**CONSOLE BASED BANKING APPLICATION**

**OBJECTIVE**:  
 To develop a console-based banking application with java by integrating it with the database using JDBC connectivity.

**Identification of Need:**

Banking is an essential thing, and building a banking application can make our transactions easier and then, easy and instant insight of transactions, account balance information can help us to be more aware of our finance status.

**Platform Specifications:**

**Hardware:**

* Intel core i3, i5, i7, i9 or MAC
* minimum 4GB RAM.

**Software:**

* JAVA
* MySQL
* JDBC.

**FUNCTIONAL REQUIREMENTS:**

1. User Account Management

* Create account
* Savings / current

1. Transaction Management

* Withdraw / Deposit
* Transaction between accounts

1. Transaction History

* Save a log of all transactions to a file.
* History of account activities

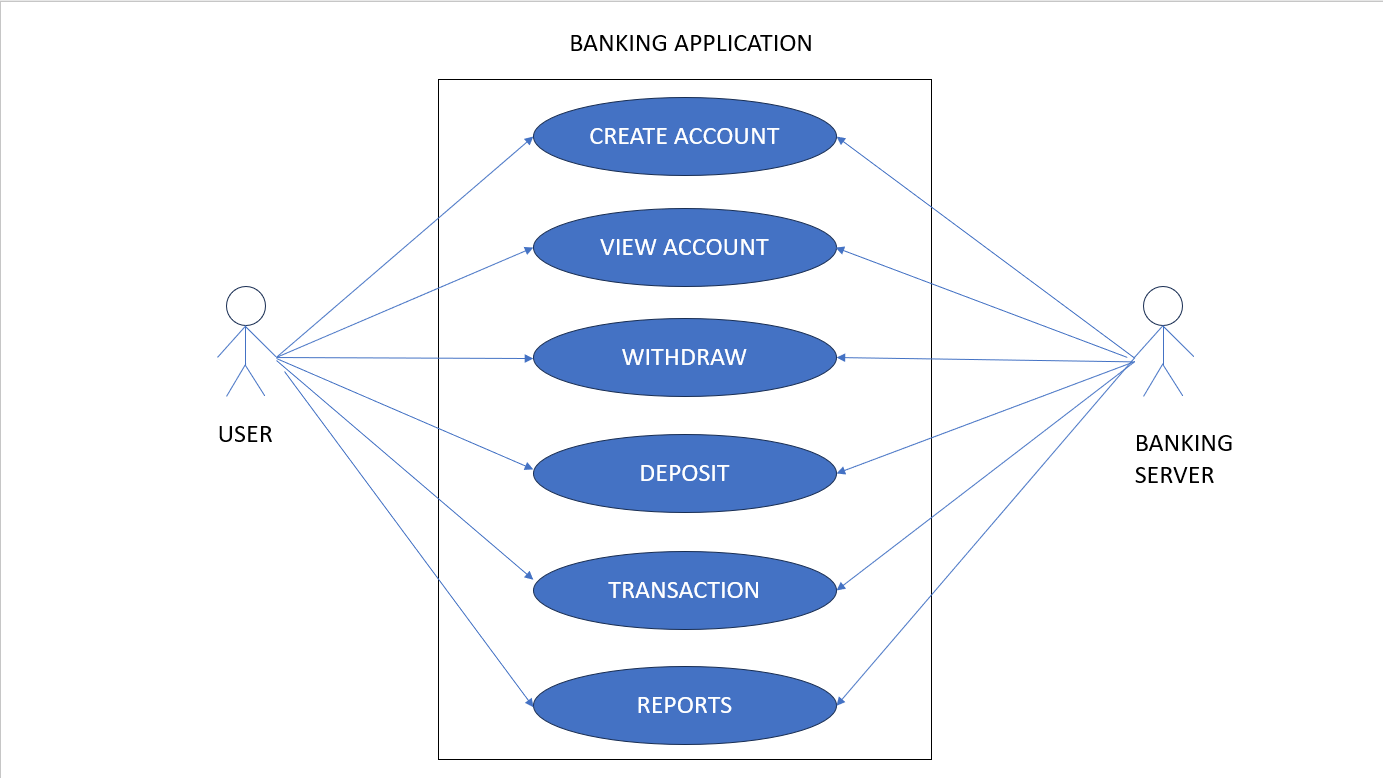
1. Database Operations

* CRUD operations
* Store account operations in a database

1. Reports

* Account details for each customer.

**USE CASE DIAGRAM:**



**SCHEMA DESIGN :**

**Bank:** Stores information about the bank.

Columns:

bank\_id INT UNIQUE (PRIMARY KEY)

bank\_name VARCHAR(100)

bank\_branch VARCHAR(100)

**Account:** Stores account details, associated with a particular bank.

Columns:

account\_id INT UNIQUE (PRIMARY KEY)

customer\_id INT

bank\_id INT (FOREIGN KEY)

account\_type VARCHAR (50)

**SavingsAccount and CurrentAccount**: Specialized table for different account types, inheriting from the **Account** table.

