**1. Introduction**

This document provides an overview of a banking application built using a microservices architecture. The application supports account creation, balance enquiry, deposits, withdrawals, transaction history, and email notifications. The services communicate via REST APIs and are designed for scalability, security, and observability.

**2. Architecture Overview**

**2.1. Microservices Overview**

* **Account Service**: Handles account-related operations such as account creation, balance enquiry, deposits, and withdrawals.
* **Transaction Service**: Manages transactions, including recording deposits and withdrawals, and provides transaction history.
* **Notification Service**: Sends email notifications for account transactions.

**2.2. Interaction Diagram**

+-----------------+ +-------------------+ +----------------------+

| Account Service |<--->| Transaction Service|<--->| Notification Service |

+-----------------+ +-------------------+ +----------------------+

**3. Design Decisions**

**3.1. Microservices Architecture**

* **Rationale**: To achieve modularity, scalability, and ease of maintenance.
* **Benefits**: Each service can be developed, deployed, and scaled independently.

**3.2. Technology Stack**

* **Spring Boot**: Chosen for rapid development and comprehensive support for microservices.
* **JPA (Hibernate)**: For database interactions to simplify ORM.
* **RestTemplate**: For inter-service communication.

**3.3. Security**

* **OAuth2**: Used to secure REST endpoints.
* **Encryption and Hashing**: Passwords are hashed using BCrypt before storing in the database.

