The following document will give you an explanation of the code:-



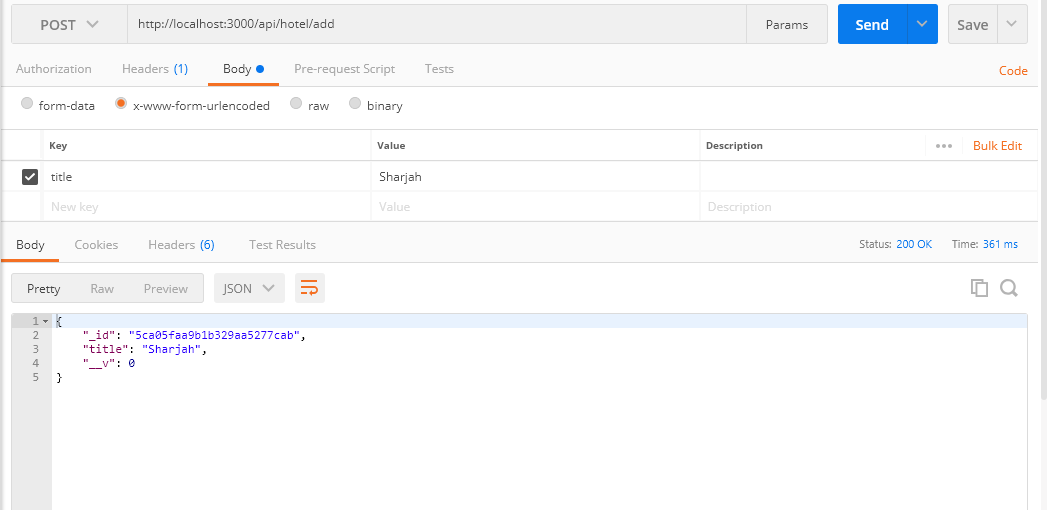
I have seperated the api routes , the model schema and the business logic into three sections. **(Functions , Models , Routes)**

1. The functions contain the business logic of the apis
2. The routes contain the express routes of the REST API
3. The model contains the db schema of the mongodb collections
4. I have used Mongoose ORM for querying MongoDB.

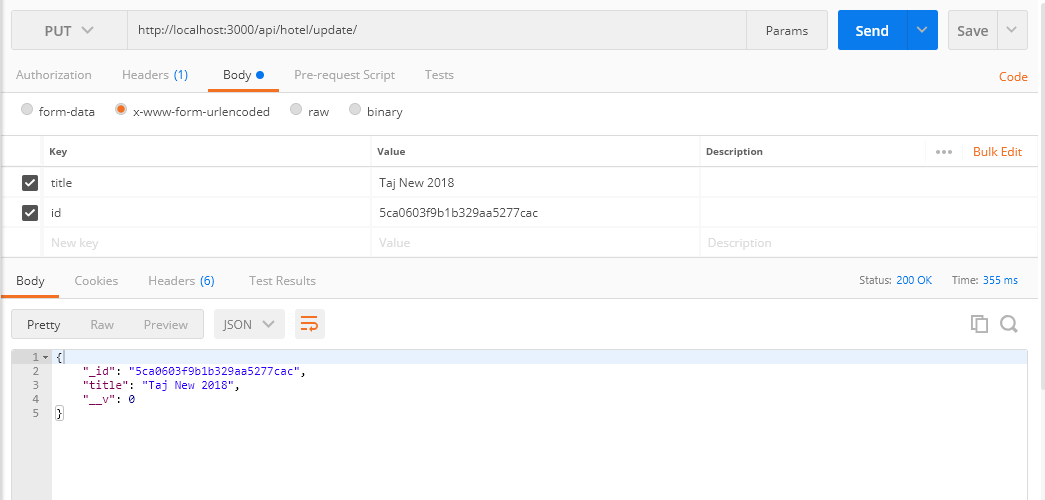
Following are the REST API along with few screenshots to run it in POSTMAN Tool

(Assuming the port is set to 3000)

