PROBLEM DESCRIPTION OF BANK MANAGEMENT SYSTEM

INTRODUCTION

The Bank Account Management System enables users keep track of customers' banking activities such as their accounts, loans and money transfers. In this project, we aim to provide an ability to display all the accounts and to intervene in all banking activities of the customers in Turkey . Users are able to trace all the monetary transactions of a citizen or a corporation in any banking environment and query suspicious transfers and loan activities. Furthermore, users are also able to suspend, resume or block customers' accounts and to intervene in transaction and loan applications if need be.

Basically, the main goal of this project is to implement a cost-effective and efficient way of observing financial relations between both citizens & corporations. Users can also filter large amount of money transfers of some certain customers (such as criminals) and query ATM locations which those customers withdraw some money. They can employ these sort of queries to discover either where the person is or which people have financial relations with him/her.

We build this project using SQL. To explain briefly, Structured Query Language (SQL) is the standard and most broadly utilized programming language for relational databases. It is mainly used to manage, control and organize information in all sorts of frameworks in which different information relationships exist.

The Bank Management System is to be built upon the request of both Ministry of Treasury and Finance of Turkey and Ministry of Justice of Turkey. Only senior officers in related departments will be allowed to have an access right to the database.

ENTITY SETS AND THEIR ATTRIBUTES

- **A bank** is identified by their **trade registry number**, **name**, **address** (where an address is identified by street (composition of *street number and street name*), city, state and the zip), **e-mail addresses** and **phone numbers**.
- **Customer**s takes these loans and customers identified by their **ID**, **name**, **job**, **date of birth**, **age**, **SSN**, **address**, **phone** and **e-mail** where a name is consisting of first name, mid name and surname, and an address is consisting of street (*street number and street name*), city, state and the zip.
- Customer can take out **loan**s and loans classified by their **ID**, **date**, **amount** of money, **due date**, **interest**, **installment and installment amount** of money (monthly payment).
- These customers have **account**s and accounts are specified by their **account number**, **type**, **currency**, **balance** and **status**.
- A **manager** is responsible for intervening in accounts. It is defined by **a unique name**, **a password** and **a ministry** for which he/she works for.
- Customers can open a bank account by visiting a bank's **branch.** A branch is defined by an **ID**, a **name**, an **address**, **phone numbers** and **emails addresses** (where an address is a composition of street (*street number and street name*), city, state and the zip) and a **salary**.
- Deposits and withdrawals can be made by accounts through **ATM**s and ATMs are identified by their **ID**s, **cash** money where a cash is defined by type and balance and **address**es where an address is a composition of street (*street number and street name*), city, state and the zip.

- Accounts have **card**s and these cards determined by their **card number**, **type**, **status** and **expiration date**. There are two types of cards: **credit** which has a limit and **debit** which has a balance.

