

Appgain.io Backend Evaluation Task

Description

Your task is to develop a simplified Order Processing System for an online store. The system should handle order processing, including stock validation, payment processing (integrated with a mock payment gateway), and sending order confirmation emails to customers.

Requirements

1. **Stock Management:** Implement functionality to validate the availability of products in the store's inventory when placing an order. Ensure that the system updates the stock count accordingly after each successful order.
2. **Payment Processing:** Integrate with a mock payment gateway to simulate payment processing. Upon successful payment, mark the order as paid and proceed with order fulfillment.
3. **Order Confirmation Emails:** Develop a feature to send order confirmation emails to customers after a successful purchase. The email should include details such as the order ID, purchased items, and total amount.
4. **Error Handling:** Implement error handling to gracefully manage any issues that may occur during the order processing flow, such as stock unavailability or payment failures.
5. **Documentation:** Provide clear and concise documentation on how to set up and run the Order Processing System.
6. **Containerization:** Dockerize the Order Processing System by creating a Dockerfile to package the application into a container. Ensure that the Dockerfile is included in the repository.
7. **Repository Integration:** Push the built Docker image to a container registry/repository of your choice (e.g., Docker Hub, GitHub Container Registry).

Bonus

- **User Authentication:** Implement basic user authentication to ensure that only registered users can place orders.
- **Testing:** Write unit tests to verify the functionality and reliability of the order processing system.
- **Logging:** Implement logging to record important events and errors that occur during order processing.
- **Customization:** Allow customization of email templates for order confirmation emails.
- **Deployment:** Deploy the Order Processing System to a cloud platform such as Heroku or AWS.

Submission

- Develop your Order Processing System according to the requirements.
- Once completed, host your code on a public repository platform such as GitHub, GitLab, or Bitbucket.

- Ensure that your repository is publicly accessible.
- Complete the README file in your repository, providing instructions for project setup and usage.
- Include any additional configuration files necessary for running the application, such as environment variables or configuration files.
- Ensure that all necessary files, including the Dockerfile and any scripts, are included in the repository.
- Push the built Docker image to a container registry/repository.
- Submit the public repository link along with your application.

Evaluation

- **Functionality:** Does the Order Processing System meet the specified requirements?
- **Code Quality:** Is the code well-structured, readable, and maintainable?
- **Documentation:** Is the setup and usage of the Order Processing System clearly documented?
- **Bonus Points:** Are any bonus requirements implemented, and how effectively?
- **Error Handling:** How effectively are errors handled within the application?
- **Efficiency:** Is the application designed to handle orders efficiently and scale if needed?

Deadline

Please make sure to submit your task within maximum 4 days from the date you received the task.

Note: This evaluation task is designed to assess your skills in backend development using Python, particularly in handling order processing flows. Feel free to reach out if you have any questions or need clarification on any aspect of the task. Good luck!

