MONGODB PROJECT

# Supreme Salon – Online Salon Appointment Booking System

**NAME : JEEVITHA R**

**REGNO : 23BAI1550**

**CAMPUS : CHENNAI**

**NAME : JEEVITHA R**

**REGNO : 23BAI1550**

**CAMPUS : CHENNAI**

**PROJECT DESCRIPTION:**

The Online Salon Appointment Booking System utilizes **MongoDB Atlas** as its cloud database, structured into three primary collections: **Users**, **Appointments**, and **Feedbacks**.

* **Users Collection:** Stores user information including name, email, password (hashed), and role (customer or admin). This collection manages authentication and role-based access.
* **Appointments Collection:** Records salon appointments booked by users. Each appointment document contains references to the user who booked it, the selected salon service (e.g., haircut, facial), the appointment date, and the chosen time slot.
* **Feedbacks Collection:** Stores customer feedback post-service. Each feedback includes a star rating represented as a numeric value (1 to 5) alongside optional textual comments. The rating is saved as a number in MongoDB for easy querying and analytics.

This full stack system, built with **Node.js and Express.js** for the backend, **React.js** for the frontend, and **MongoDB Atlas** as the database, offers a scalable and efficient platform for both customers and salon administrators. Customers can create accounts, book and manage appointments, and submit feedback. Admin users have privileged access to monitor all bookings and feedback through a centralized dashboard.

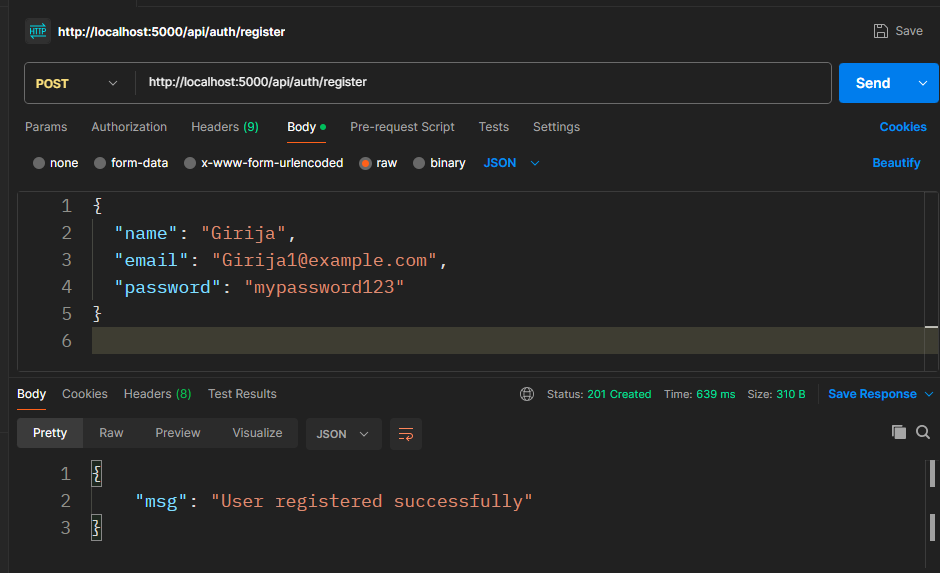
All CRUD operations for users, appointments, services, and feedback are fully supported and validated. These APIs are thoroughly tested using **Postman** through GET, POST, PUT, and DELETE requests to ensure reliability and correctness.

The system ensures secure authentication, smooth client-server communication, and role-based data access, streamlining salon operations and enhancing user experience.

**BACKEND & POSTMAN SCREENSHOT**

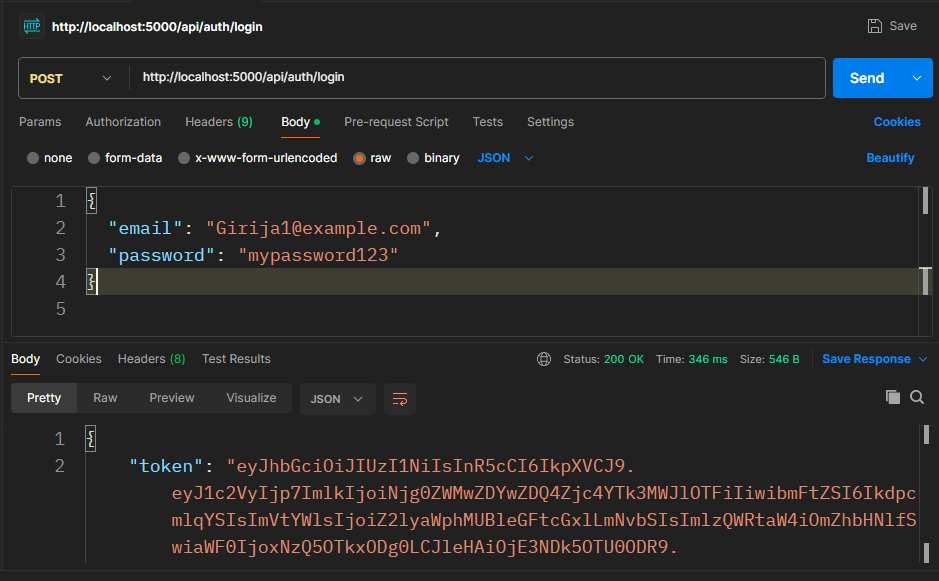
1. **User Registration via Postman (POST /auth/register)**

