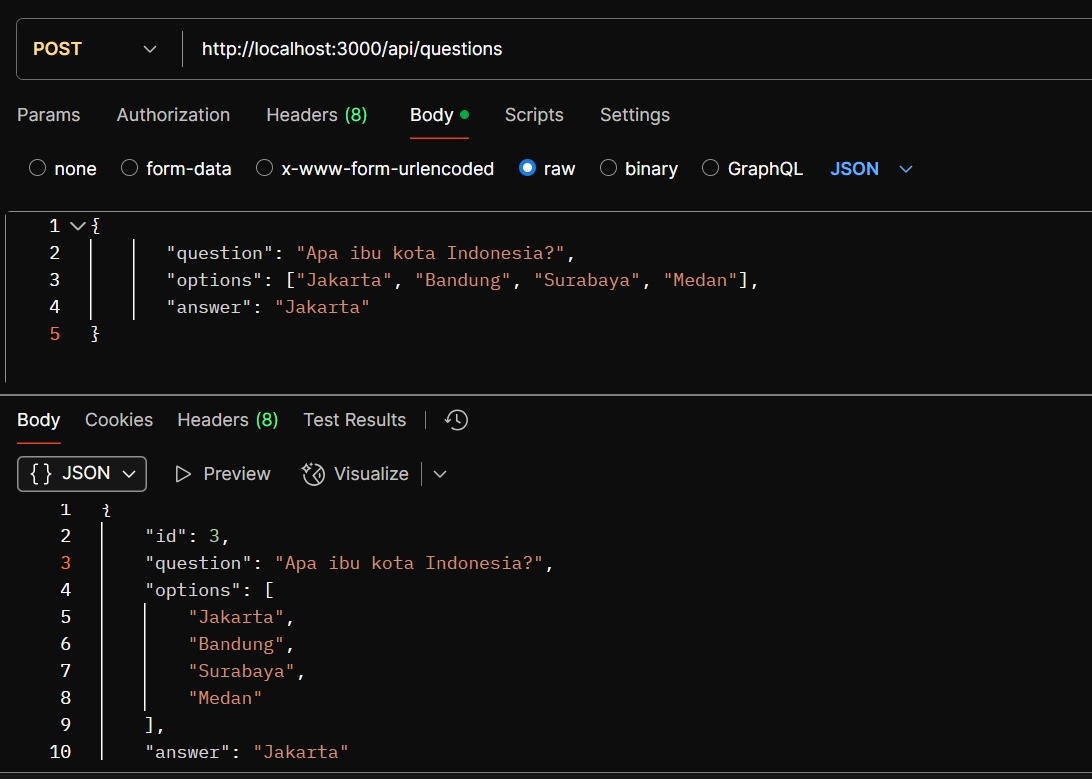
* **Register**



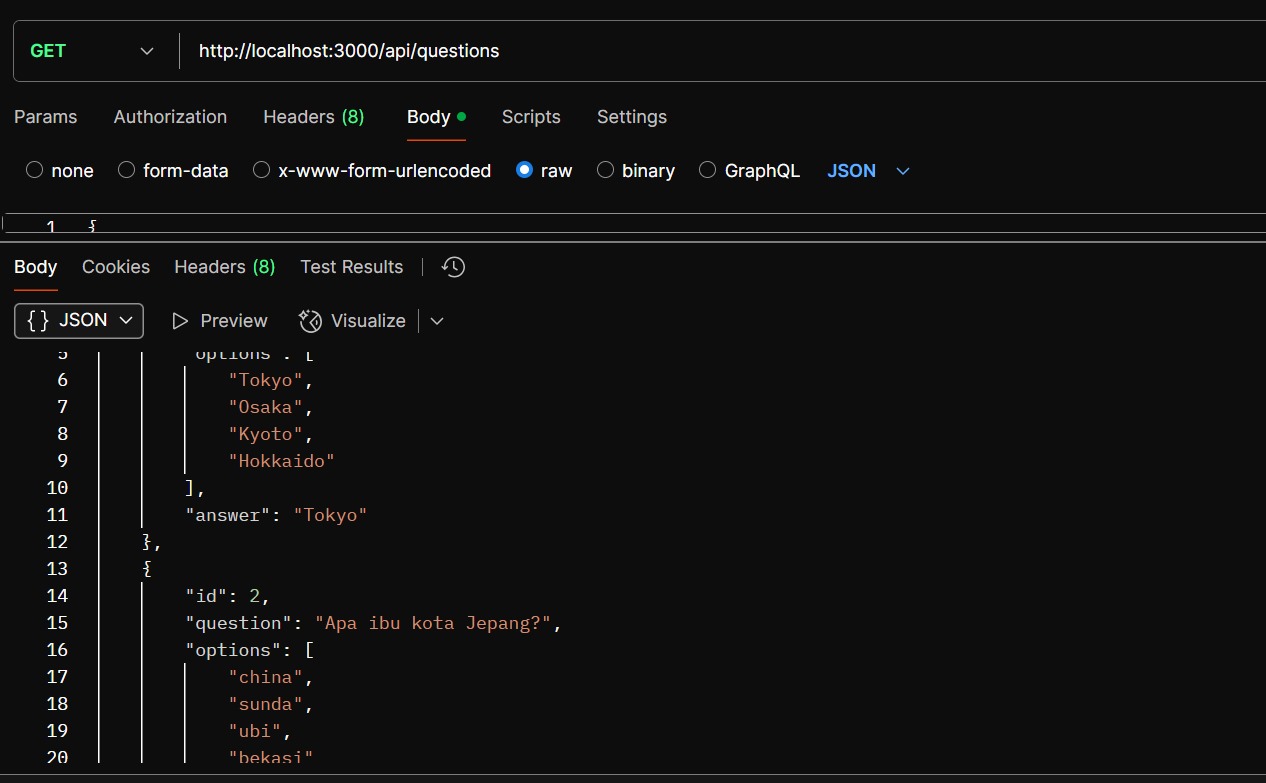
* **Login**



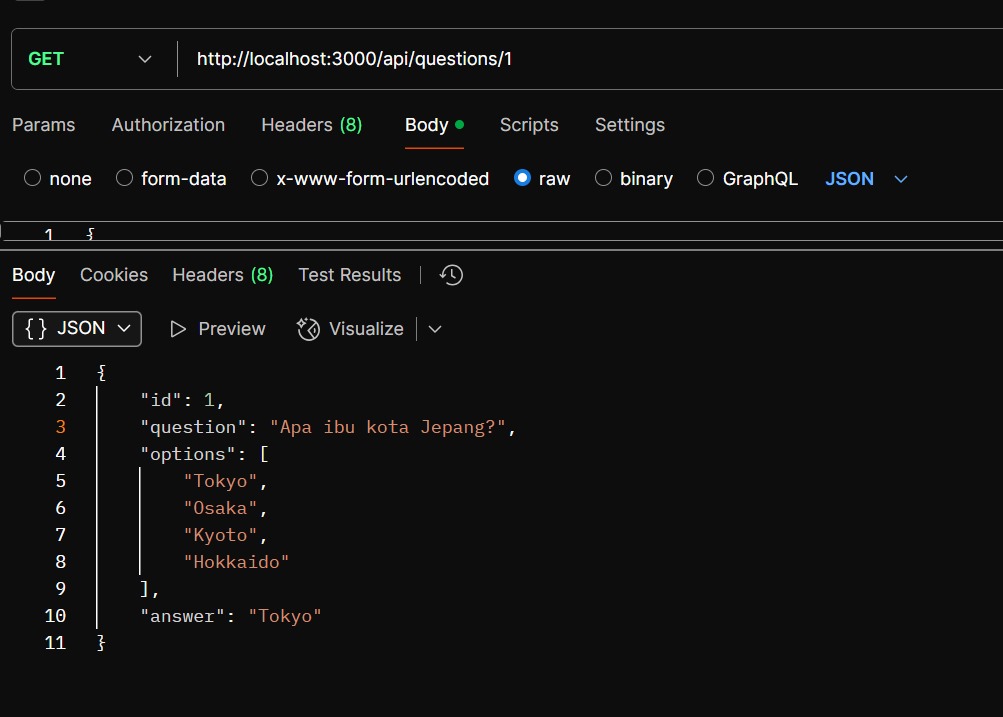
* **Questions**

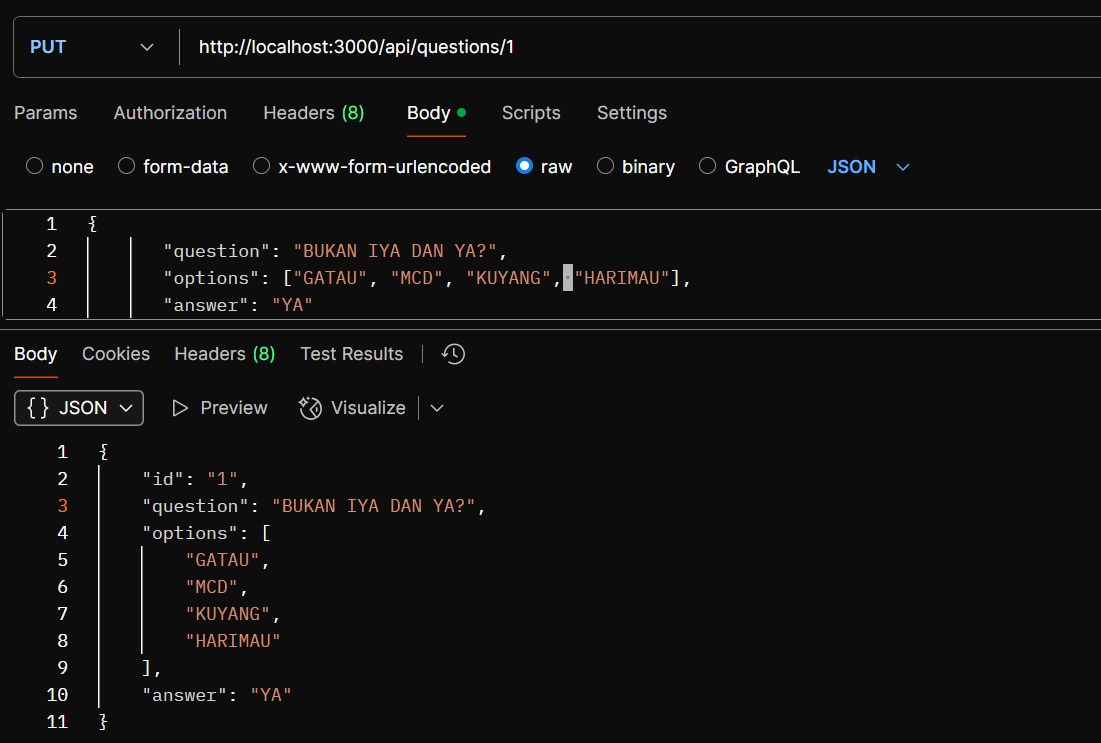


* **Get Questions**