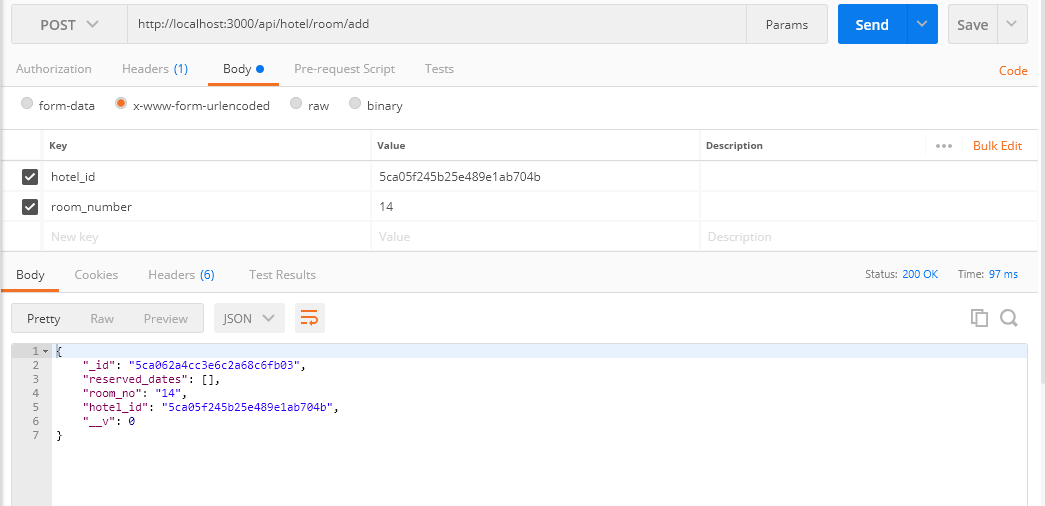
1. Create / update / delete Hotel
2. <http://localhost:3000/api/hotel/add>



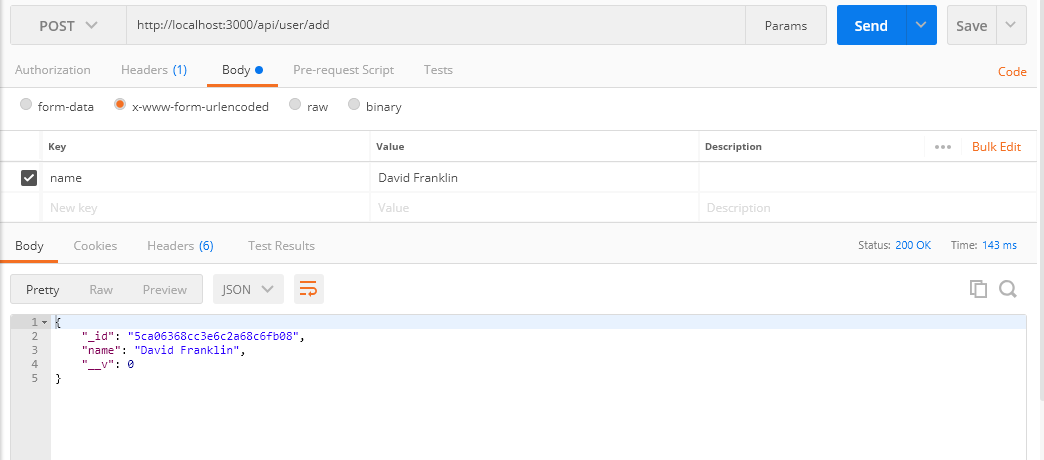
1. <http://localhost:3000/api/hotel/update>



