Backend Intern Hiring Task

Submission Date - **11:59 pm IST on Tuesday 9th April, 2024**

Time to Learn and Solve - **4-6 hours**

Problem Statement

You have been given a task to build a Library Management System, specifically the backend layer of the application. You are supposed to build the API Layer of this system, as mentioned in the requirements below.

Tech Stack

- Language - Python
- Framework - FastAPI
- Database - MongoDB (Use M0 Free Cluster of MongoDB Atlas)

You are free to use the PyMongo or Motor package to connect to MongoDB from your FastAPI application.

Requirements

You need to build APIs in **FastAPI** for the Library Management System, using MongoDB as the database. Make sure you are not connecting to local MongoDB and using MongoDB Atlas M0 cluster (free forever)

The APIs need to follow this specification **exactly**. That means you need to build all APIs mentioned in the spec document, as well as have the same exact request and response structure.

Make sure -

- You **read the description** of each API, request body, parameters as well as response body.
- You are sending the correct status code in the response (200, 201, 204, etc)
- You are sending back **id** and not **_id**.
- You have the correct endpoint defined
- You are checking which field is **required** and which is not, by clicking on the ***Schema*** button on top of the given sample request body or response body.

Deployment Criteria

You need to deploy your application to some Cloud or Deployment platform so that we can access your application in a running mode. Some options you can explore -

1. Deploy to AWS on EC2 machine (use free tier)

2. Deploy to Coherence (free plan)
3. Deploy to Render App (free plan)
4. Deploy to GCP / Digital Ocean / Azure (use free tier)
5. Deploy anywhere else which can host your python application

You are free to use Docker, if you want to deploy as a Docker container.

Judging Criteria

You will be judged on the correctness of your solution, by testing your solution in a Live Running environment. We will check -

- Correct data storage of students (in your MongoDB Database)
- Correct queries to return or update students in the database.
- No errors being raised by the system.
- Code quality, variable naming and structuring

Using Base URL

You need to submit the Base URL of your application in the submission form. For example, if you host your app on an EC2 server having IP address 12.34.45.56 and your app is running on port 8000 then submit **http://12.34.45.56:8000**

Similarly if you have deployed to render or other platforms, you might have a sample base URL – **https://example.com/api/<endpoints>**

Our system will **automatically append** the endpoints to your base url - such as **/students** and then make the API call to your system.

How to submit

Once done, make sure your application is deployed and you have a base URL ready, and your code uploaded to Github. Make sure the github repo is public or access is given to username: **shreybatra**

You can then go ahead and submit your application via this form, **on or before 11:59 PM IST on 9th April, 2024**