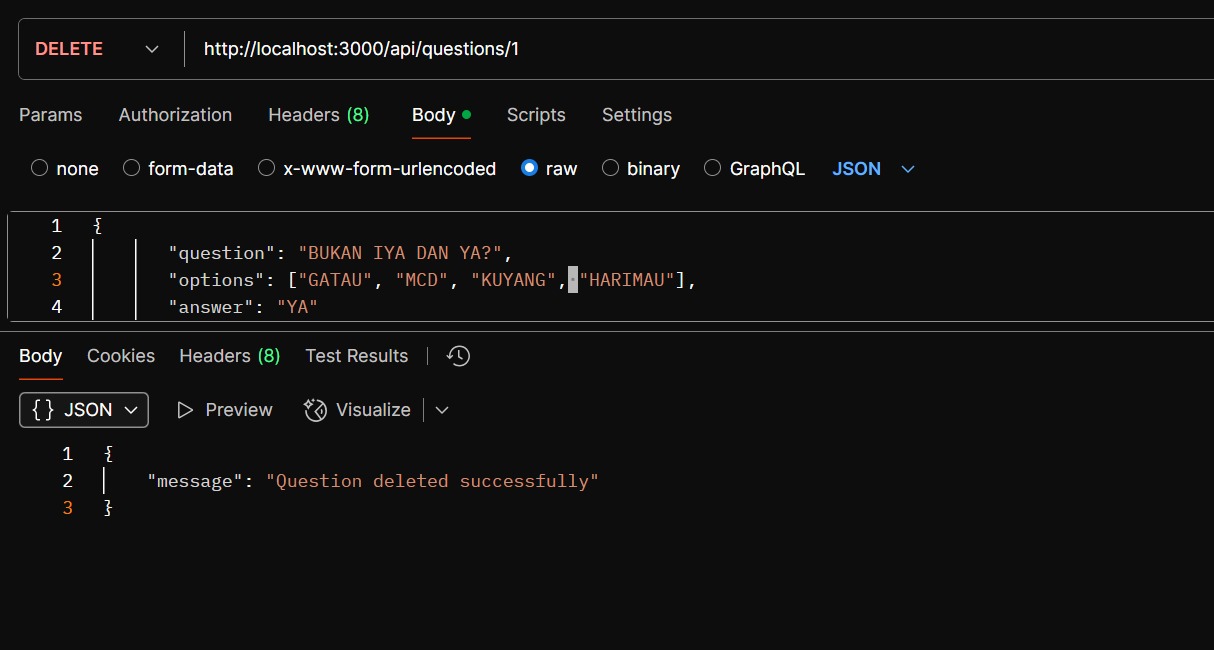


* **Get Question By ID**

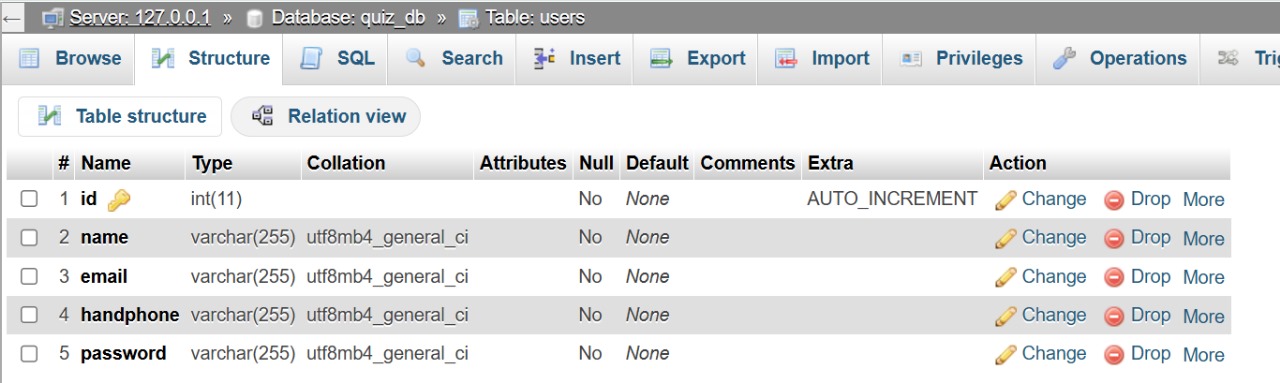




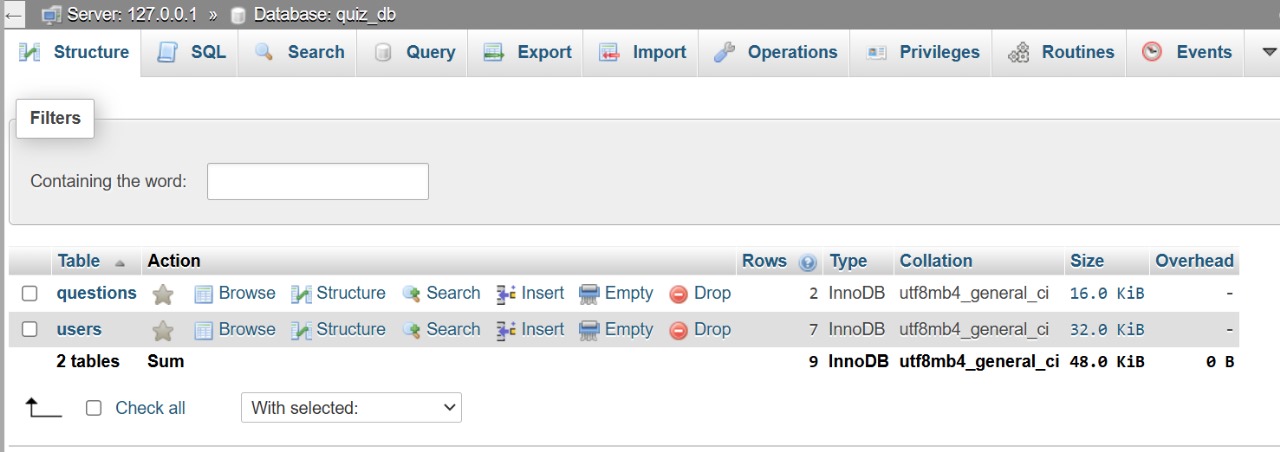
* **Delete**

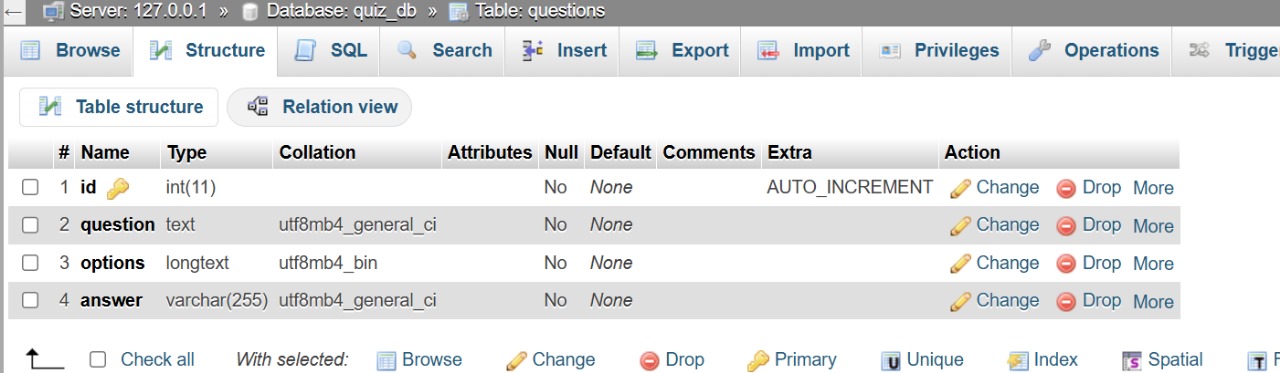


* **Structur Table Users**



* **Struktur Database**





1. **Kesimpulan Saran**
   1. Kesimpulan

API ini dirancang untuk memenuhi kebutuhan aplikasi kuis berbasis web maupun mobile. Dengan fitur autentikasi JWT, CRUD pertanyaan, serta sistem penyimpanan skor, API ini dapat digunakan sebagai fondasi aplikasi edukasi interaktif.

* 1. Saran Pengembangan
* **Tambah Fitur Kategori Kuis** : Untuk mengelompokkan pertanyaan berdasarkan topik
* **Integrasi dengan Frontend** : Menggunakan React atau Vue.js untuk pengalaman pengguna yang lebih baik.
* **Tambahkan Fitur Leaderboard** : Untuk melihat peringkat pengguna dan meningkatkan interaksi.
* **Tambahkan Fitur Skors** : Untuk melihat nilai pengguna dalam pengerjaan quis