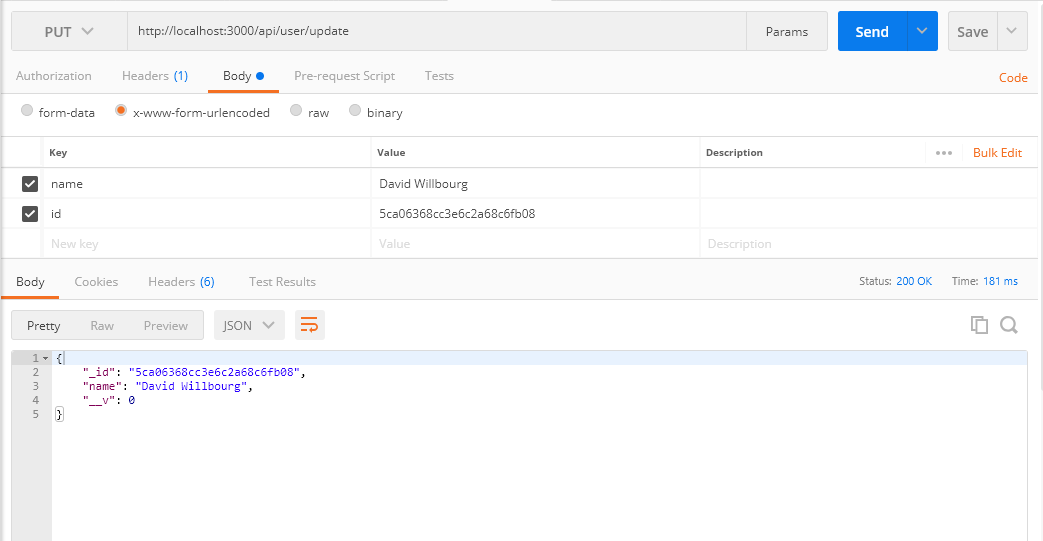
1. http:/localhost:3000/api/hotel/delete/<hotel-id>
2. Create rooms for hotel
3. http:/localhost:3000/api/hotel/room/add



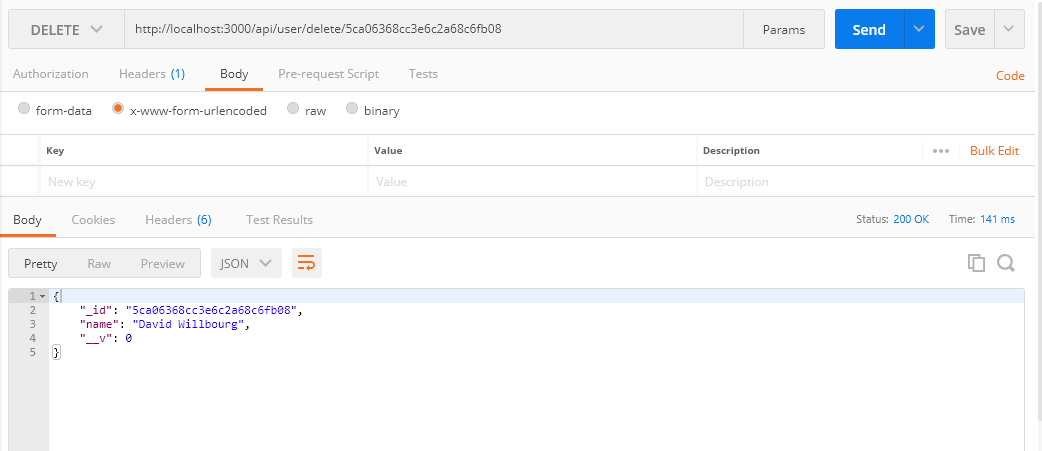
1. Create / Update / Delete Users
2. <http://localhost:3000/api/user/add>