* **Type:** POST
* **URL:** http://localhost:5000/api/auth/register



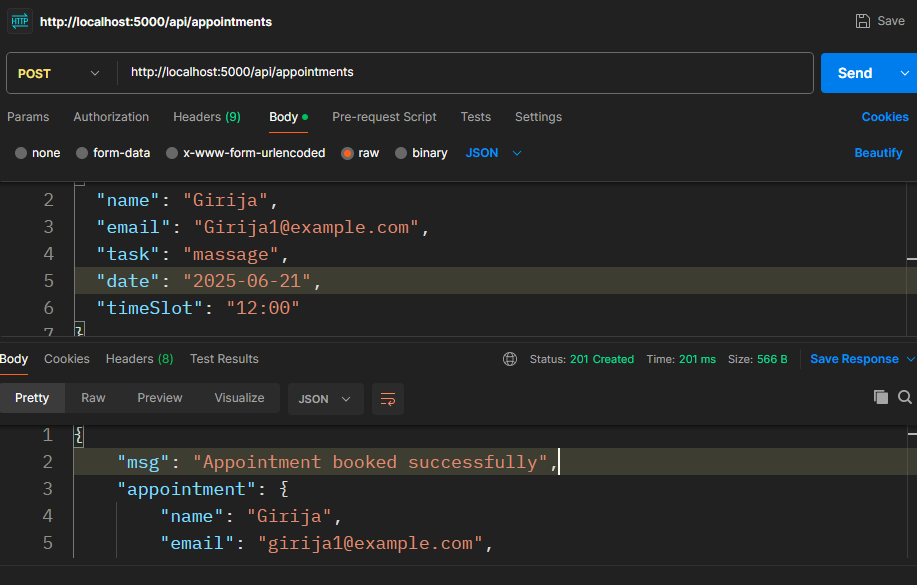
1. **User Login (Get Token)**

* **Type:** POST
* **URL:** <http://localhost:5000/api/auth/login>

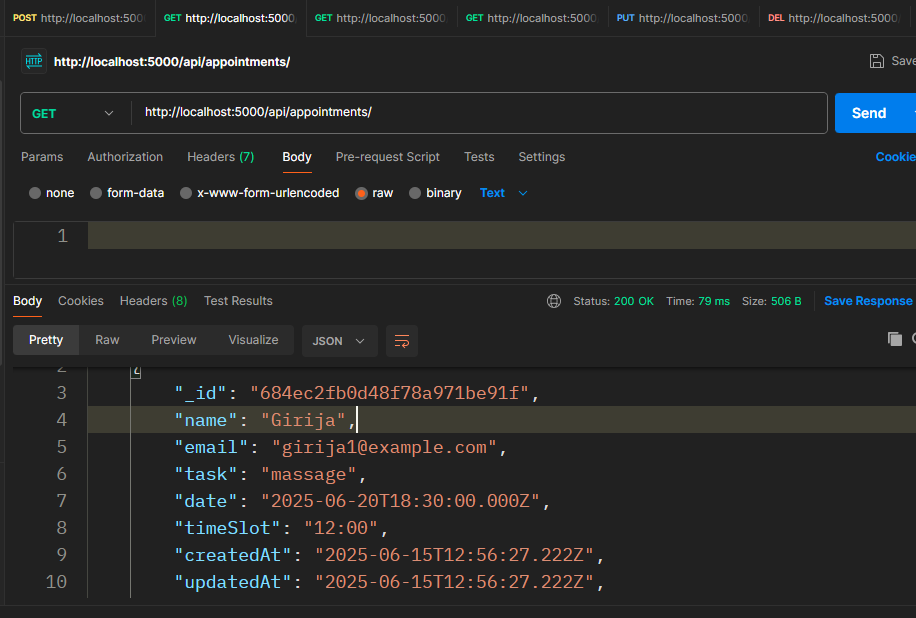


1. **Book an Appointment**

* **Type:** POST
* **URL:** <http://localhost:5000/api/appointments>



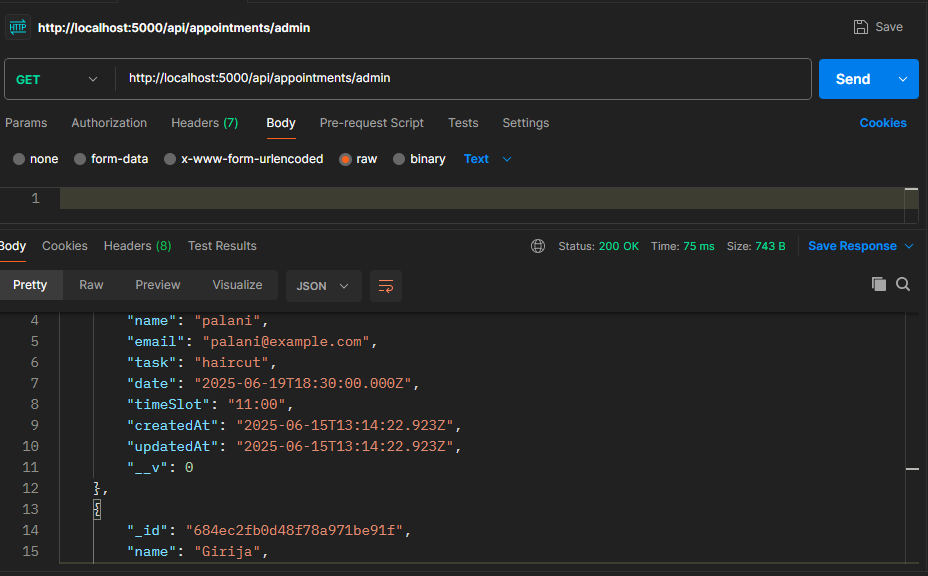
1. **Get Appointment**



1. **Get All Appointments (Admin only)**

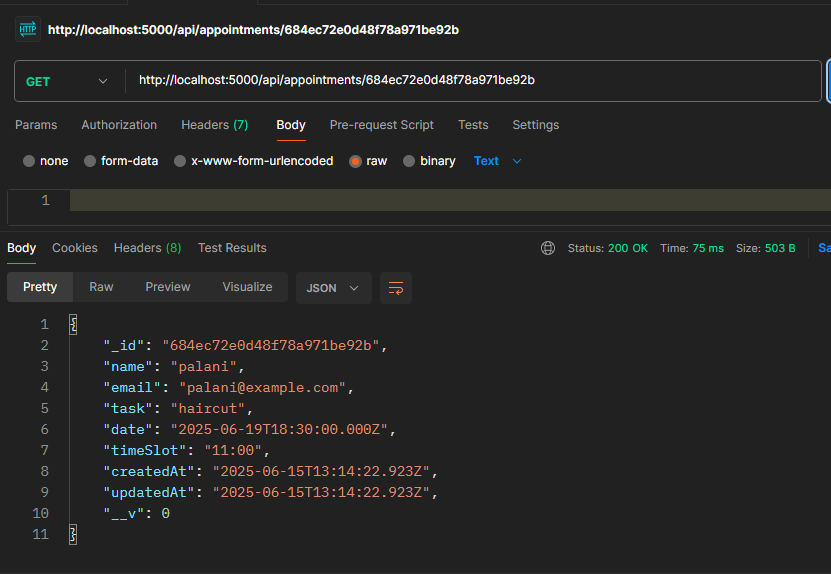
 **Type:** GET

 **URL:** http://localhost:5000/api/appointments/admin



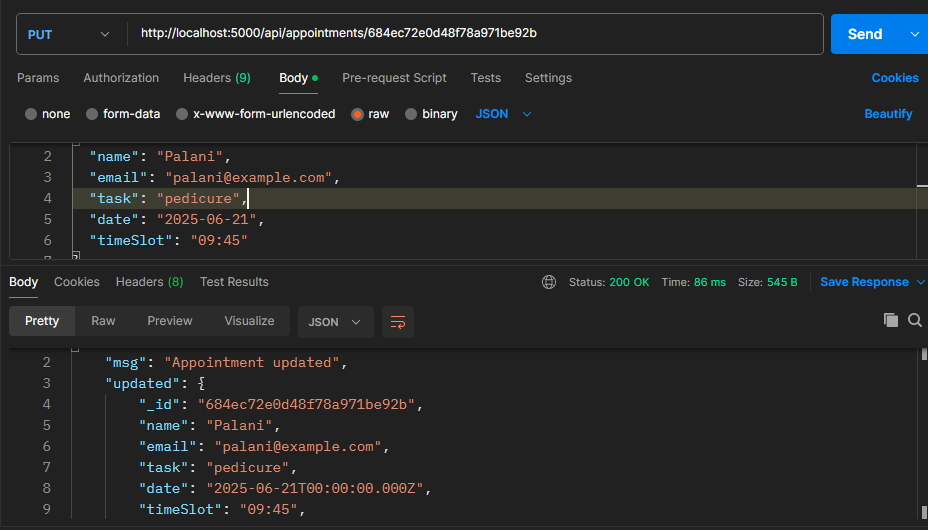
1. **Get Appointment by ID**

