

Đại học Quốc gia Hà Nội
Trường Đại học Công nghệ



BÁO CÁO BÀI TẬP LỚN

Bài tập lớn Cơ sở dữ liệu

Hà Nội, 2024

Mục lục

I	Đề tài	2
1.	Lý do chọn đề tài	2
2.	Phát biểu bài toán	2
3.	Mô tả nghiệp vụ hệ thống	3
II	Mô hình của hệ thống	4
1.	Mô hình ER	4
2.	Mô hình quan hệ	4
III	Đặc tả dữ liệu	5
1.	Dữ liệu tại mỗi bảng	5
2.	Mối quan hệ giữa các bảng	5
IV	Cài đặt	6
1.	Tạo cơ sở dữ liệu và chèn dữ liệu	6
2.	Truy vấn dữ liệu	8
3.	Ràng buộc dữ liệu bằng trigger	8
4.	Giải quyết nghiệp vụ bằng procedures	8

I

Đề tài

1. Lý do chọn đề tài

Trong thời đại công nghệ số hiện nay, các ngân hàng đang ngày càng phụ thuộc vào việc ứng dụng công nghệ thông tin để nâng cao hiệu quả hoạt động và chất lượng dịch vụ. Một trong những yếu tố cốt lõi để hỗ trợ các hoạt động này chính là hệ thống cơ sở dữ liệu. Tuy nhiên, sự phức tạp của các nghiệp vụ ngân hàng như quản lý thông tin khách hàng hay xử lý giao dịch tài chính đòi hỏi các giải pháp cơ sở dữ liệu không chỉ đảm bảo hiệu suất mà còn phải đáp ứng các yêu cầu nghiêm ngặt về tính chính xác và an toàn.

Vì vậy, nhóm đã lựa chọn đề tài **Xây dựng cơ sở dữ liệu cho Hệ thống quản lý ngân hàng**. Việc chọn đề tài này xuất phát từ nhu cầu thực tiễn trong ngành, khi mà các ngân hàng cần một nền tảng dữ liệu mạnh mẽ để quản lý khối lượng thông tin ngày càng lớn, hỗ trợ ra quyết định kịp thời và tối ưu hóa trải nghiệm người dùng.



2. Phát biểu bài toán

Cơ sở dữ liệu dành cho hệ thống quản lý ngân hàng được xây dựng nên để thực hiện các nhiệm vụ như lưu trữ thông tin khách hàng, quản lý tài khoản, xử lý các giao

dịch, theo dõi các khoản vay và rủi ro tài chính, quản lý hiệu suất làm việc của nhân viên và mỗi chi nhánh trong hệ thống ngân hàng.

3. Mô tả nghiệp vụ hệ thống

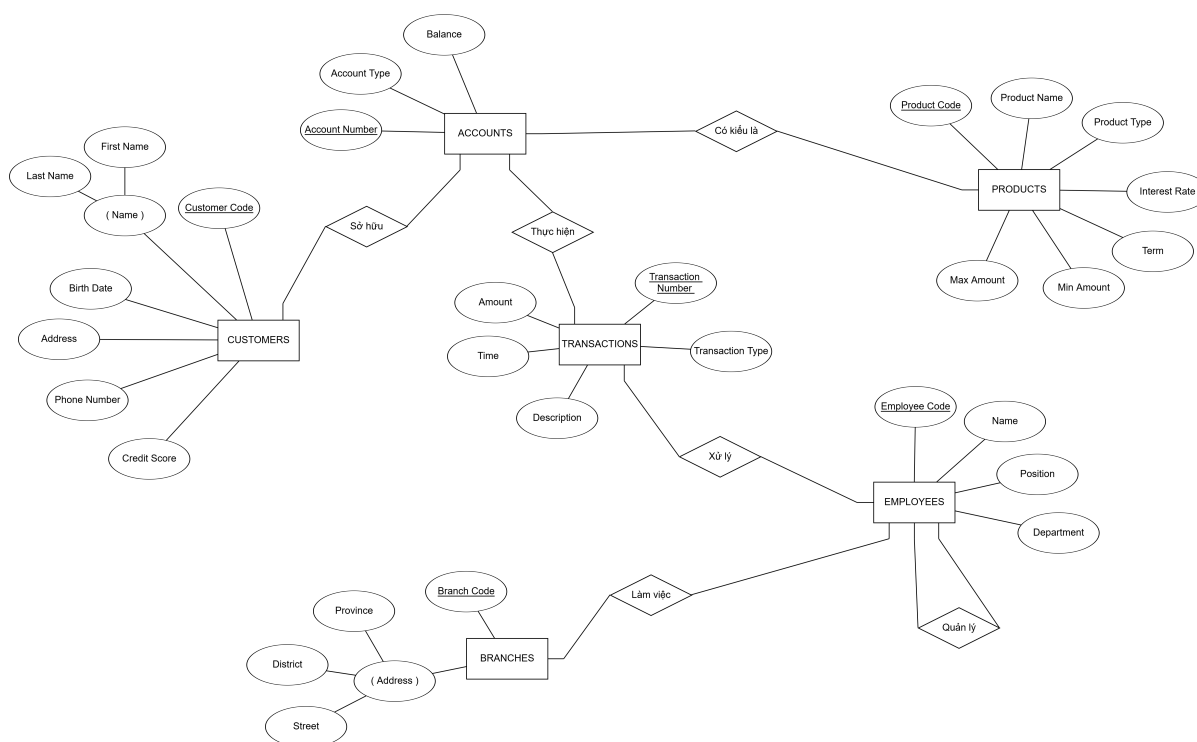
Nhìn chung, một hệ thống quản lý ngân hàng sẽ bao gồm các nghiệp vụ sau đây:

