Banking System

Arrenn Baral, Prakriti Basyal

Team 10

CS 157A Summer 2020

Instructor Mike Wu

**Project Description**

This​ ​project​ ​will​ ​be​ ​a​ ​database​ ​application​ ​that​ ​is​ ​based​ ​in​ ​the​ ​general banking system. ​ ​The​ ​goal​ ​of this​ ​application​ ​is​ ​to​ ​provide​ ​a​ ​convenient​ ​way​ ​to​ ​access user’s banking information. Users​ ​will​ ​be​ ​able​ ​to​ ​login with designated username and password, access bank account, deposit, withdraw, get a report of transaction if a user wishes to. ​This​ ​application​ ​will​ ​also​ ​provide​ ​a​ ​user-friendly​ ​interface​ ​that​ ​is​ ​clear​ ​and intuitive. We use bank system in our daily lives. ​We login into bank systems like Chase, Wells Fargo, Bank of America, and list goes on. Deposits, withdraws, transfers, statements are some of the example of our daily used operations on the bank systems. But we never thought what actually runs behind the screen. Since we are taking database class, we are getting great opportunity to explore system behind the screen. We got the motivation from our daily used operations in the bank system. Let’s start with simple structure our structure of the banking system

**Word on structure**

We will be Apache Tomcat webserver to run our banking system.

We will be implementing our code through Java IDE.

For database system, we are using MySQL but data entries.

We will be using DMLs, and DDLs for operations like deposits, withdrawals, and get statement.

We will be using HTML and CSS for beautify our banking system. It is important to give user-friendly look to users.

**System Environment**

***Structure of the System***



Apache Tomcat Request JSP Files

HTML, CSS

Java Response







SQL Queries Response



Database

MySQL

***Hardware and Software***

Apache Tomcat

MacOS

***RDMS***

MySQL Community Server 8.0.20

Application Languages

Java, HTML, CSS, PHP (maybe), SQL

**Functional Requirements**

***How user will access the system***

The system provides functionality to create an account. User will be asked to set up a new account if he/she is new user else simply log in using their existing user credential in the database. If user wishes to setup a new account, the system opens new page for the user where the system asks several information for setting up the account such as First name, Last name, Date of Birth, Email Address, Gender, username, and password. Once the setup process completed, user will be redirected to banking system where they can select different operations. User cannot login if user’s credential is not saved in the database. User cannot access without setting up the account. If user tries to access with fake username and password, they will be redirected to login page to use valid username or password. However, once user logs in, system will not ask to sign in again. System will assume user logged in through trusted device. Once logged in, users will be able to use several functions.

**Functions**

*Log in*

User shall set up an account to get an access to the banking system.

User’s credentials must be on saved in the database.

*Withdraw*

Once logged in, user shall select the withdraw option to get their money.

Once user chooses this function, the withdrawal sum will be deducted from the total amount.

*Deposit*

Once logged in, user shall use this function to deposit the money.

Once deposited, the deposited sum will be added to the total amount.

*Statement*

User shall choose this function to get a statement of the transaction after successful withdrawal and deposits.

*Apply for Loan*

User shall use this function to apply for a loan.

User gets approved only if thousand dollars is available in his bank account.

*Order a cheque book*

User can use this function to order a cheque book.

There will be a charge of $1.00 for ordering a check.

*Manage/update account information*

User shall use this function to edit account information such as address,

and phone number.

*Transfer fund*

User shall use this function to transfer fund to other.

Receiving person much be setup before transferring.

*Search statement by date*

If user wants to search statement from specific date, they can choose this function to select the date and search.

*Sort the transactions*

If user wishes to sort the transactions from higher to lower or lower higher, they can use this function to sort as wished.