**Method:** GET  
**URL:** http://localhost:5000/appointments/<APPOINTMENT\_ID>

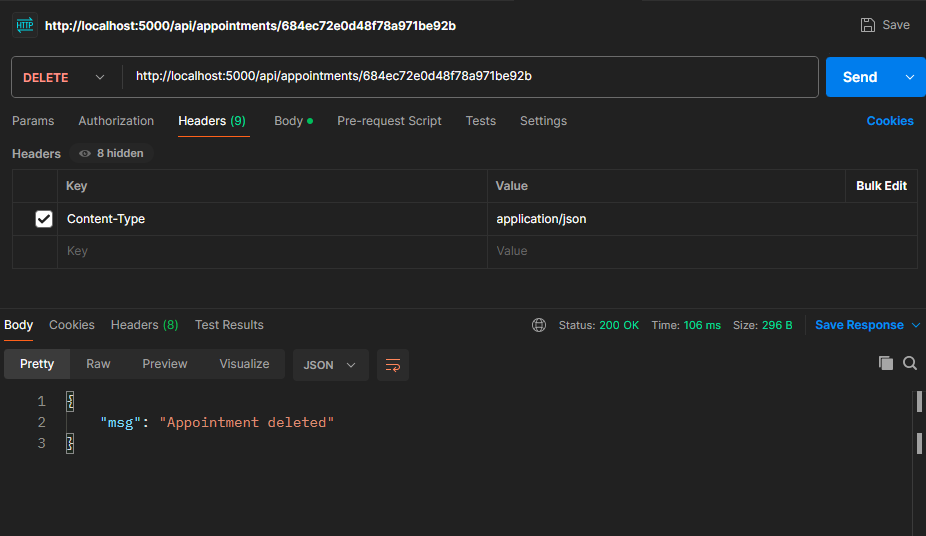


1. **Update Appointment**

**Method:** PUT  
**URL:** http://localhost:5000/appointments/<APPOINTMENT\_ID>



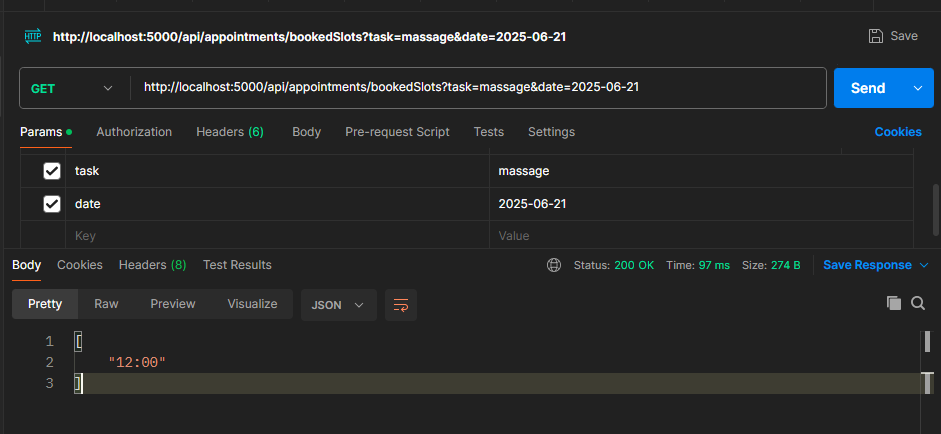
1. **Delete Appointment**



### Get Booked Time Slots for a Task and Date

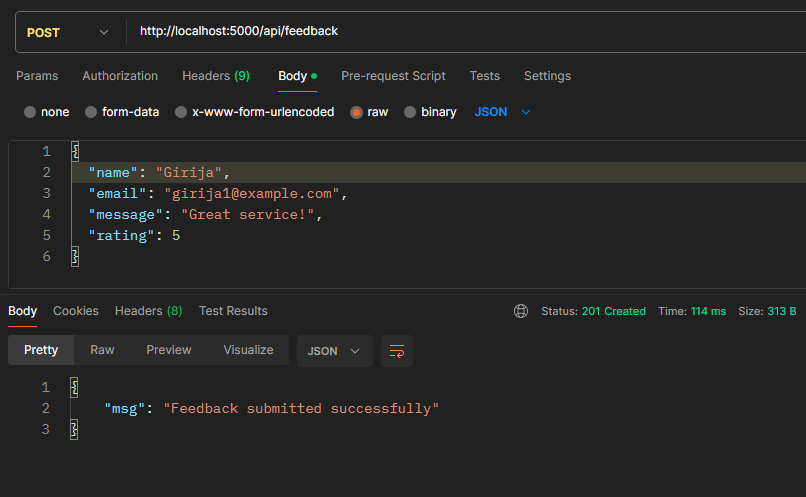
**Method:** GET  
**Endpoint:** http://localhost:5000/api/appointments/bookedSlots?task=massage&date=2025-06-21

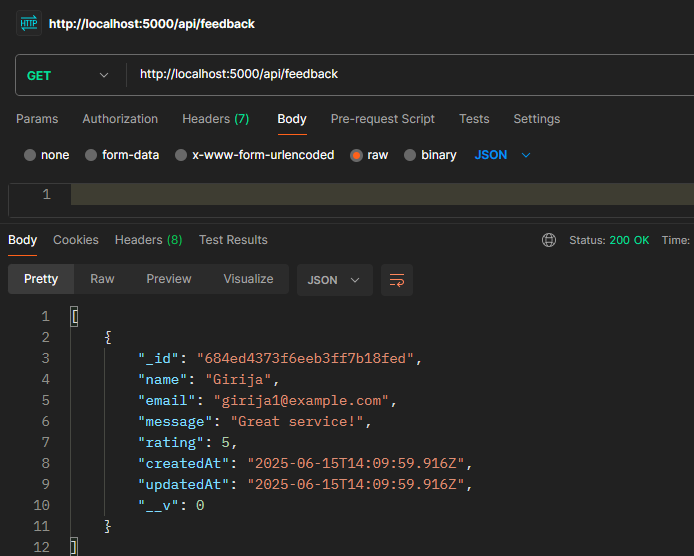
(Because the stored dates are in UTC, a booking made on 2025-06-20 UTC might correspond to 2025-06-21 in IST. Therefore, the query should use the user’s local date to retrieve accurate booked slots.)

