

Assignment: Flask Application for CRUD operations on MongoDB

You need to develop a Flask application that performs CRUD (Create, Read, Update, Delete) operations on a MongoDB database for a User resource using a REST API. The REST API endpoints should be accessible via HTTP requests and tested using Postman.

Requirements

- The application **must** be developed using Flask and MongoDB.
- The application should provide REST API endpoints for CRUD operations on a User resource.
- The User resource should have the following fields:
 - id (a unique identifier for the user)
 - name (the name of the user)
 - email (the email address of the user)
 - password (the password of the user)
- The application should connect to a MongoDB database.
- The application should provide the following REST API endpoints:
 - GET /users - Returns a list of all users.
 - GET /users/<id> - Returns the user with the specified ID.
 - POST /users - Creates a new user with the specified data.
 - PUT /users/<id> - Updates the user with the specified ID with the new data.
 - DELETE /users/<id> - Deletes the user with the specified ID.
- Use the most suitable libraries to complete the assignment. Think beyond the simplest solution. Go for the best solution.
- The application is fairly simple, what we are trying to see is how well you know Flask and libraries around it to achieve this in the most scalable way possible.
- High emphasis will be placed on how the code is structured.
- We will judge you on the beauty of system design. A plain solution to the assignment will not take you further.
- **Using Docker is mandatory**

Testing

1. Open Postman and create a new HTTP request for each of the REST API endpoints.
2. Send requests to the endpoints to test the CRUD operations on the User resource.
3. Verify that the responses are correct and the database is being updated correctly.

Submission

1. Submit the complete code for the Flask application and any additional files needed.
 2. Include a README file with instructions on how to set up and run the application.
 3. Push the code to your GitHub repo.
 4. Share the link to GitHub repo.
-

Any assignment submitted which reeks of ChatGPT will be immediately rejected. Since, there are hundreds of submissions it is not very difficult to determine which are the ones solved by AI.

Good luck with your assignment! Let me know if you have any questions.