Junior Software Developer Take-Home Test

Overview

In this take-home test, you will be building a simple document management dashboard application. The dashboard will display user information, overall document statistics, and a list of individual documents with their statuses. Users will be able to view and update document statuses.

The primary goals of this test are to assess your skills in:

- Developing a functional web application using Python and JavaScript.
- Creating RESTful APIs and integrating them with a front-end.
- Implementing basic containerization using Docker (optional).
- Writing and running tests for your code.
- Following best practices in documentation and code organization.

Instructions

1. Create a new GitHub repository for this project.

2. Set up a mock RESTful API using Python and Flask:

    - Create a simple Flask application that serves the following endpoints:
        - GET /user: Returns user information
        - GET /stats: Returns overall document statistics
        - GET /documents: Returns the list of documents
        - PUT /documents/{id}: Updates a document's status (Valid statuses: In Draft, In Review, Pending Approval, Complete)

3. Create a front-end application using JavaScript (vanilla JS or a framework of your choice):

    - Design and implement a simple dashboard page that displays:
        - User information (name, email, role)
        - Overall document stats (total documents, documents by status)
        - A list of documents showing document name, status, and last edited date
    - Implement functionality to update a document's status

4. Integrate the front-end with the back-end API:

    - Use JS to make HTTP requests to the appropriate API endpoints.
    - Implement error handling for API requests.

5. Implement testing:

    - Write tests for the back-end API endpoints.
    - Write front-end tests covering key functionality (e.g., rendering components, updating document status).

6. Containerize the application (optional):

- Create a Dockerfile for the back-end service.

7. Push your code to the GitHub repository and provide us with the link at devtest@adeptforcegroup.com

Provided JSON Data

Use this data to populate your API (no database required):


```
{
  "user": {
    "id": 1,
    "name": "John Doe",
    "email": "john.doe@example.com",
    "role": "Admin"
  },
  "stats": {
    "totalDocuments": 5,
    "inDraft": 1,
    "inReview": 1,
    "pendingApproval": 1,
    "complete": 2
  },
  "documents": [
    {
      "id": 1,
      "name": "Project Proposal",
      "status": "In Review",
      "lastEdited": "2024-06-10"
    },
    {
      "id": 2,
      "name": "Quarterly Report",
      "status": "Pending Approval",
      "lastEdited": "2024-06-08"
    },
    {
      "id": 3,
      "name": "Marketing Plan",
      "status": "In Draft",
      "lastEdited": "2024-06-12"
    },
    {
      "id": 4,
      "name": "Development Approach",
      "status": "Complete",
      "lastEdited": "2024-06-08"
    },
    {
      "id": 5,
      "name": "Personnel Overview",
      "status": "Complete",
```

```
      "lastEdited": "2024-06-04"
    }
  ]
}
```

Requirements

- Use Python (Flask) for the back-end and JavaScript for the front-end.
- Implement a RESTful API to serve and update document data.
- Implement error handling.
- Create a simple, functional user interface to display and interact with the data.
- Containerize the application using Docker (optional).
- Write tests for the front-end and back-end.

Evaluation Criteria

- Proficiency in Python and JavaScript programming.
- Implementation of RESTful API design principles.
- Robust handling of errors.
- Front-end functionality and user experience.
- Proper use of Docker for containerization (optional).
- Code quality, organization, and adherence to best practices.
- Testing implementation and coverage.

Timeline

Please submit your completed project by the provided due date. If you have any questions or need further clarification, please don't hesitate to reach out to devtest@adeptforcegroup.com. We look forward to reviewing your submission. Good luck!