**4. Testing**

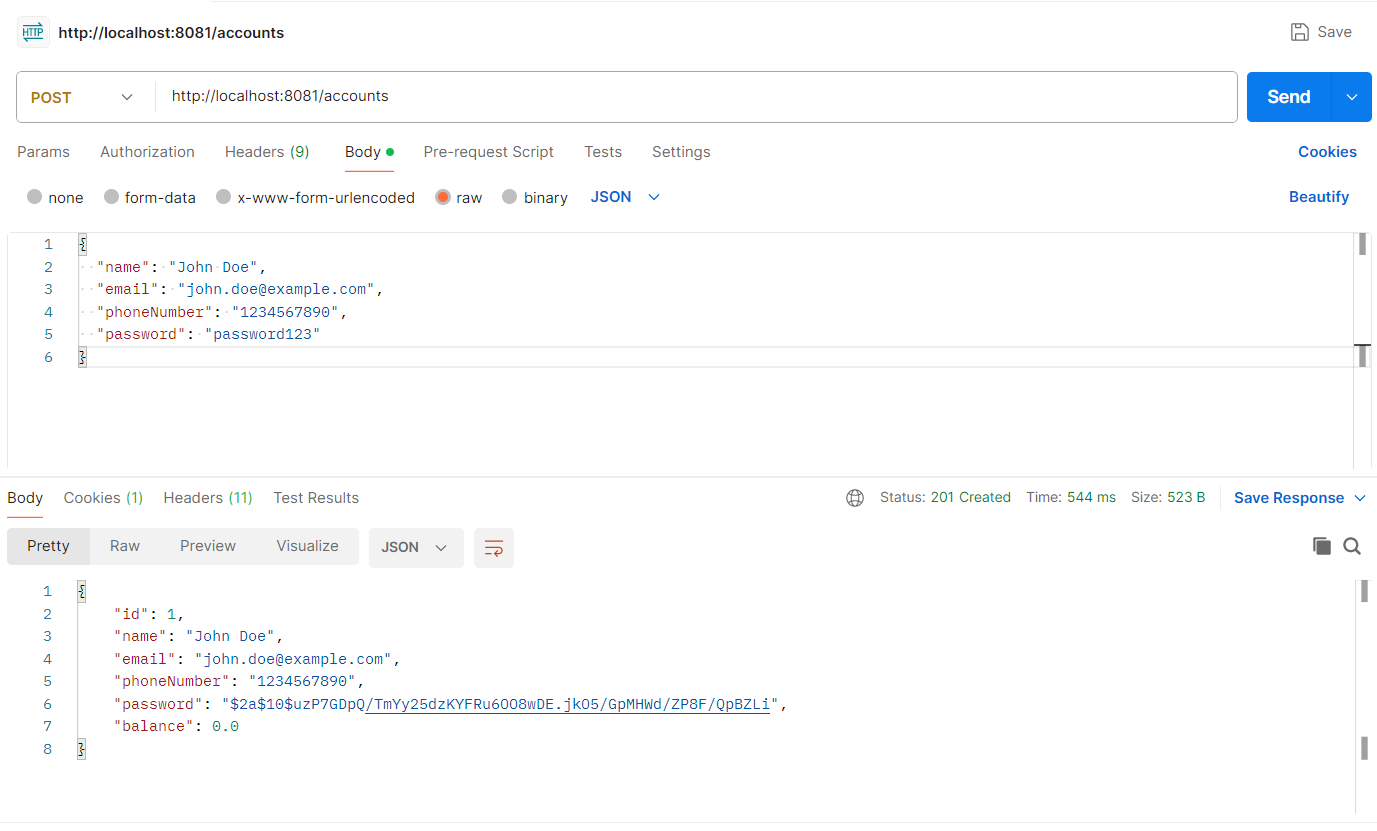
**4.1. Unit Tests**

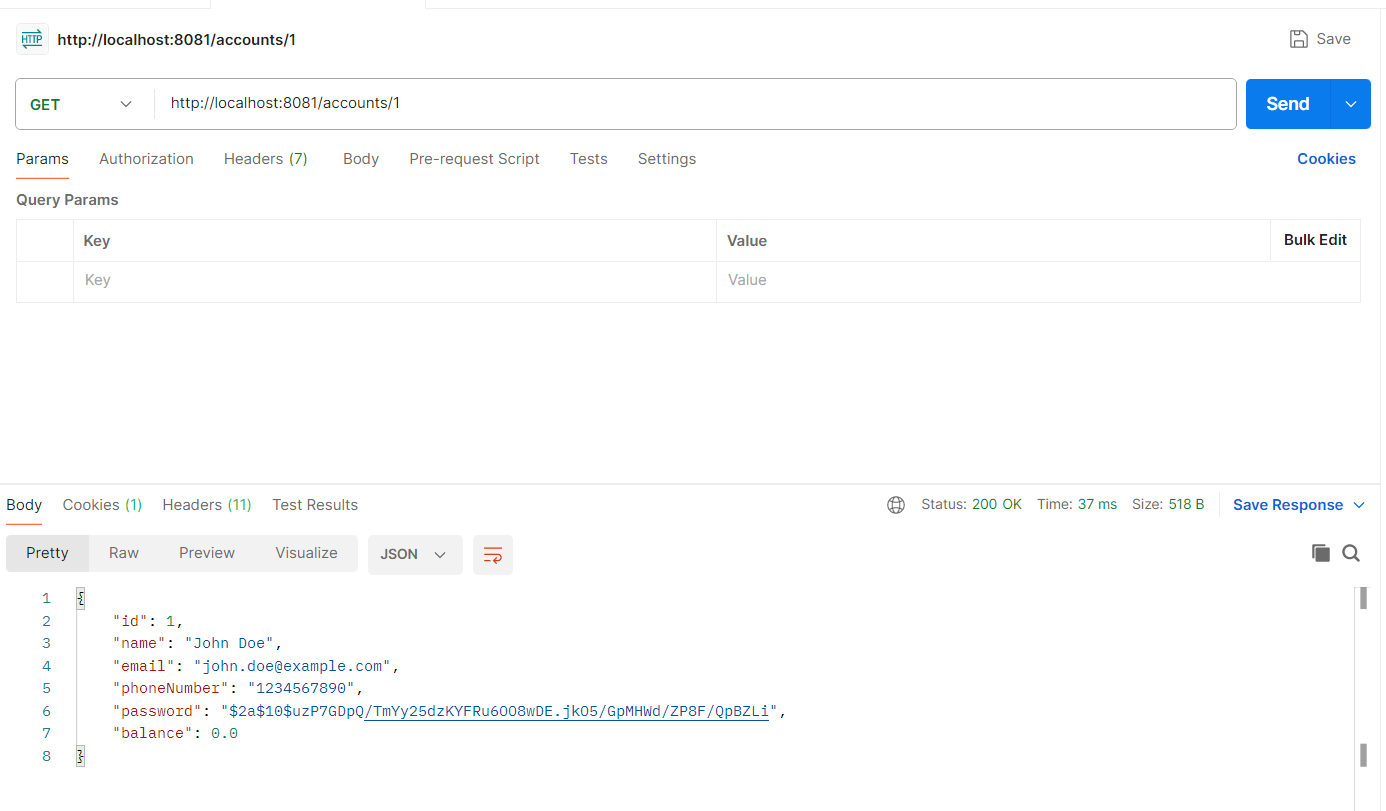
* **Framework**: JUnit
* **Coverage**: Controllers, Services, and Repositories