****

**10.Postman Request for Creating Feedback & Get All Feedbacks**

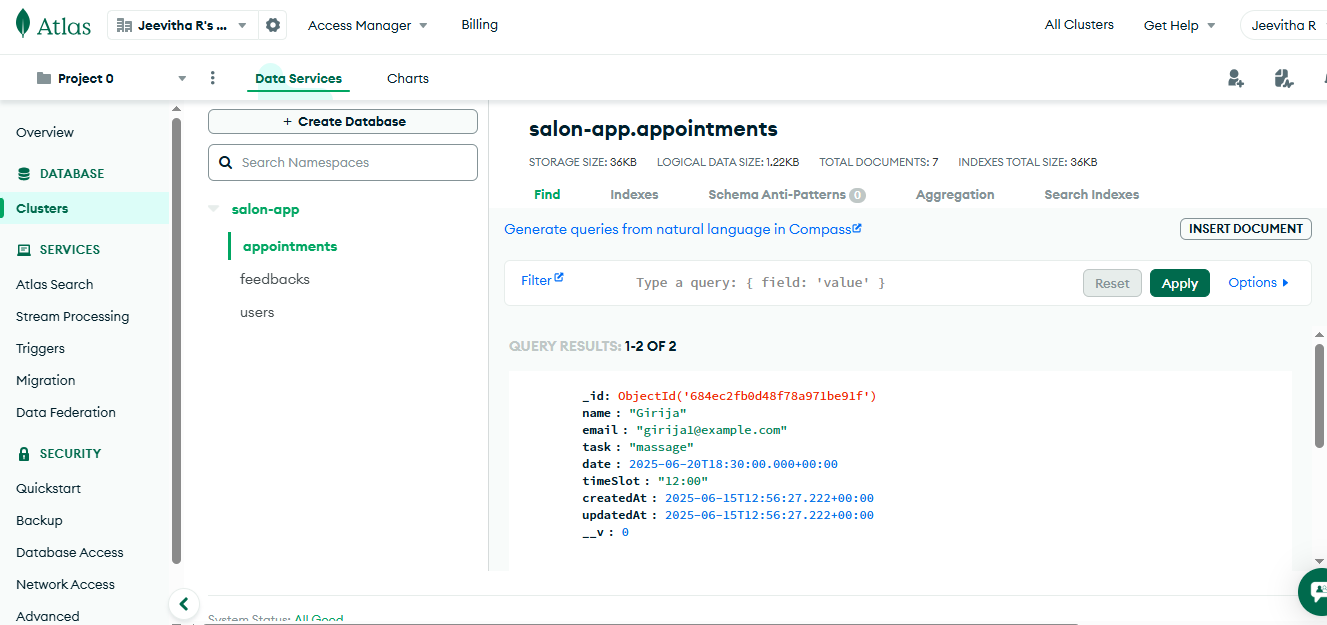
* **Method:** POST & GET
* **URL:** http://localhost:5000/api/feedback