1. **Quản lý khách hàng:** Lưu trữ thông tin cá nhân của mỗi khách hàng tại ngân hàng. Mỗi khách hàng là một cá nhân duy nhất và họ có thể mở nhiều tài khoản khác nhau. Mỗi người họ cũng có thể có những đặc điểm tài chính khác nhau và những thông tin này cần được lưu trữ để quản trị rủi ro tài chính cho ngân hàng.
2. **Quản lý tài khoản:** Lưu trữ thông tin các tài khoản tại ngân hàng. Mỗi tài khoản là một tài sản của khách hàng nào đó gửi tại ngân hàng, chỉ được gắn với một khách hàng duy nhất. Tài khoản ngân hàng cũng chính là phương tiện để một khách thanh toán, nên đi kèm với tài khoản ngân hàng sẽ có những thông tin như số dư, loại hình và lịch sử giao dịch.
3. **Quản lý giao dịch:** Giao dịch là nhật ký ghi lại những biến động về số dư và hướng đi của nguồn tiền trong ngân hàng. Mọi hoạt động tài chính của mỗi khách hàng cần phải được ghi nhận, xử lý, lưu trữ một cách chính xác và minh bạch. Hệ thống phải có khả năng ghi nhận các loại giao dịch khác nhau như chuyển khoản, nộp tiền, rút tiền, v.v. Mỗi giao dịch gắn với các thông tin như mã giao dịch, thời gian thực hiện, số tài khoản nguồn và đích, số tiền giao dịch và trạng thái giao dịch.
4. **Quản lý sản phẩm tài chính:** Khi cung cấp đa dạng các dịch vụ và tiếp cận đến nhiều đối tượng khách hàng thì đòi hỏi phải có sự lưu trữ thông tin các sản phẩm này. Khi khách hàng mở tài khoản, thì tài khoản đó sẽ ứng với một sản phẩm tài chính nhất định. Ngân hàng có thể cung cấp nhiều loại sản phẩm tài chính khác nhau như tài khoản tiết kiệm, gói vay, thẻ tín dụng, v.v. Mỗi sản phẩm sẽ có những đặc điểm khác nhau về lãi suất, kỳ hạn và điều kiện tài chính.
5. **Quản lý nhân viên và chi nhánh:** Ngân hàng được vận hành và quản lý bởi các nhân viên làm việc tại ngân hàng. Mỗi nhân viên phụ trách những công việc khác nhau, chủ yếu là thực hiện xử lý các giao dịch. Ngoài ra, ngân hàng cũng có các chi nhánh khác nhau, là nơi để các nhân viên làm việc và để khách hàng mở tài khoản. Việc quản lý nhân viên và chi nhánh bao gồm theo dõi công việc, đánh giá hiệu suất của mỗi nhân viên và chi nhánh.

II

Mô hình của hệ thống

1. Mô hình ER



Hình II.1: Sơ đồ mô hình thực thể - quan hệ

2. Mô hình quan hệ

III

Đặc tả dữ liệu

1. Dữ liệu tại mỗi bảng

Dữ liệu khách hàng được lưu trữ tại bảng **customers** với các trường dữ liệu sau:

Dữ liệu	Ý nghĩa	Kiểu	Minh họa
customerID	Mã định danh cho mỗi khách hàng tại ngân hàng, gồm 8 kí tự.	varchar	23020346
lastName	Họ của khách hàng	varchar	Trần
firstName	Tên của khách hàng	varchar	Minh Huy
birthDate

2. Mối quan hệ giữa các bảng

Cơ sở dữ liệu có các mối quan hệ sau:

- Giữa bảng **customers** và **accounts** có mối quan hệ được liên kết bởi trường

IV

Cài đặt

1. Tạo cơ sở dữ liệu và chèn dữ liệu

a. Tạo cơ sở dữ liệu:

```
CREATE DATABASE IF NOT EXISTS bank_db;
```

- Bảng branches:

```
CREATE TABLE IF NOT EXISTS bank_db.branches (  
    branchCode VARCHAR(50) NOT NULL  
        PRIMARY KEY,  
    branchName VARCHAR(256) NOT NULL,  
    province VARCHAR(50) NULL,  
    district VARCHAR(50) NULL,  
    street VARCHAR(256) NULL  
);
```

- Bảng customers:

```
CREATE TABLE IF NOT EXISTS bank_db.customers (  
    customerID VARCHAR(50) NOT NULL  
        PRIMARY KEY,  
    lastName VARCHAR(50) NOT NULL,  
    firstName VARCHAR(50) NOT NULL,  
    birthDate DATE NOT NULL,  
    phoneNumber VARCHAR(15) NULL,  
    address VARCHAR(256) NULL,  
    creditScore INT NULL  
);
```

- Bảng employees:

```
CREATE TABLE IF NOT EXISTS bank_db.employees (
  employeeCode VARCHAR(50) NOT NULL
    PRIMARY KEY,
  lastName     VARCHAR(50) NOT NULL,
  firstName    VARCHAR(50) NOT NULL,
  position     VARCHAR(256) NULL,
  department   VARCHAR(256) NULL,
  managerCode  VARCHAR(50) NULL,
  branchCode   VARCHAR(50) NULL
);
```

- Bảng products:

```
CREATE TABLE IF NOT EXISTS bank_db.products (
  productCode VARCHAR(50) NOT NULL
    PRIMARY KEY,
  productType  VARCHAR(256) NULL,
  interestRate FLOAT       NULL,
  term         INT         NULL,
  minAmount    BIGINT      NULL,
  maxAmount    BIGINT      NULL
);
```

- Bảng accounts:

```
CREATE TABLE IF NOT EXISTS bank_db.accounts (
  accountNumber VARCHAR(256) NOT NULL
    PRIMARY KEY,
  accountName    VARCHAR(256) NOT NULL,
  balance        BIGINT       NOT NULL,
  customerID     VARCHAR(50)  NULL,
  productCode    VARCHAR(256) NULL,
  openingBranch  VARCHAR(50)  NULL,
  openingDate    DATETIME DEFAULT (NOW()) NULL
);
```

- Bảng transactions:

```
CREATE TABLE IF NOT EXISTS bank_db.transactions (
  transactionNumber INT AUTO_INCREMENT
    PRIMARY KEY,
  sourceAccount     VARCHAR(50) NULL,
  targetAccount     VARCHAR(50) NULL,
  transactionType   VARCHAR(50) NULL,
  amount            BIGINT      NOT NULL,
  time              DATETIME DEFAULT
    CURRENT_TIMESTAMP NOT NULL,
  description       VARCHAR(256) NULL,
  employeeCode      VARCHAR(50) NULL
);
```


- Tạo các ràng buộc dữ liệu:

```
ALTER TABLE bank_db.employees
ADD CONSTRAINT branch___fk
FOREIGN KEY (branchCode)
REFERENCES bank_db.branches (branchCode),
ADD CONSTRAINT employees_managers___fk
FOREIGN KEY (managerCode)
REFERENCES bank_db.employees (employeeCode);

ALTER TABLE bank_db.accounts
ADD CONSTRAINT customerID___fk
FOREIGN KEY (customerID)
REFERENCES bank_db.customers (customerID),
ADD CONSTRAINT openingBranch___fk
FOREIGN KEY (openingBranch)
REFERENCES bank_db.branches (branchCode),
ADD CONSTRAINT productCode___fk
FOREIGN KEY (productCode)
REFERENCES bank_db.products (productCode);

ALTER TABLE bank_db.transactions
ADD CONSTRAINT sourceAccount___fk
FOREIGN KEY (sourceAccount)
REFERENCES bank_db.accounts (accountNumber),
ADD CONSTRAINT targetAccount___fk
FOREIGN KEY (targetAccount)
REFERENCES bank_db.accounts (accountNumber),
ADD CONSTRAINT transactions___fk
FOREIGN KEY (employeeCode)
REFERENCES bank_db.employees (employeeCode);
```

- b. Chèn dữ liệu

2. Truy vấn dữ liệu

3. Ràng buộc dữ liệu bằng trigger

4. Giải quyết nghiệp vụ bằng procedures