1. <http://localhost:3000/api/user/update>

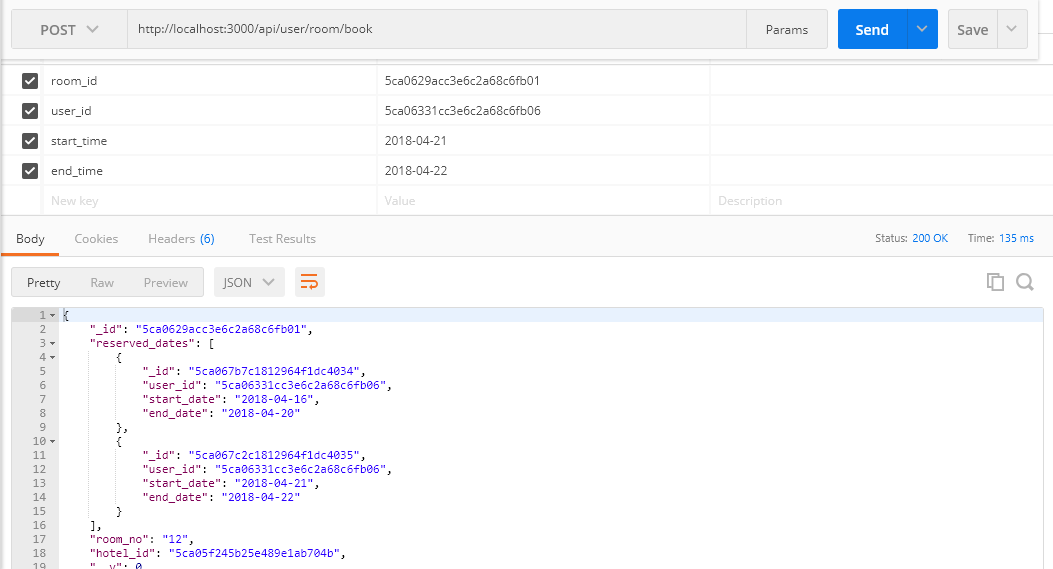


1. <http://localhost:3000/api/user/delete/<user-id>>



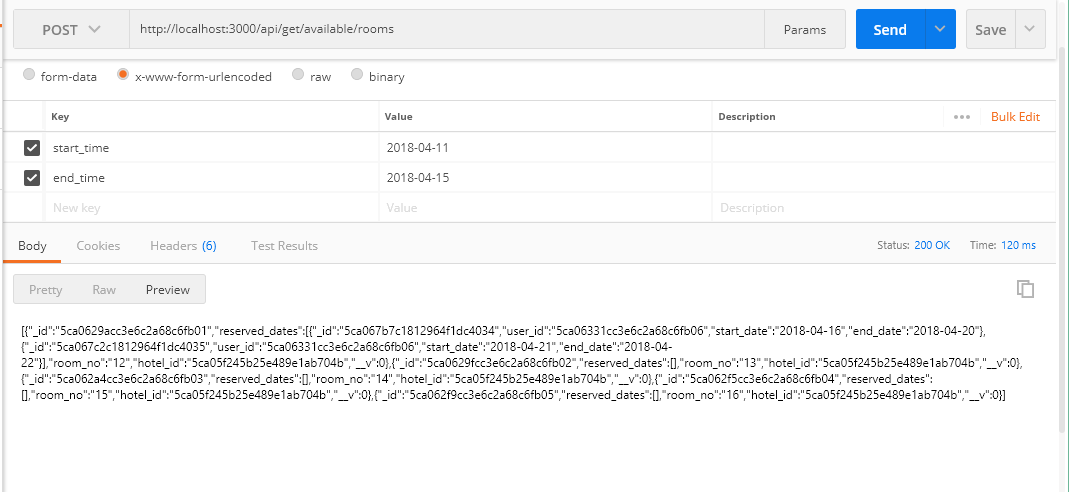
1. Booking room for a specific date

<http://localhost:3000/api/user/room/book>



1. Check available rooms for specific date

<http://localhost:3000/api/get/available/rooms>



**Please note the parameters to paas and the setting to select, is present in the screenshot.**

**Unit Testing of the code:**

Unique key constraint added on the model of Hotel,Room and Users therefore to avoid duplication of names

**In Terms of scaling the application**

The nodejs code can be built using serverless framework by deploying the code on AWS lambda or google cloud functions

The benefit is that since AWS lambda is autoscalable therefore we don’t need to worry about the load also the REST API can be deployed on AWS Lamda and also integrating it with API Gateway.

Another way is creating a microservices based architecture and deploying the code either on Kubernetes cluster or AWS lambda.

We can use Serverless Framework for integrating the code with AWS lambda and API Gateway using cloud formation stack.

I have deployed a simple nodejs lambda function on AWS , the link is below:-

<https://lqjq9dl9ak.execute-api.ap-south-1.amazonaws.com/dev/peopleinteractive>

Even though the above url is a simple REST GET API but since it is built on AWS lambda it can handle any amount of traffic spikes with ease **(the traffic could also be in millions)**