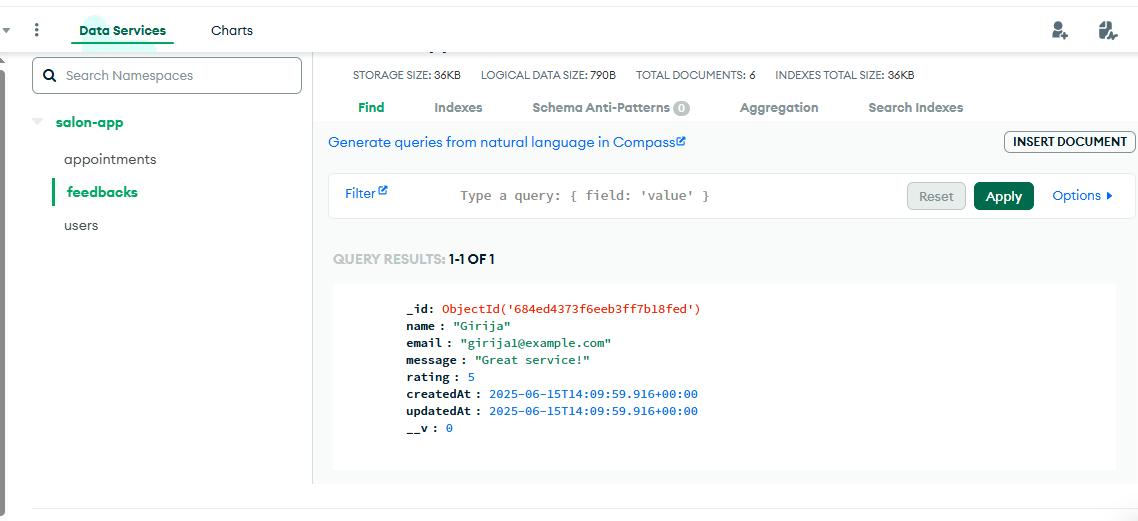


**MONGODB ATLAS AS A DATABASE:**

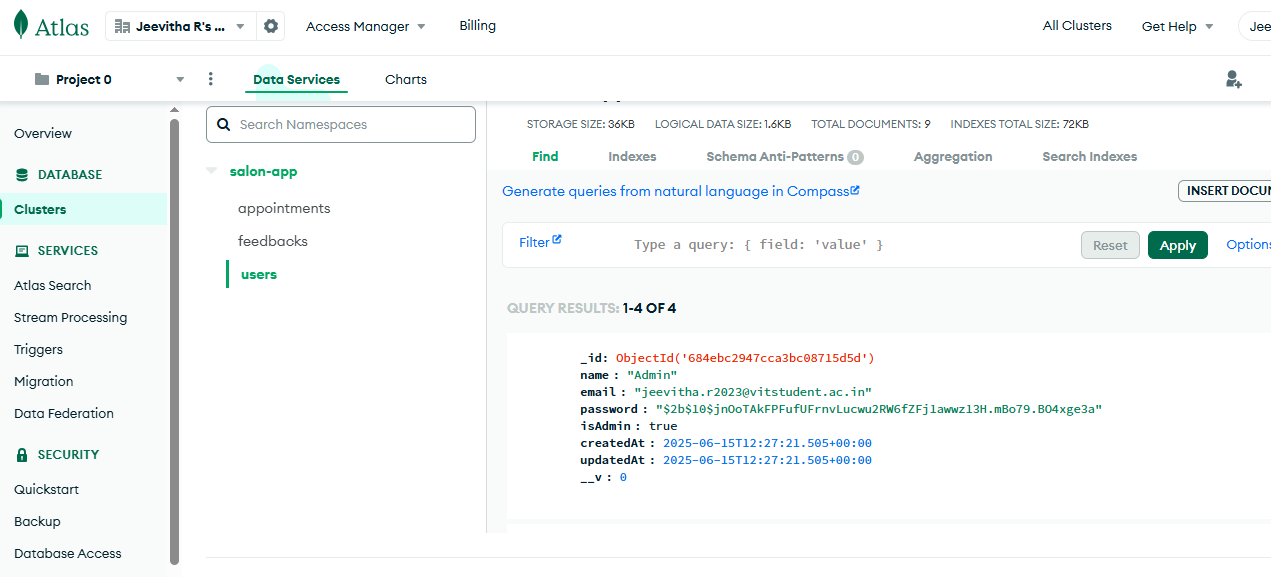
**APPOINTMENTS FROM POSTMAN:**



**FEEDBACKS FROM POSTMAN:**

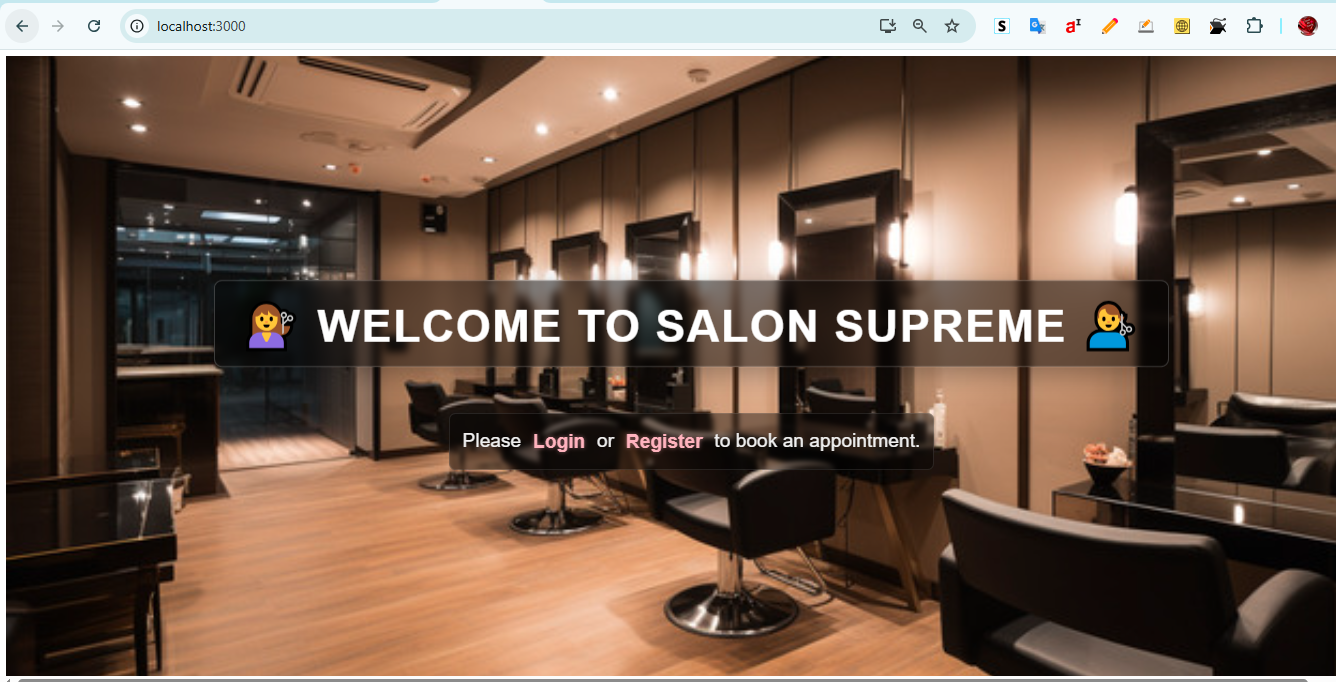


**USERS FROM POSTMAN:**

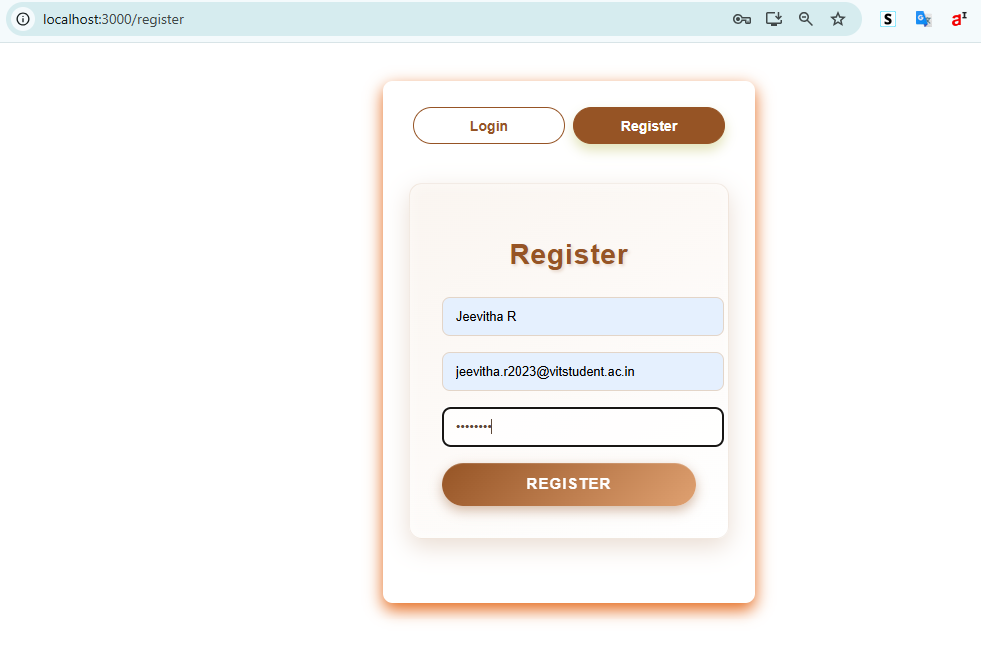


**FRONTEND:**

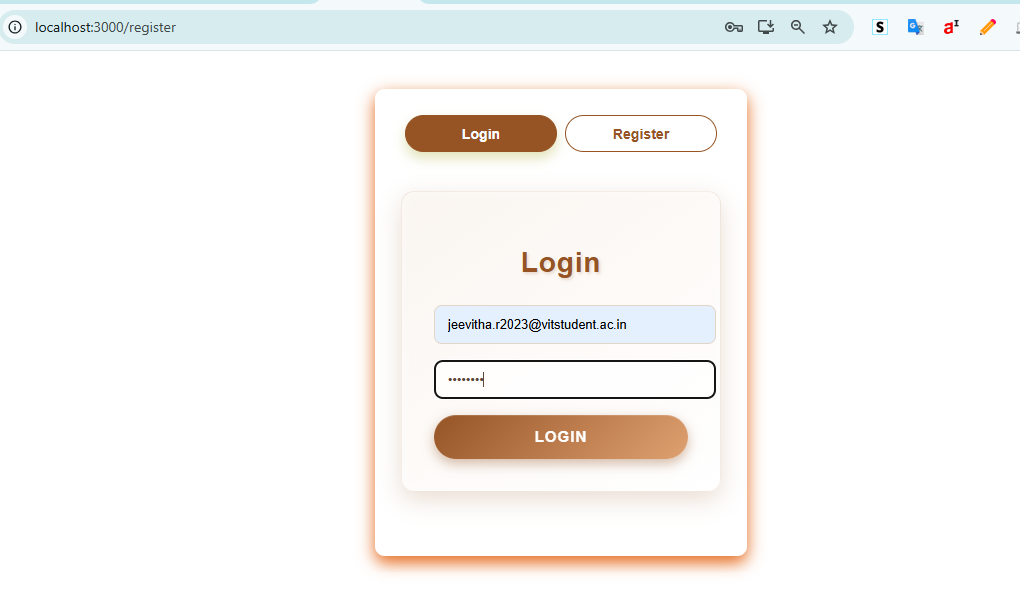
