**Banking Application – REST API Documentation**

**🔐 Authentication APIs**

**1. Register User**

* **POST** /auth/register
* **Description**: Registers a new user with username, password, and role.
* **Request Body**:

Json Request Payload:

|  |
| --- |
| {  "username": "john",  "password": "password123",  "role": "CUSTOMER" // or "ADMIN"  } |

* **Response**:

201 Created

**2. Login and Generate JWT Token**

* **POST** /auth/login
* **Description**: Authenticates a user and returns a JWT token.
* **Request Body**:

Json Request Payload:

|  |
| --- |
| {  "username": "john",  "password": "password123"  } |

* **Response**:

Json Response :

{

"token": "eyJhbGciOiJIUzUxMiJ9..."

}

**Notes**: Send this token in Authorization header as Bearer <token> for secured APIs.

**👤 Customer APIs**

**3. Create Customer**

* **POST** /customers
* **Role Required**: CUSTOMER or ADMIN
* **Request Body**:

Json Request Payload:

|  |
| --- |
| {  "name": "John Doe",  "email": "john@example.com"  } |

* **Response**: json

|  |
| --- |
| * { * "id": 1, * "name": "John Doe", * "email": "john@example.com" * } |

**4. Inquire Customer**

* **GET** /customers/{id}
* **Role Required**: CUSTOMER or ADMIN
* **Response**:

Json Request Payload :

{

"id": 1,

"name": "John Doe",

"email": "john@example.com"

}

**💰 Account APIs**

**5. Create Account**

* **POST** /accounts/create
* **Role Required**: ADMIN
* **Request Body**:

Json Request Payload

{

"customerId": 1,

"accountType": "SAVINGS"

}

* **Response**:

json

{

"accountNumber": "ACC123456",

"accountType": "SAVINGS",

"status": "ACTIVE",

"balance": 0.0

}

**6. Deposit Cash**

* **POST** /accounts/deposit
* **Role Required**: CUSTOMER
* **Request Body**:

Json Request payload

{

"accountNumber": "ACC123456",

"amount": 1000.0

}

* **Response**:

json

{

"accountNumber": "ACC123456",

"balance": 1000.0

}

**7. Withdraw Cash**

* **POST** /accounts/withdraw
* **Role Required**: CUSTOMER
* **Request Body**:

Json request payload

{

"accountNumber": "ACC123456",

"amount": 500.0

}

* **Response**:

json

{

"accountNumber": "ACC123456",

"balance": 500.0

}

**8. Close Account**

* **PUT** /accounts/close
* **Role Required**: ADMIN
* **Request Body**:

Json request payload

{

"accountNumber": "ACC123456"

}

* **Response**:

json

{

"accountNumber": "ACC123456",

"status": "CLOSED"

}

**9. Inquire Account**

* **GET** /accounts/{accountNumber}
* **Role Required**: CUSTOMER or ADMIN
* **Response**:

Json

{

"accountNumber": "ACC123456",

"accountType": "SAVINGS",

"status": "ACTIVE",

"balance": 500.0,

"customer": {

"id": 1,

"name": "John Doe"

}

}

**🔐 Security**

* JWT token required for all endpoints except /auth/\*\*.
* Role-based access:
  + CUSTOMER can **view/create** their own customer and perform deposits/withdrawals.
  + ADMIN can **create/close** accounts and access all customers/accounts.

**🗃️ Technologies Used**

* **Spring Boot 3**
* **Spring Security with JWT**
* **H2 Embedded Database**
* **Hibernate / Spring Data JPA**
* **Maven**
* **Lombok**
* **Postman** (for testing)
* **Spring Boot DevTools.**

**Note**: Sorry I don’t time to implements swagger.