**5. Results**

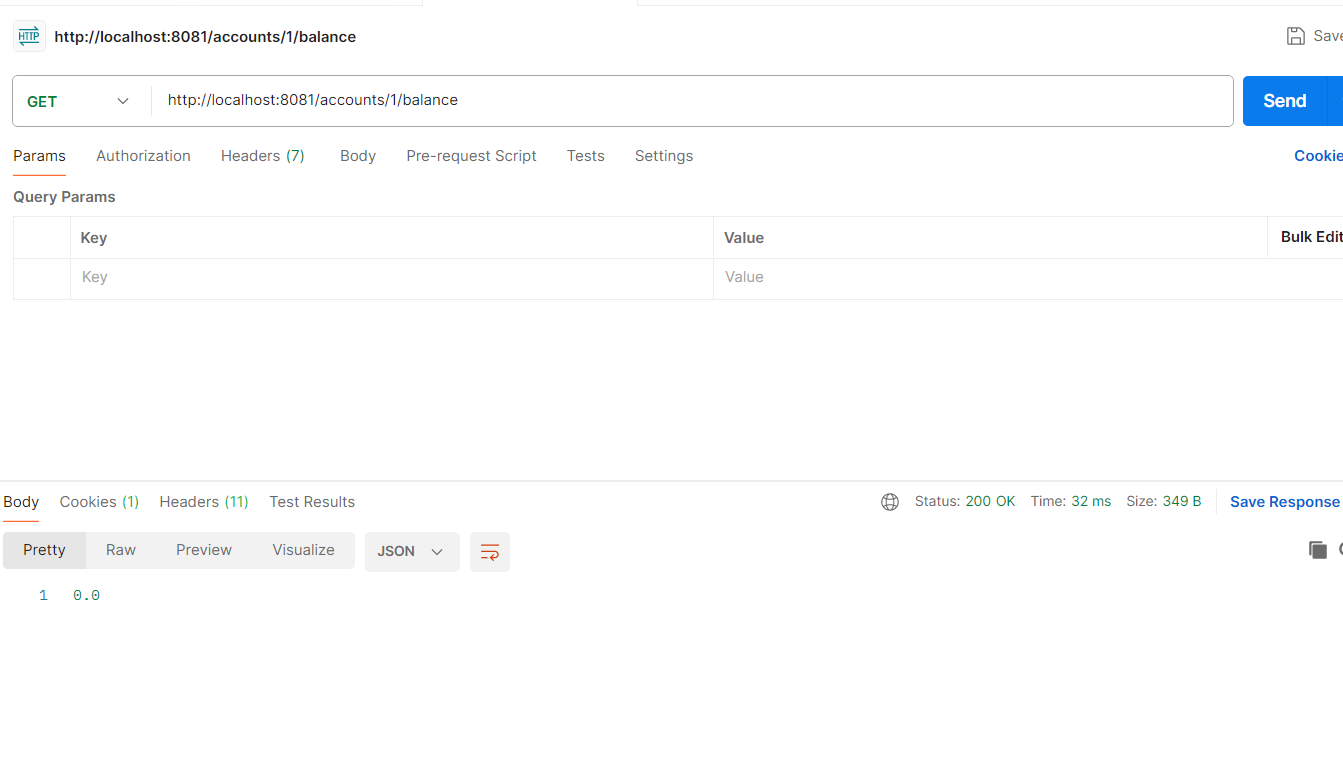
Account Microservice