**HOME PAGE:**



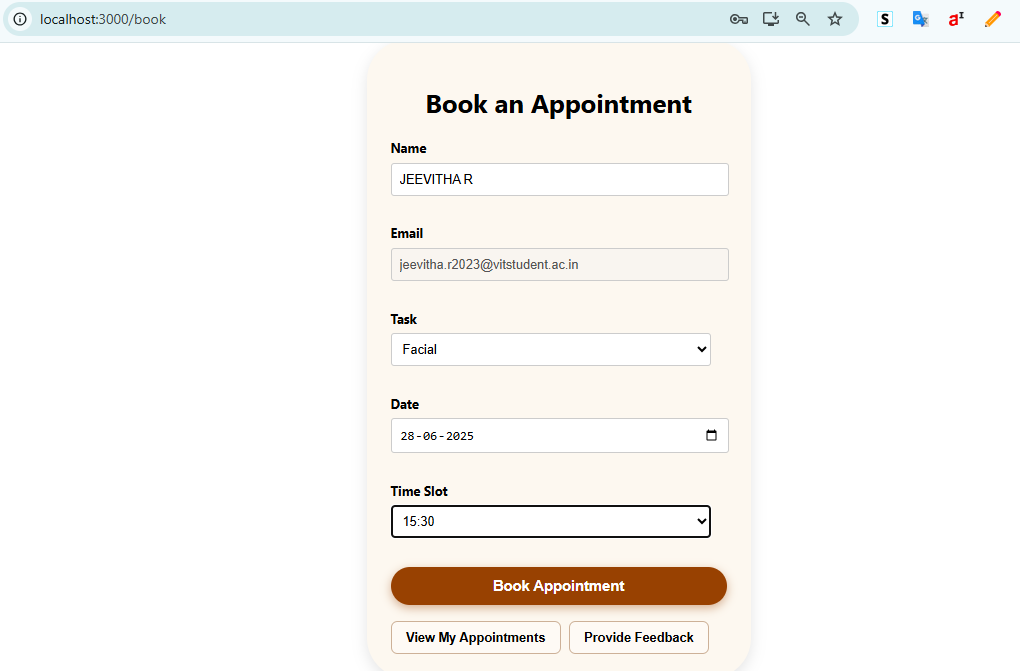
**REGISTER PAGE:**



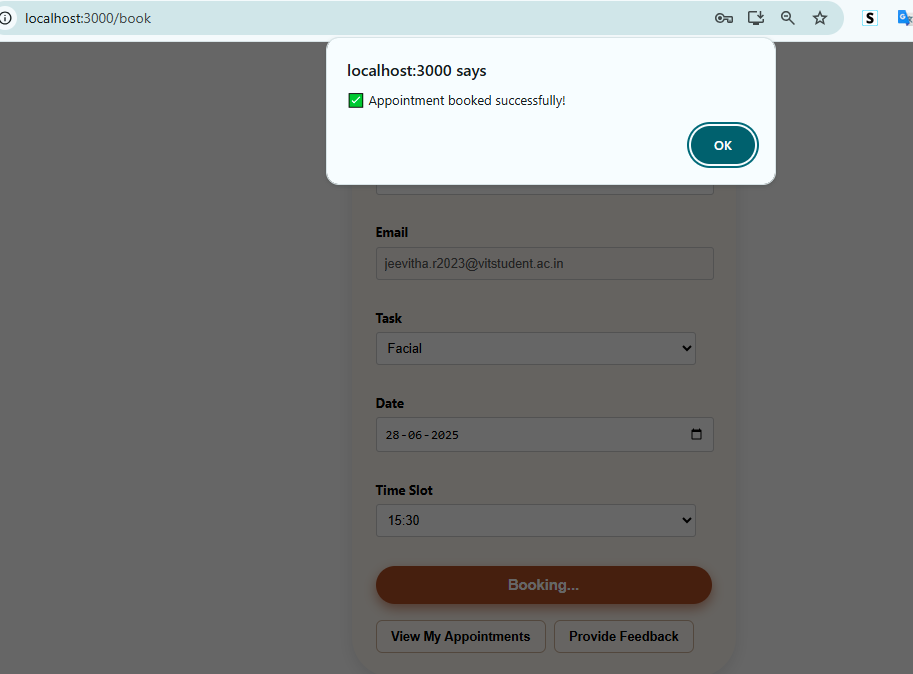
**LOGIN PAGE:**



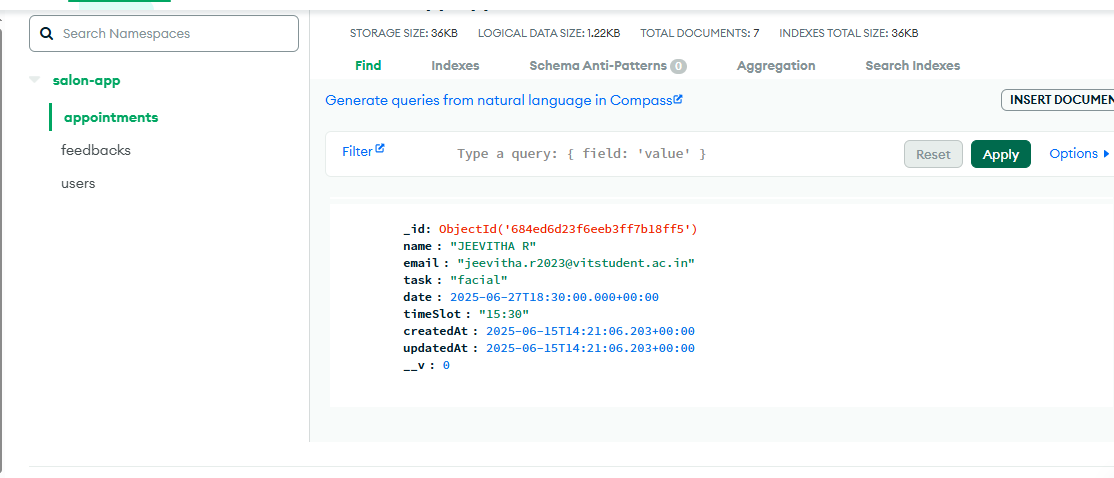
**APPOINTMENT BOOKING PAGE:**



**BOOKING CONFIRMED!**

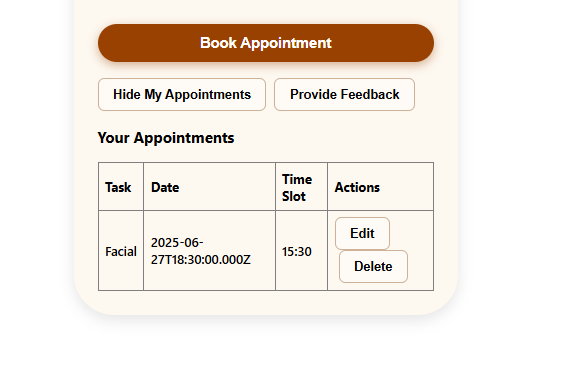


**STORED IN BACKEND (MONGODB ATLAS)**

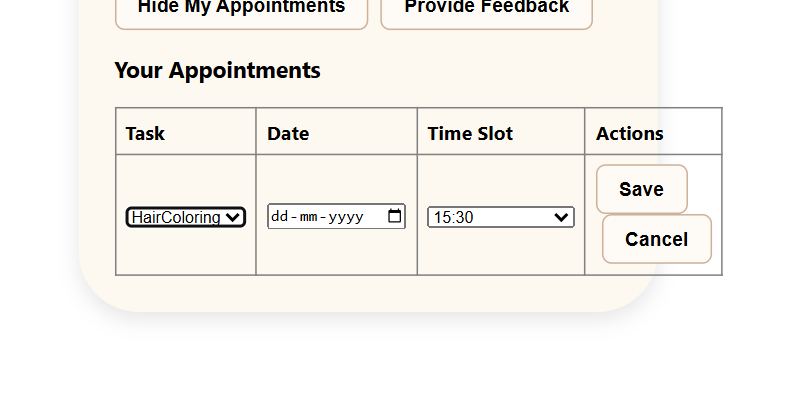


