İDRİS DAĞ – 2014100006 – 02.01.2019

BANK SYSTEM

My project will be on a bank's working system. In short, there will be a system that will keep the database and general transactions of a bank. It will keep the data of not only the bank, but also other important factors such as customers according to the transactions they will make. There will be classes in this system according to the transactions carried out by the bank. These classes will be subdivided into each other and listed according to their features. For example; a bank provides online transactions and customers can log in online at any time.

My project consists of 15 different classes. All classes have different relationship with each other and they have different numbers of tables. I have changed my tables from the time I had got feedbacks from my teacher. I have added 3 new tables to make the details transactions clear. I have been learning what a bank does and what are the trasactions of a bank thanks to this project.

**BUSINESS RULES**

An employee has at least zero or one Bank

A bank has at last one or many employees.

A bank has at least 1 or many branchs.

A branch has at least 1 or one bank.

A bank has zero or many accounts.

An account has at least one or many transactions.

A transaction has one or one account.

A transaction has zero or many ATM transactions.

A transaction has zero or many online transactions.

Online transaction has one or many transaction.

ATM transaction has one or many transaction.

A customer can login zero or many times.

A customer has zero or many accounts.

An account can be owned by one or only one customer.

A customer has zero or many saving account.

A customer has zero or many current account.

ATM transaction have zero or many withdrawal transactions and transfer transactions.

Online transactions can be login by custmers zero or many times.

A customer can change his/her PIN zero or many times.

A customer has zero or many loan account.

SQL

Bank(Name, BankID, AccountNo, BranchCode, Adress, AddAccount, RemoveAccount, AddCustomer, RemoveCustomer, Credit, Debit)

Branch(BranchCode, BankCode, Name, Address, City)

Bank Employees(TCNo, BankID, Name, Surname, Phone, Email, Salary, PhoneNumber)

Accounts(Account No, Type, BranchCode, BankID, OpenedDate, ClosedDate)

Current Account(Account No, CustomerTC, CustomerName, CustomerSurname, BranchCode, Balance)

Saving Account(SavingAccountNo, CustomerTC, CustomerName, CustomerSurname, AccountNo, BranchCode, Balance, InterestRate, InteretPeriod)

Loan Account(LoanAccountNo, CustomerTC, CustomerName, CustomerSurname, LoanLimit, Balance, BranchCode, AccountNo)

Transactions(Transaction ID, Account No, TransactionDate, Transaction Type)

Online Transactions(Card no, Password, CVV No, TransactionType, Date, TransactionID, Amount, PostBalance)

ATM Transactions(ATMBranchCode, TransactionID, TransactionDate, TransactionType, Amount, PostAmount)

Customers( CustomerTC, AccountNo, Name, Surname, DateOfBirth, Adress, PhoneNumber, Email)

Login(Username, Password, SecurityQuestion, SecurityAnswer, LoginTime)

Withdrawal transactions(PIN, ATMBranchCode, Amount, PostAmount, TransactionDate)

