

Test Task for Python Developer: Cyberpunk Inventory Management System

Overview:

Your task is to develop a simple inventory management system for a cyberpunk-themed game using FastAPI, SQLAlchemy, and PostgreSQL. The system will manage items that players can acquire in the game. These items could range from cybernetic enhancements to weapons and gadgets. The API will provide basic CRUD (Create, Read, Update, Delete) functionality for these items.

Requirements:

1. FastAPI Application Setup:

- Initialize a new FastAPI project.
- Ensure the project is properly structured for scalability.

2. Database Model with SQLAlchemy:

- Use PostgreSQL as the database backend.
- Design a database schema for the inventory items. Each item should have the following fields:

- `id` (Primary Key)
- `name` (String, unique)
- `description` (Text)
- `category` (String, e.g., "Weapon", "Cybernetic", "Gadget")
- `quantity` (Integer)
- `price` (Float, representing the item's value in-game)
- Use SQLAlchemy ORM for database interactions.

3. CRUD Operations:

- **Create:** Implement an endpoint to add new items to the inventory.
- **Read:** Implement endpoints to:
 - Retrieve a list of all items, with support for pagination.
 - Retrieve a single item by its ID.
- **Update:** Implement an endpoint to update the details of an existing item.
- **Delete:** Implement an endpoint to remove an item from the inventory.

4. Validation and Serialization:

- Use Pydantic models for request validation and response serialization.
- Ensure that all CRUD operations validate incoming data according to the database schema constraints.

5. Testing:

- Write pytest for all endpoints, ensuring that each operation functions correctly.

Bonus (Optional):

- Implement authentication and authorization for the API, restricting CRUD operations to authenticated users only.
- Dockerize the FastAPI application for easy deployment using docker compose.

Submission Guidelines:

- Push your code to a GitHub repository and provide the link for review.
- Include a README file with:
 - Instructions on how to set up and run your project.
 - A brief explanation of your project's structure and any special features you implemented.
- Ensure your application is properly commented on, especially complex or critical sections of the code.

Evaluation Criteria:

- Correct implementation of FastAPI, SQLAlchemy, and PostgreSQL.
- Code quality and project structure.
- Implementation of CRUD functionality.
- Proper validation and error handling.
- Presence and quality of tests.
- Documentation and ease of use.

Good luck, and we look forward to seeing your cyberpunk-themed inventory management system!