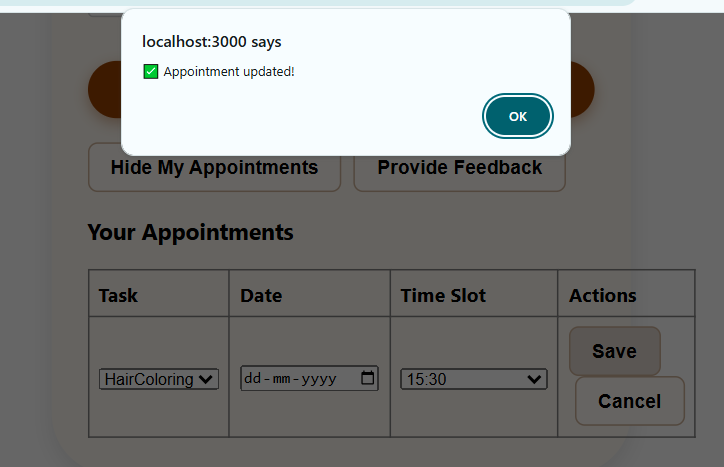
**VIEW MY APPOINTMENTS:**

* **OPTIONS FOR USER :**
* **TO EDIT**
* **TO DELETE**

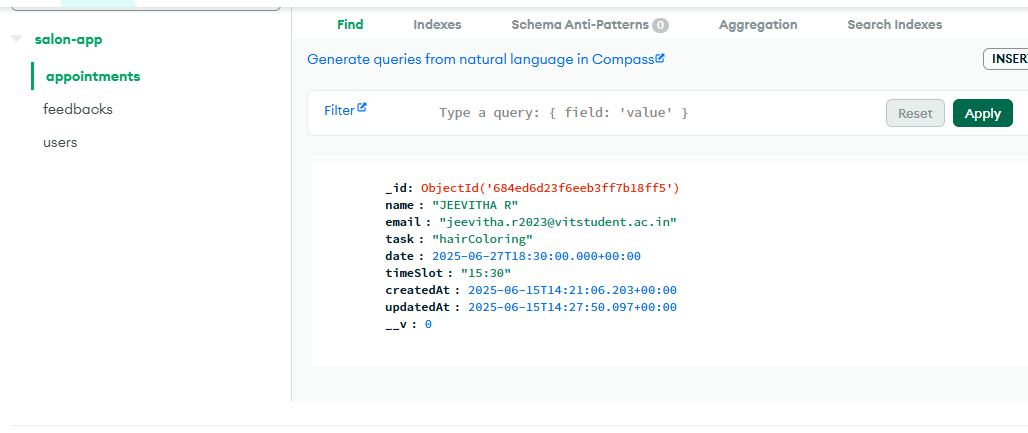


**TO UPDATE USER APPOINTMENT (Edit) :**

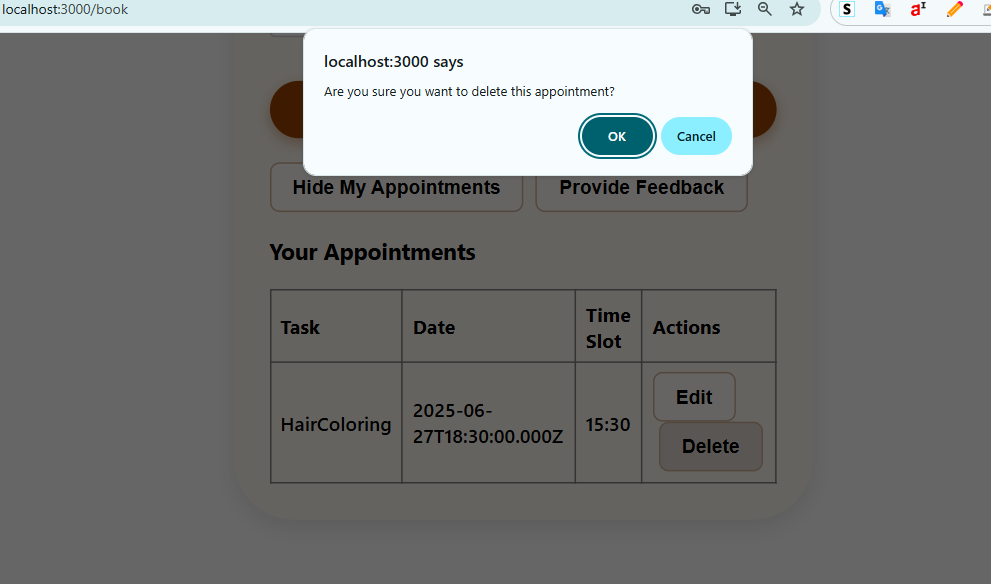


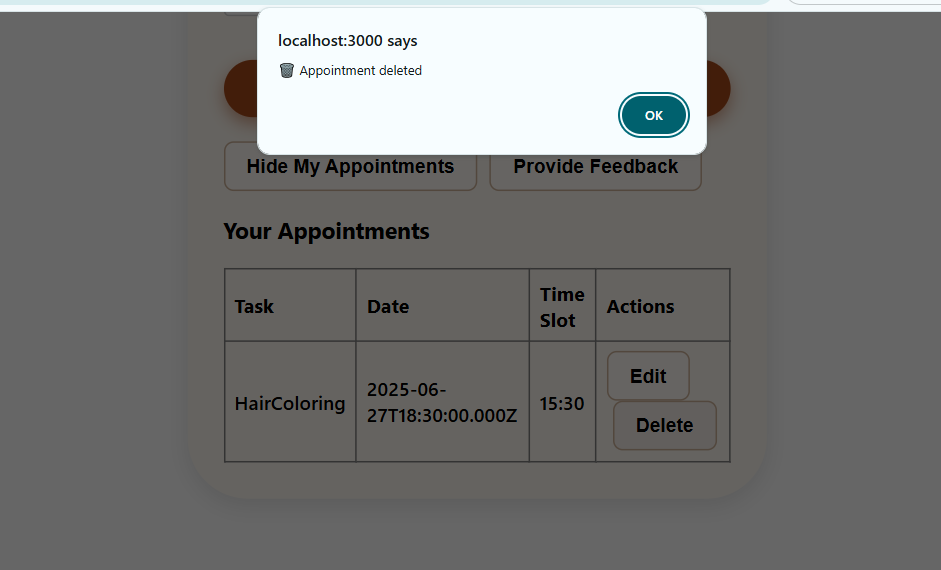


**UPDATED IN BACKEND:**

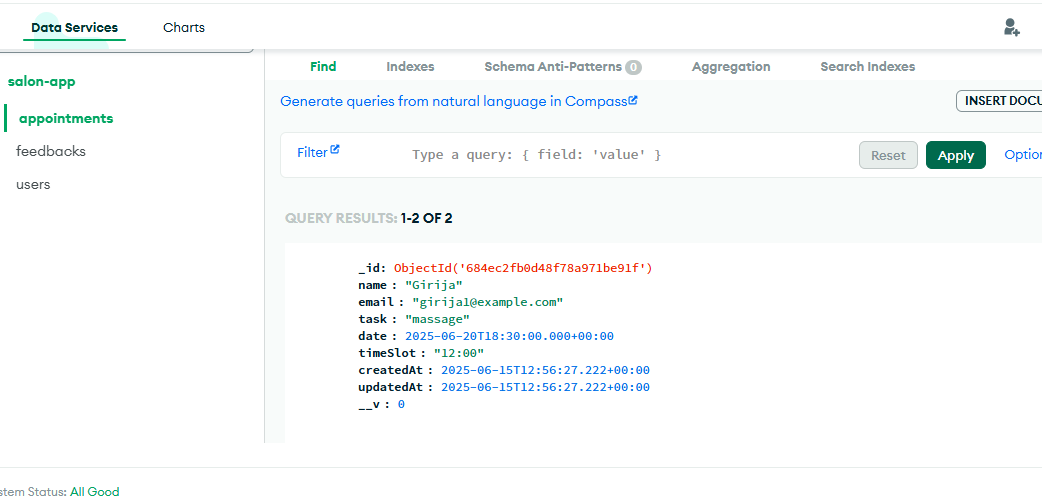