*Delete account.*

If user wishes to terminate the account, they can simply use this function to delete the account. Once deleted, account will not be recovered. It will

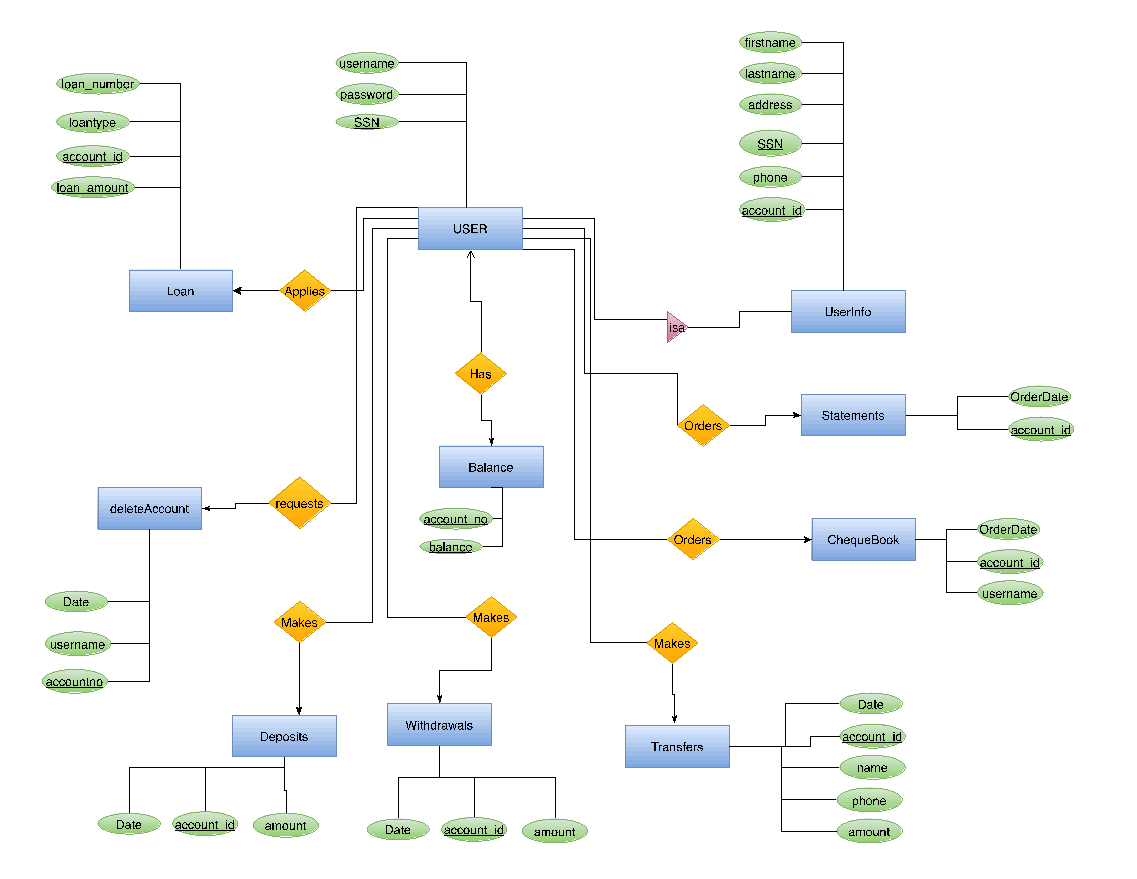
be permanent.

*Display Account Information*

The user will be able to view the account information through this function.

**Graphics User Interface (GUI)**

For the GUI, we will be using HTML to provide user-friendly environment. If possible, we will allow user to upload a picture of them for the account.

The​ ​security​​ ​of​ ​each​ ​user​ ​account​ ​will​ ​be​ ​protected​ ​by​ ​a​ ​username​ ​and​ ​password created​ ​by​ ​the​ ​user. ​ ​The​ ​username​ ​for​ ​each​ ​account​ ​will​ ​not be their email. User name will be the one which they choose while setting up the account.​ ​The user​ ​account​ ​information​ ​will​ ​be​ ​securely​ ​stored​ ​in​ ​the​ ​server.​ ​​ ​In​ ​order​ ​to​ ​login​ ​to the​ ​system​ ​the​ ​user​ ​must​ ​provide​ ​an​ ​existing​ ​username​ ​account​ ​along​ ​with​ ​the associated​ ​password.​ ​The​ ​website​ ​will​ ​be​ ​implemented​ ​using​ ​https​ ​providing​ ​an encrypted​ ​connection​ ​between​ ​user​ ​and​ ​server.

