

API Development Test with Database Integration

Objective: Develop a RESTful API for a payment system that interacts with a database to manage user credits. The API will handle credit top-ups, deductions, and balance inquiries.

Requirements:

1. API Endpoints

- **Credit Top-Up Endpoint**

- **Path:** /api/credits/top-up
- **Method:** POST
- **Input:** UserID, Amount
- **Functionality:** Record a credit top-up transaction for the user.
- **Response:** Confirmation of transaction, including a timestamp.

- **Credit Deduction Endpoint**

- **Path:** /api/credits/deduct
- **Method:** POST
- **Input:** UserID, Amount
- **Functionality:** Record a credit deduction transaction for the user.
- **Response:** Confirmation of transaction, including a timestamp.

- **Check Balance Endpoint**

- **Path:** /api/credits/balance
- **Method:** GET
- **Input:** UserID
- **Functionality:** Calculate and return the current credit balance by summing all top-ups and subtracting all deductions.
- **Response:** Current credit balance.

2. Database Interactions

- Design a simple database schema necessary for these operations, focusing on Users and Transactions.
- Transactions should include a timestamp.
- Balance should be calculated dynamically from the transaction records.

3. General Instructions

- **Data Validation:** Implement basic input validation and error handling.
- **Mock Database:** Use a mock database or pseudo-code for database interactions.
- **Code Quality:** Write clean, readable, and maintainable code.

- **Documentation:** Briefly document the purpose and usage of each endpoint.
- **Testing:** Provide basic instructions on how to test these endpoints.

Evaluation Criteria:

- **Functionality:** Correct implementation of the specified tasks.
- **Database Schema Design:** Efficiency and appropriateness of the mock database schema.
- **Code Organization and Quality:** Clarity and maintainability of the code.
- **Error Handling:** Robustness in managing edge cases and incorrect inputs.
- **Documentation and Testing:** Completeness and clarity of documentation and testing instructions.