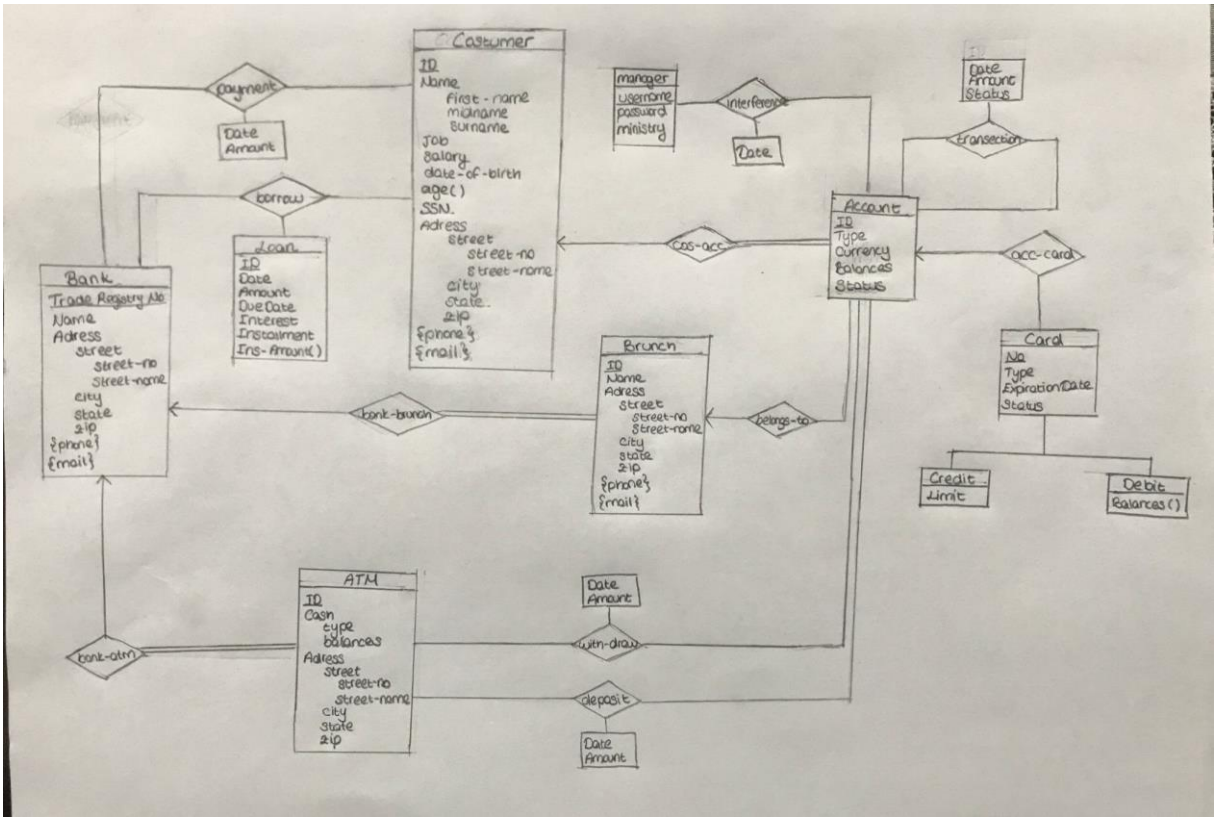
RELATIONSHIP SETS AND THEIR ATTRIBUTES

- Money **transaction**s can be made between these accounts, transactions are defined by their **date**, **amount** of money and the **status**.
- **Deposit**s can be made through ATMs,then money comes into customer's bank account. A deposit is identified by **date** and **amount** of cash money.
- **Withdrawal**s can also be made through ATMs, money goes out of customer's bank account. A withdrawal is identified by **date** and **amount** of cash money.
- A customer can **borrow** some money (namely loan) from any bank. Borrow is identified by a **trade registry no** which belongs to the associated bank, **loan id** which belongs to the associated loan and **customer id** which belongs to the customer borrowed that loan.
- Customer can make a **payment** to any bank. A payment is identified by **customer id** which belongs to the associated customer, **trade registry no** which belongs to the associated bank, **amount**, **date** and **status**.
- **Interference** in bank accounts can be made by a manager who works for either Ministry of Treasury or Ministry of Justice. It is identified by a **username of the manager** and an **id of the account** in which he or she intervenes.

USERS OF THE SYSTEM

- This system will only be used by managers in related departments in both Ministry of Treasury and Finance and Ministry of Justice of Turkey.

ER DIAGRAM



- A bank branch has to be owned by a bank and a bank may have many branches in different locations.
- Each account in the system has to get involved in a relation with at least one customer and each account has to be opened in a bank branch. Again a customer is allowed to have many accounts.
- Each ATM has to be related to a certain bank and any bank is allowed to have multiple ATMs.
- There might be many different transactions between accounts.
- Accounts are allowed to have many different cards, either debit type or either credit type.
- Loan can be given to many different customers by any bank and a customer can get a loan from many different banks.
- Each customer in the database has to get involved in a relation to a bank and a bank may have many customers.
- An account may put or withdraw some money through many ATMs and an ATM may serve to many accounts.
- A manager is allowed to intervene in any bank account and any bank account can be accessed by any manager.

ADDITIONAL BUSINESS RULES & CONSTRAINTS

Since a successful deployment requires to establish business requirements within business constraints, we should also define some additional rules. These constraints might be of salary, date, account of monetary transactions in a limited time or other type of rules.

For instance, many of the bank management systems should adhere to transaction constraints such as a limitation on the amount of money.

As we have limited resources and may need to reuse the same resources in different phases of the project, it is essential to define these integrity constraints in order to enforce the organization rules and prevent bad data entry.

Here are the integrity constraints of our system:

1. Transaction Limitations :
   - Amount of transition can not be more than 10K liras at once.
2. Withdrawal & Deposit Limitations :
   - Amount of withdrawal & deposit can not be more than 5K liras.
   - An account can not make deposit or withdraw money more than three times in a day.
3. Birth date of a customer must be at least 2003.
4. Card expiration date must be at least 2024.
5. Credit card limit can not be less than 5K and more than 50K.
6. Any customer can not take out a loan whose amount is more than 50*salary.
   - Due date of loan can not be after 2040.
7. Amount of any transaction can not be more than the balance of the sender account.

ER DIAGRAM TO RELATIONAL SCHEMA

Bank (TradeRegisteryNo, Name, street_no, street_name, city, state, zip)

Bank_phone(TradeRegisteryNo, phone)

Bank_mail(TradeRegisteryNo, mail)

Branch(ID, Name, street_no, street_name, city, state, zip, TradeRegisteryNo)

Branch_phone(ID, phone)

Branch_mail(ID, mail)

Customer(ID, first_name, midname, surname, job ,date_of_birth, age, SSN, street_no, street_name, city, state, zip)

Customer_phone(ID, phone)

Customer_mail(ID, mail)

Loan(ID, date, amount, Due_date, interest, installment, install_amount)

Payment(Customer_ID, Bank_ID, amount, date, status)

Borrow (Loan_ID, Customer_ID, Bank_ID)

Manager(username, password, ministry)

Account(ID, type, currency, balances, status, Customer_ID, Branch_ID)

Transaction(Sender_Acc_ID, Receiver_Acc_ID, amount, date, status)

Interference(username, account_id, idate)

Atm(ID, type, balances, street_no, street_name, city, state, zip, TradeRegisteryNo)

Deposit(Atm_ID, Account_ID, date, amount)

Withdrawal(Atm_ID, Account_ID, date, amount)

Card(No, Type, expiration_date, status, Account_ID)

Credit(No, limit)

Debit(No, balances)