User:

User entity is the service offered by the bank where one may deposit, withdraw or transfer money,check balance,apply loan,check statement in their account information. Every customer has a unique account. The entity set contains the customers username and password to login and the ssn. SSN is the primary key.

USERINFO:

The UserInfo entity is the entity that holds the informations of the user.This entity represents the unique customer archetype information which is the customer’s firstname,lastname,address,phone,SSN,account\_id.The SSN and the account\_id are the primary key.It is Is-a relationship so it inherits the attributes from the User.

BALANCE:

The Balance entity is the entity that represents the total amount of the fund in the user's account.Every user has a balance.It has two attributes the account\_no and the balance which gives them the information. And both of them are the primary key.

*Loan:*

Loan entity represents lending money to one or more individuals with the required interest rate from the bank.The user can apply for a loan.So the loan entity has the attributes loan number, Loan\_type,account\_id,loan amount where account\_id and the loan amount is the primary key.

DELETED ACCOUNT:*This is an entity provided to the user as a service to delete the account.A user can request to delete the account if they no longer are interested in the service.Its attributes are Date,username and account no where account number is the primary key.*

Statements:

Statement entity represents a record of the financial activities done by the customer which are also given as the written reports. It basically keeps track of the record list of transactions done by the customer over a certain period of time.A user can order the statement any time. So, orderdate and the account\_id are the attributes where account\_id is the primary key..

*Transfer:*

This entity set represents the electronic fund transfer from one person to another to the bank accounts. It is one of the services provided by the bank so that people can transfer, making people's lives easier. This entity set has the amount, date,name,phone number and the account\_id so they can transfer directly to their account.

*Deposit:*

Deposit is an entity set which is available to customers to deposit money into their bank account. These deposits can be made through cash, check by the customer.Deposit entity set has the date,account\_id and the amount as an attribute.Account\_id is the primary key.

*Withdrawal:*

Withdrawal entity set represents giving the customer an option to withdraw money whenever they want. The bank grants access to the customer for the withdrawals except for the unusual activities. This entity set contains the date,account\_id and the amount as an attribute.Account\_id is the primary key here.

CHEQUEBOOK:

This entity is another service provided to the customer where the User can order the Chequebook through the orderdate,account\_id and the username attributes.

*UserAppliesLoan*

UserAppliesLoan is a relationship that connects Users and Loan entity sets. This is a many to many relationship because there can be many users and apply for different loan amounts.

*UserGeneratesStatement:*

UserGeneratesStatement is a relationship that connects the user and statement table. Here, the various users look for various transactions according to the date and this is also many to many relationships.

*UserMakesDeposit*

UserMakesDeposit is a relationship that connects the User and the Deposit entity sets. This is a many to many relationships since there can be several users making several deposits.

*UserMakesTransfer*

UserMakesTransfer is a relationship that connects the User and the Transfer entity sets. This is also a many to many relationship. Here the user transfers the amount to the receiver's name.

*UserMakesWithdrawal*

UserMakesTransfer is a relationship that connects the User and the Withdrawal entity sets. This is a many to many relationship because many users can withdraw certain amount of money at any time.

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer screen

Description automatically generatedA screenshot of a computer screen

Description automatically generated

A screenshot of a computer screen

Description automatically generatedA screenshot of a computer screen

Description automatically generated

A screenshot of a computer screen

Description automatically generatedA screenshot of a computer screen

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer screen

Description automatically generated

A screenshot of a computer screen

Description automatically generatedA screenshot of a computer screen

Description automatically generated