Account creation

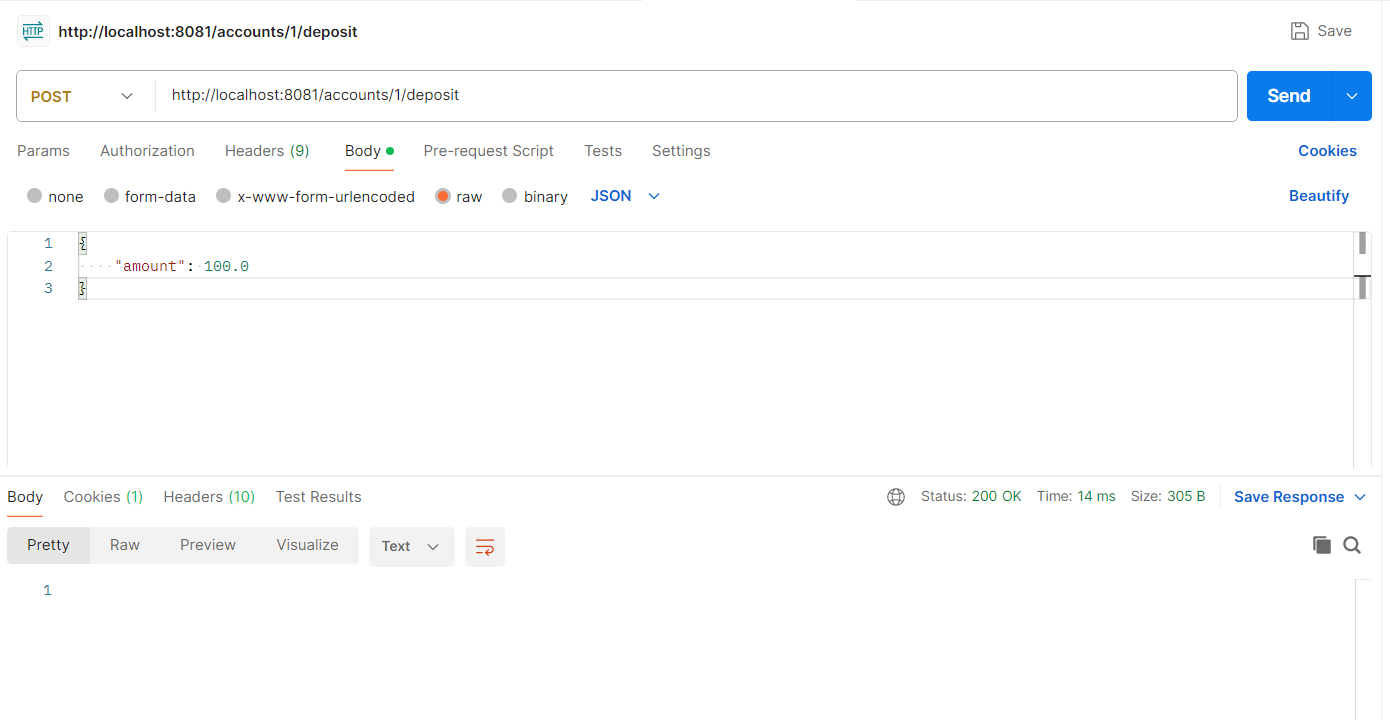


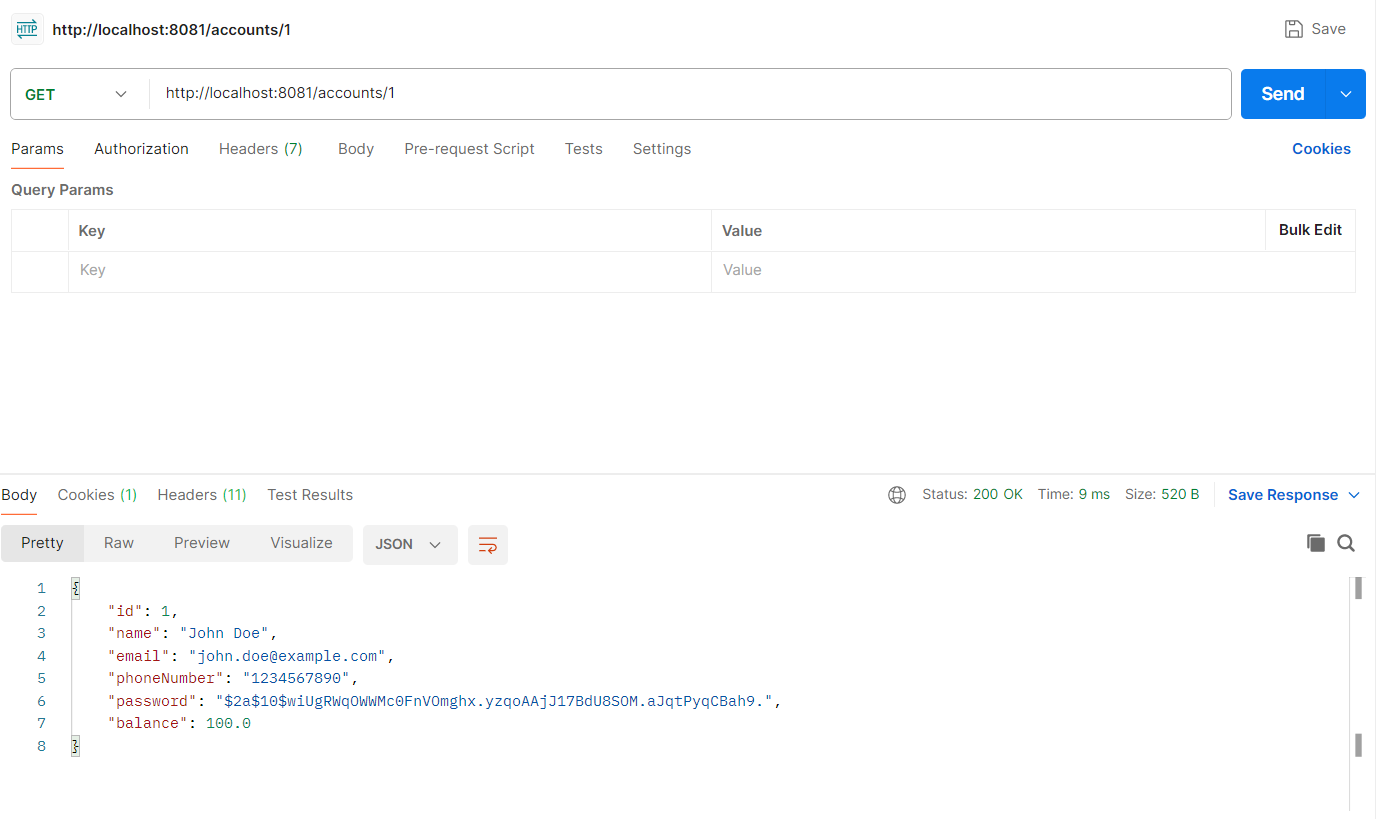


