**Create the Account Entity**

An **entity** represents a table in the database.

## ****Create the Account Repository****

A repository is responsible for interacting with the database.

## ****Create the Account Service (Business Logic)****

The service layer contains business logic for managing accounts.

## ****Create the Account Controller****

The **controller** exposes APIs that allow external clients (like frontend applications) to interact with the Account Service.

### ****Tables Needed****

1. **accounts** (Stores account details)
2. **transactions** (Records all transactions like deposit, withdrawal, transfer)

**Accounts Table**

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Data Type** | **Description** |
| id | BIGSERIAL (PK) | Unique ID for each account |
| user\_id | BIGINT | References the user who owns the account |
| account\_number | VARCHAR(20) UNIQUE | Unique account number |
| account\_type | VARCHAR(20) | Type of account (SAVINGS, CURRENT) |
| balance | DECIMAL(15,2) | Current account balance |
| created\_at | TIMESTAMP | Account creation timestamp |

|  |
| --- |
| CREATE TABLE accounts (  id SERIAL PRIMARY KEY,  user\_id BIGINT NOT NULL,  account\_number VARCHAR(20) UNIQUE NOT NULL,  account\_type VARCHAR(20) NOT NULL CHECK (account\_type IN ('SAVINGS', 'CURRENT')),  balance DECIMAL(15,2) NOT NULL DEFAULT 0.00,  created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP  ); |

**Transactions Table**

|  |  |  |
| --- | --- | --- |
| **Column Name** | **Data Type** | **Description** |
| id | BIGSERIAL (PK) | Unique ID for each transaction |
| account\_id | BIGINT (FK) | References accounts(id) |
| transaction\_type | VARCHAR(20) | DEPOSIT, WITHDRAW, TRANSFER |
| amount | DECIMAL(15,2) | Transaction amount |
| status | VARCHAR(20) | SUCCESS, FAILED |
| created\_at | TIMESTAMP | Transaction timest |

|  |
| --- |
| CREATE TABLE transactions (  id SERIAL PRIMARY KEY,  account\_id BIGINT NOT NULL REFERENCES accounts(id),  transaction\_type VARCHAR(20) NOT NULL CHECK (transaction\_type IN ('DEPOSIT', 'WITHDRAW', 'TRANSFER')),  amount DECIMAL(15,2) NOT NULL,  status VARCHAR(20) NOT NULL CHECK (status IN ('SUCCESS', 'FAILED')),  created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP  ); |

**APIs Needed**

Account APIs

|  |  |  |
| --- | --- | --- |
| **HTTP Method** | **Endpoint** | **Description** |
| POST | /accounts | Create a new bank account |
| GET | /accounts/{id} | Get account details by ID |
| GET | /accounts/user/{userId} | Get accounts for a specific user |

Transaction APIs

|  |  |  |
| --- | --- | --- |
| **HTTP Method** | **Endpoint** | **Description** |
| PUT | /accounts/{id}/deposit | Deposit money into an account |
| PUT | /accounts/{id}/withdraw | Withdraw money from an account |
| POST | /accounts/transfer | Transfer money between accounts |
| GET | /transactions/account/{id} | Get transaction history for an account |