ATM Microservices Documentation

Account Microservice

The Account Microservice is responsible for managing customer accounts. It provides functionality to retrieve the account balance and update the account balance.

API functionalities:

- Get Account Balance:
  GET /atm/account/{accountNumber}/balance
  Description: Retrieves the account balance for the specified account number.

- Update Account Balance
  POST /atm/account/{accountNumber}/balance
  Description: Updates the account balance for the specified account number.
  Request Body: JSON object containing the new account balance.
  Response: 200 OK if the balance was successfully updated.

Transaction Microservice

The Transaction Microservice is responsible for processing transactions. It provides functionality for depositing and withdrawing funds from customer accounts.

API functionalities

- Deposit:
  POST /api/transactions/deposit
  Description: Deposits the specified amount into the account with the given account number.
  Request Body: JSON object containing the account number and deposit amount.
  Response: JSON object containing the updated account balance.

- Withdrawal:
  POST /api/transactions/withdrawal
  Description: Withdraws the specified amount from the account with the given account number if the balance is sufficient.
  Request Body: JSON object containing the account number and withdrawal amount.
  Response: JSON object containing the updated account balance if the withdrawal is successful, or an error message if there are insufficient funds.

  Note: The Transaction Microservice maintains an inventory of 5 bills each for denominations 200, 100, 50, and 20.

Running the Microservices Locally:

- Set up the database: Make sure you have a SQL Server instance available.
- Update the connection string in the code to match your SQL Server configuration.

- Run the application: Start the application, and the microservices will be hosted on a local development server.

Testing the Microservices:

- Use tools like Postman or cURL to send HTTP requests to the provided API endpoints.
- Replace placeholders like {accountNumber} in the URLs with the actual account number you want to test.
- Include the necessary request parameters or request bodies as described in the API documentation.
- Send the request and examine the response to verify the functionality of the microservices.

Swagger support: The microservices are equipped with Swagger, which provides a user-friendly interface for exploring and testing the APIs.