Balance Enquiry

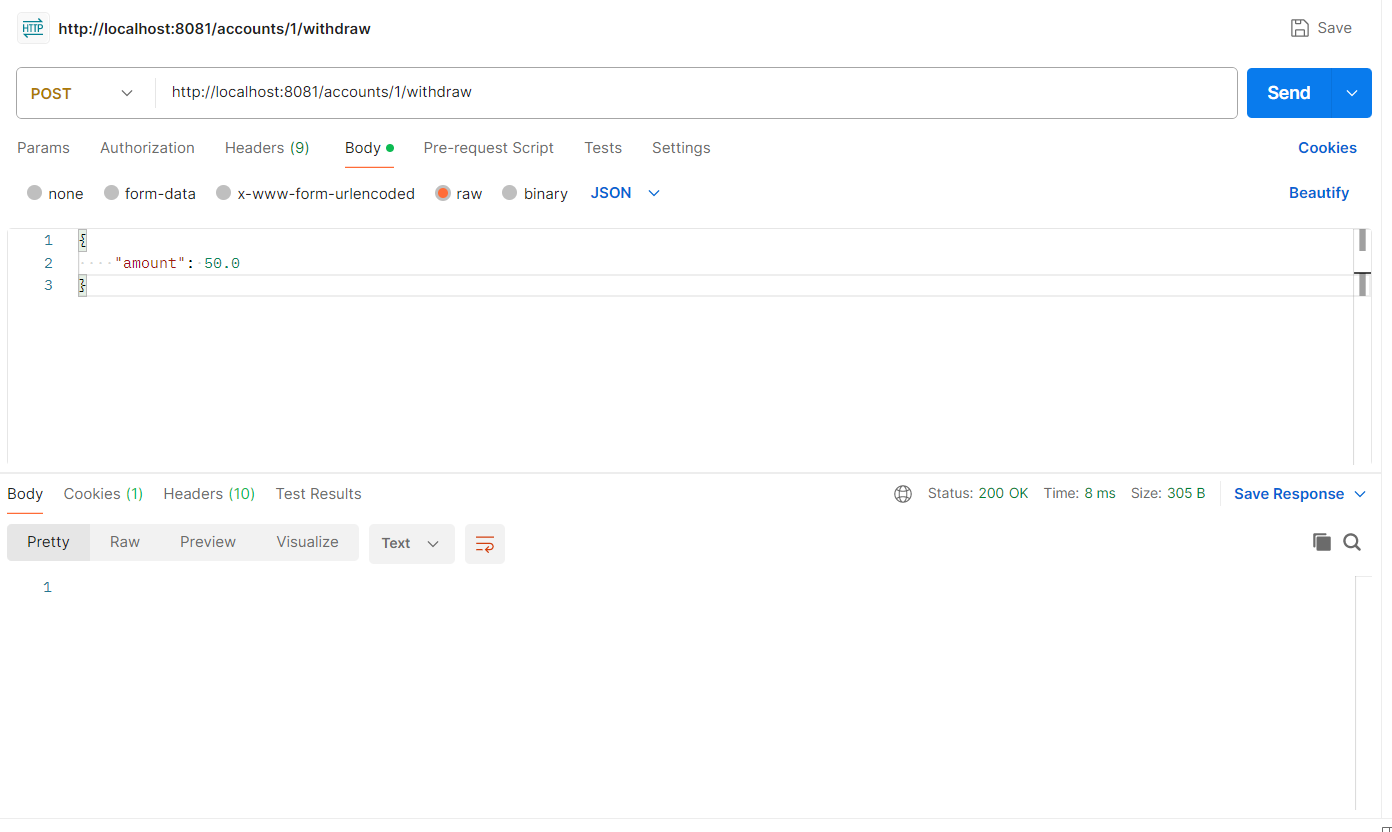


