Wallet Service Assignment

Objectives

We are seeking to create a wallet service that manages users' money. This service will support operations for depositing, withdrawing, and transferring funds between users.

Although this is a simplified project to respect your time, please approach it as you would a real production environment.

Instructions:

- **Scope**: The project should be completable within 6 to 8 hours. This is a guideline, not a strict deadline.

- **Professionalism**: Implement the project with the same care and attention to detail as you would in a professional setting.

  In this code assessment, if you deal with any requirements, behaviors, or implementation details that are not clearly defined, please make sure to document the assumptions you made to proceed with your solution. Clearly state each assumption and provide a brief rationale explaining your reasoning. This approach demonstrates your analytical skills and ensures that your solution can be reviewed and validated in the proper context.

- **Time Tracking**: Upon completion, please provide a rough estimate of the time you invested in the project.

- **Technology**: The project should be implemented in Java.

Functional Requirements

- **Create Wallet:** Allow the creation of wallets for users.

- **Retrieve Balance:** Retrieve the current balance of a user's wallet.

- **Retrieve Historical Balance:** Retrieve the balance of a user's wallet at a specific point in the past.

- **Deposit Funds:** Enable users to deposit money into their wallets.

- **Withdraw Funds:** Enable users to withdraw money from their wallets.

- **Transfer Funds:** Facilitate the transfer of money between user wallets.

Non-functional requirements

This service is mission-critical. Any downtime would significantly impair the platform's ability to perform most operations.

Given that the service handles monetary transactions, it is imperative to ensure full traceability of all operations to facilitate auditing of wallet balances.

Technical Requirements

All technical choices, including the selection of REST/HTTP frameworks or libraries, are at your discretion.

Deliverables

A GitHub repository containing:

- The implementation of the microservice.

- Instructions on how to install, test, and run the service.

- A written explanation of the design choices, detailing how the implementation meets both functional and non-functional requirements.

- A written explanation of any compromises or trade-offs made due to time constraints.

All documentation should be in English.