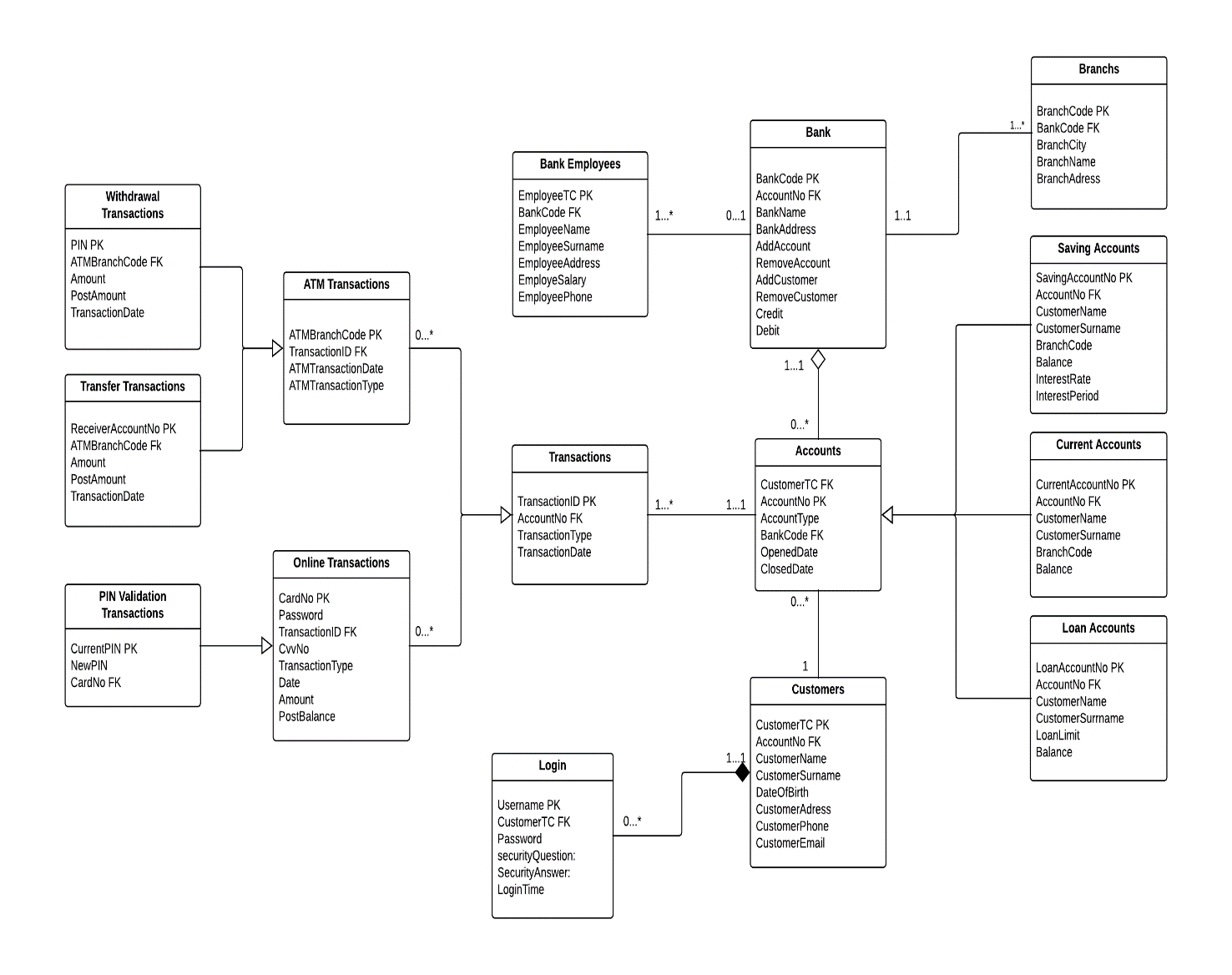
Transfer Transactions(ReceiverAccountNo, ATMBranchcode, Amount, PostAmount, TransactionDate)

PIN Validation Transactions(CurrentPIN, NewPIN, CardNo)

Physical Diagram



UML Diagram



5 SQL CÜMLESİ

insert into Customers(CustomerTC, AccountNo, CustomerName, CustomerSurname)

values ('555', '1', 'erhan', 'kadir');

insert into [Bank Employees](EmployeeTC, EmployeeName, EmployeeSurname, EmployeeAddress, EmployeeSalary, BankCode)

values ('15', 'turgut', 'yildirim', 'maltepe', '5000','1');

select \* from [Loan Accounts]

select \* from dbo.Transactions

update [Login]

set LoginTime='1111'

where SecurityQuestion='aa';

select [Bank Employees].EmployeeName, [Bank Employees].EmployeeTC, Customers.CustomerTC, Customers.CustomerName

from [Bank Employees], Customers where [Bank Employees].BankCode = Customers.AccountNo4

PROCEDURE- 1

create procedure ChangePIN @NewPIN nvarchar(50), @adi nvarchar(50)

as

begin

if @adi= ( select CardNo from [PIN Validation Transactions] where [PIN Validation Transactions].CardNo = @adi)

begin

update [PIN Validation Transactions] set NewPIN = @NewPIN where [PIN Validation Transactions].CardNo = @adi

print ' yeni PIN '+ @NewPIN

end

else

begin

print 'Yeni PIN Girilmedi'

end

end

PROCEDURE – 2

Create Procedure ekle @CustomerTC nvarchar(50), @CustomerName nvarchar(50), @CustomerSurname nvarchar(50)

as

begin

insert into Customers(CustomerTC, CustomerName, CustomerSurname)

values ('123456', 'burak', 'taydas')

exec ekle '11823020298', 'idris', 'dag'

end

TRIGGER -1

create trigger toplammusteri

on Customers

after insert

as

select COUNT(Customers.CustomerName) as TotalProductsNumber from Customers

TRIGGER – 2

create trigger LoanAccountHolder

on [Loan Accounts]

after update

as

declare @Loan nvarchar(50);

set @Loan = (select LoanAccountNo from inserted)

print 'Yeni Loan ' + @Loan;

VIEW – 1

create view allcustomers as

select Customers.CustomerName, CustomerSurname, CustomerTC, [Bank Employees].EmployeeName, EmployeeSurname, EmployeeTC

from Customers, [Bank Employees]

where CustomerTC = EmployeeTC

select \* from allcustomers

VIEW – 2

create view OldAndNewPINs as

select [PIN Validation Transactions].CurrentPIN, NewPIN, [Withdrawal Transactions].PIN

from [PIN Validation Transactions], [Withdrawal Transactions]

where CardNo = '123456' and CardNo ='654321' and ATMBranchCode = '1' and ATMBranchCode = '2'

select \* from OldAndNewPINs