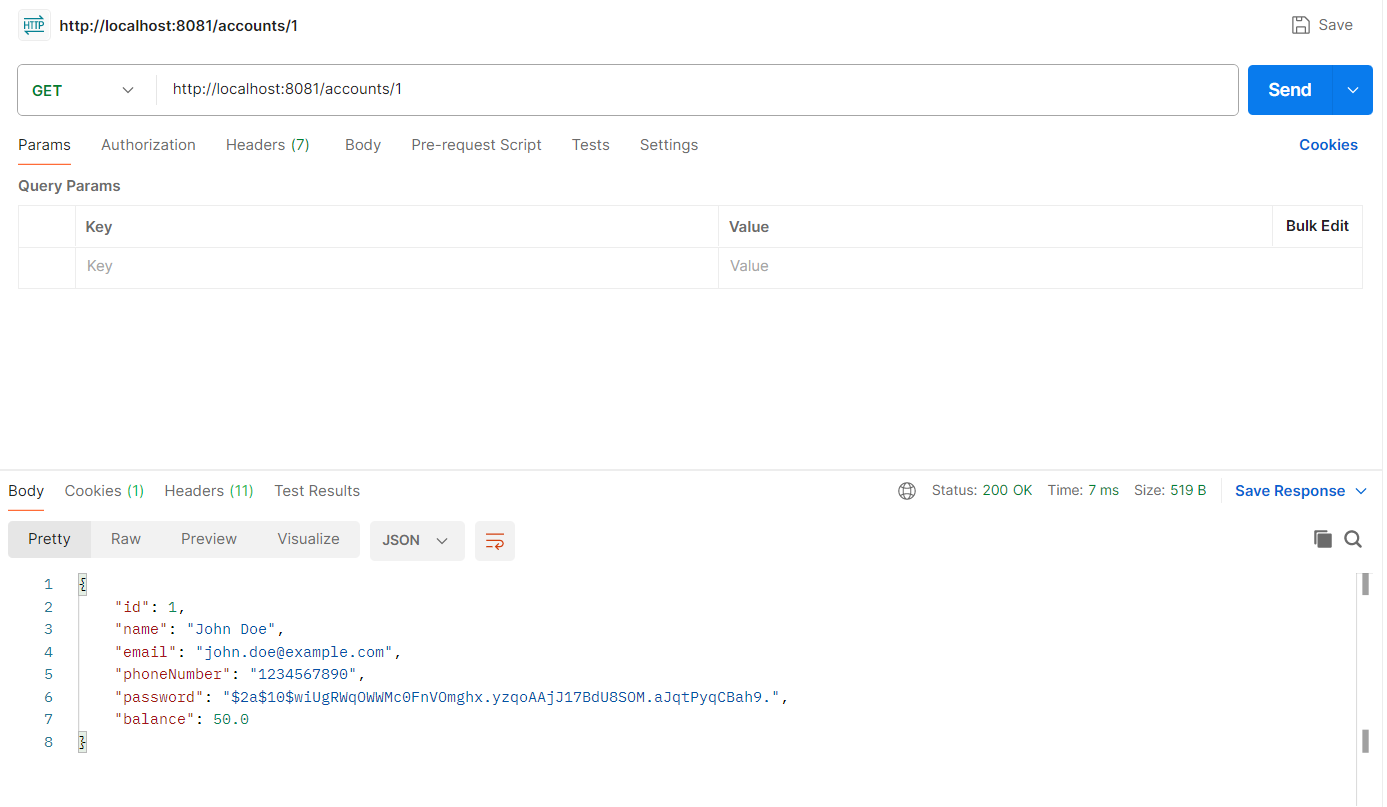
Deposit





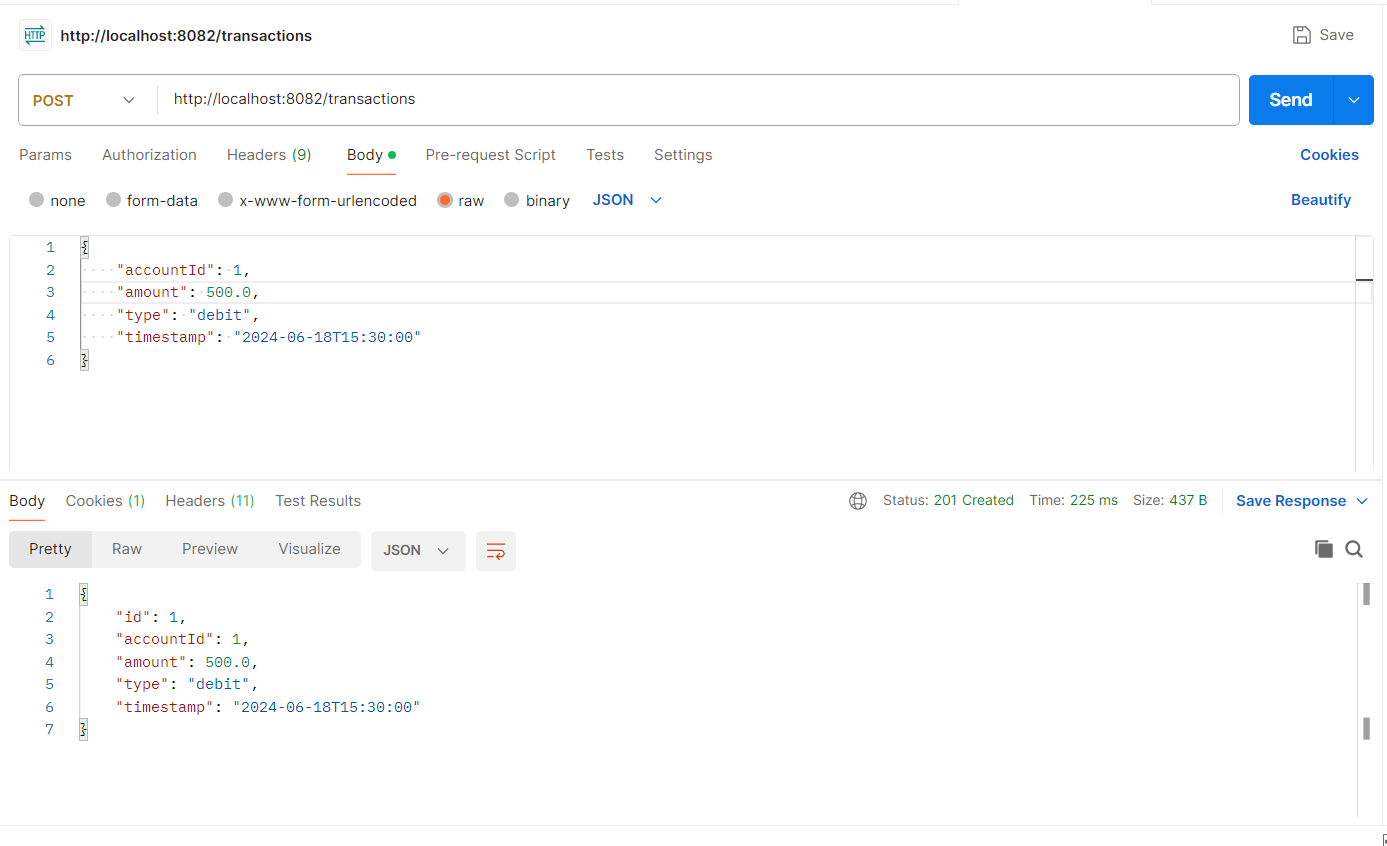
Withdraw





Transaction Microservice

Creating Transaction



View Transaction