Columns (Savings Account):

account\_id INT (FOREIGN KEY)

Interest\_rate DECIMAL (5,2)

Columns (Current Account):

account\_id INT (FOREIGN KEY)

overdraft\_limit DECIMAL (5,2)

**Transaction:** Records all transactions (deposit, withdraw, transfer) Linked to accounts.

Columns:

transaction\_id INT (PRIMARY KEY)

account\_id INT (FOREIGN KEY)

transaction\_type VARCHAR(50)

amount DECIMAL(15,2)

transaction\_date TIMESTAMP

**DepositTransaction, WithdrawTransaction:** Specialized tables for different transaction types, inheriting from the transaction table

Columns (DepositTransaction):

transaction\_id INT (FOREIGN KEY)

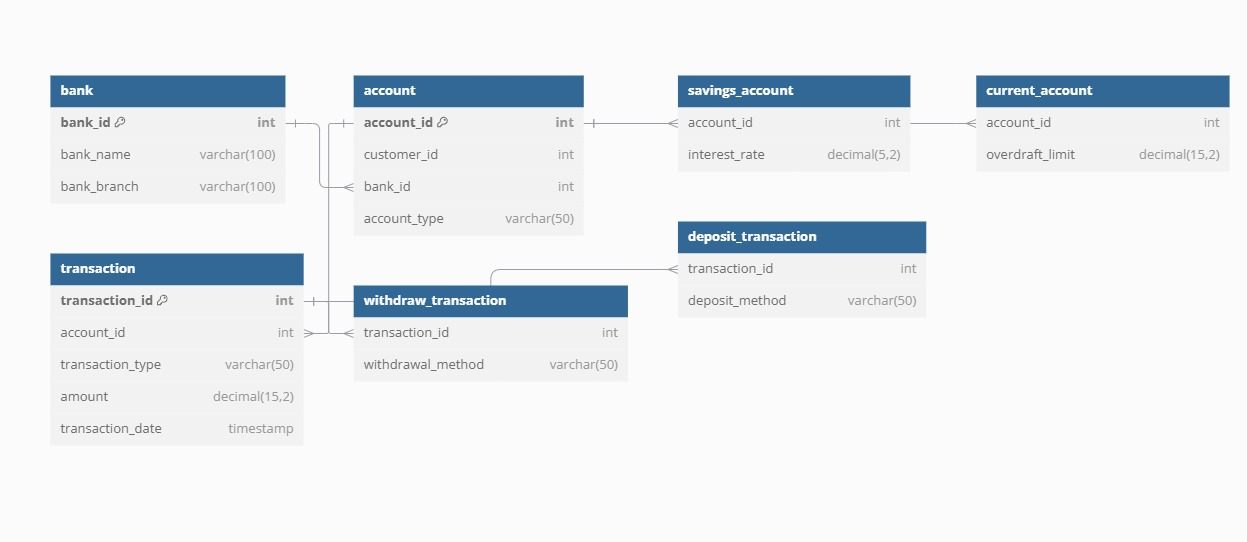
deposit\_method VARCHAR(50)

Columns (WithdrawTransaction):

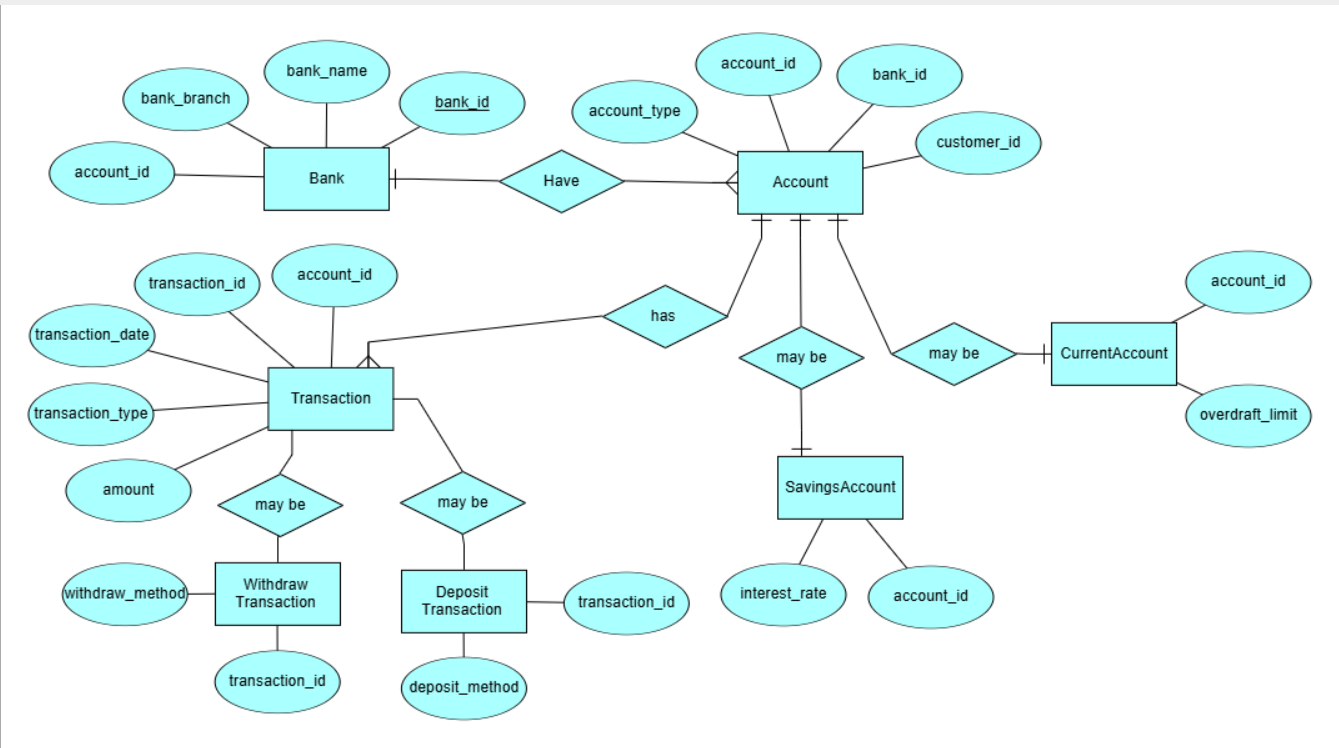
transaction\_id INT (FOREIGN KEY)

withdraw\_method VARCHAR(50)

**SCHEMA DIAGRAM:**



**ER DIAGRAM:**

****