**DELETE THE APPOINTMENT ( delete):**

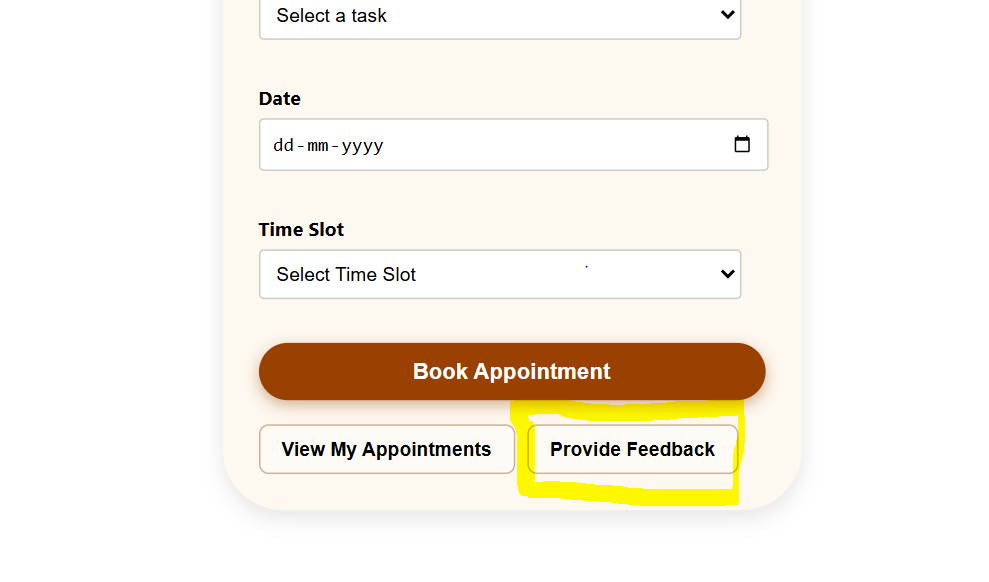


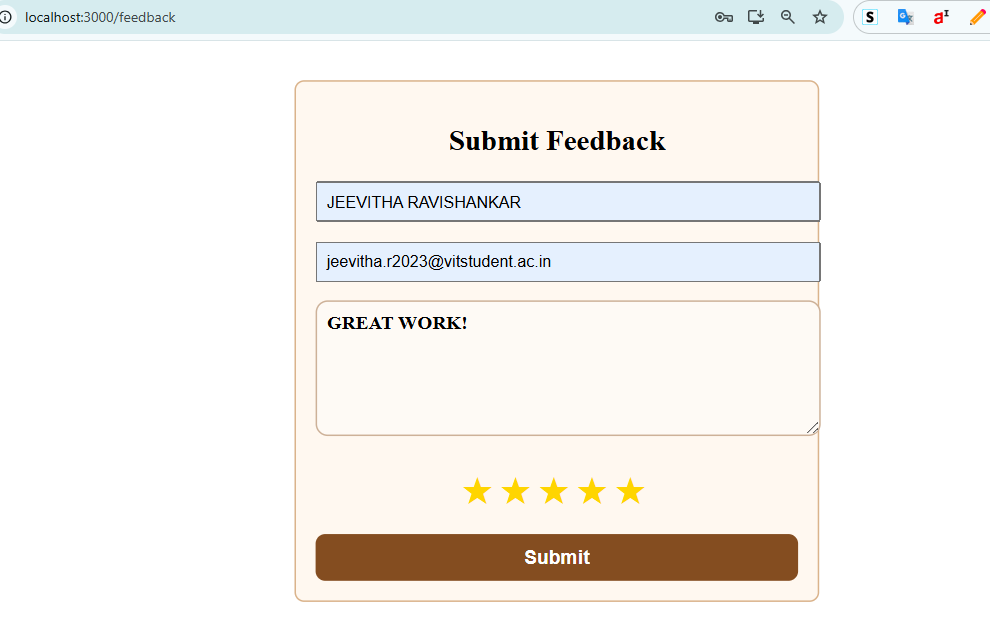


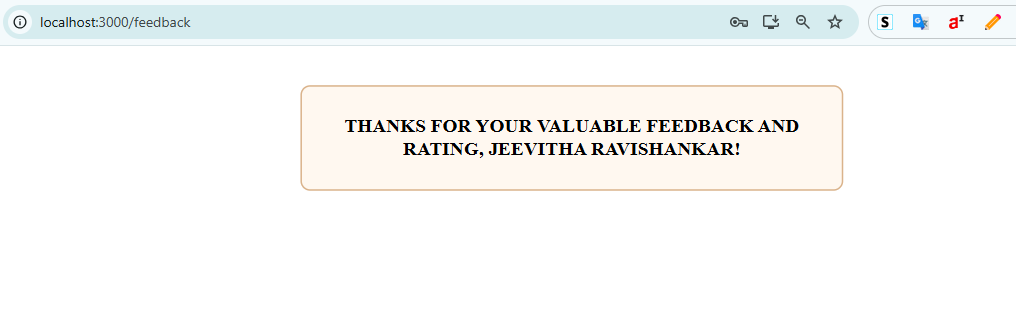
**DELETED IN BACKEND:**

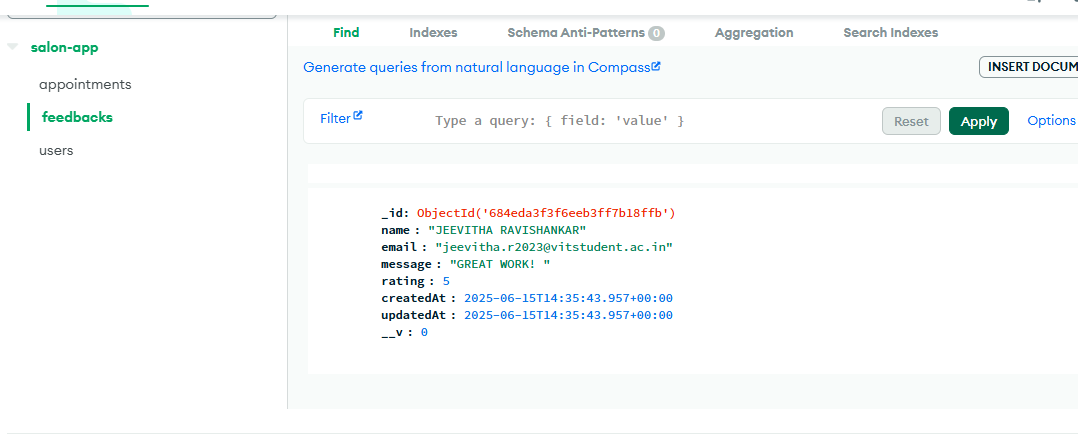


**FEEDBACK PAGE:**





**FEEDBACK STORED IN BACKED:**



**ADMIN PAGE:**

**TO VIEW ALL APPOINTEMENTS AND FEEDBACK OF THE CUSTOMER:**

