Project Description

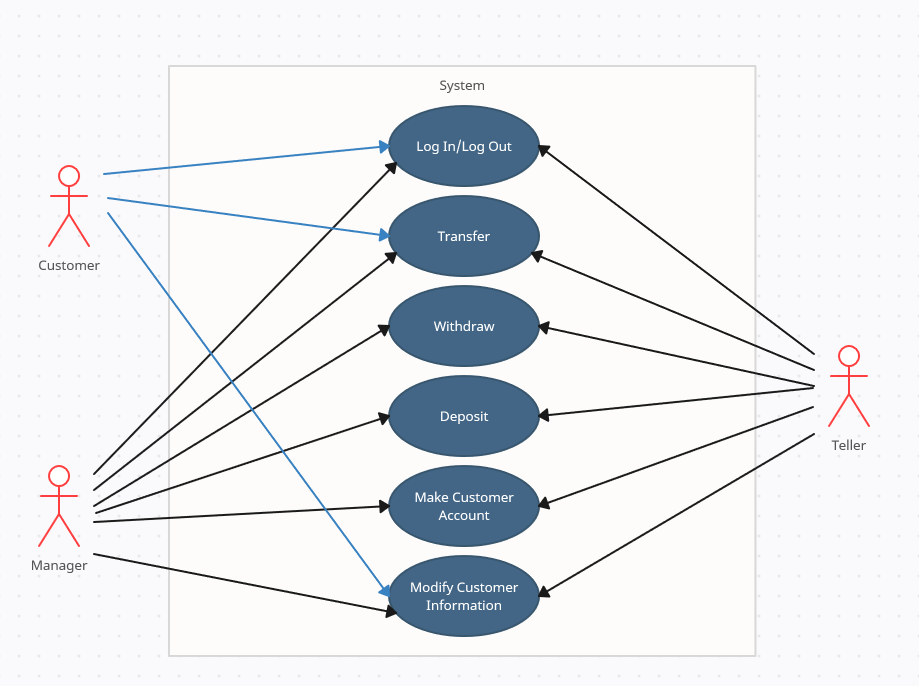
The purpose of building this application was to form an understanding with how we as a group could be tasked with building a mock program that performs a specific task. This application was a Credit Union application where a user could log in and access their funds and even get a Teller to withdraw or deposit cash for them as needed. The over-all goal was to build a high-fidelity menu for the user, while the employee views were left clunky yet easy to use. We will get database training as well as object oriented programming training as well. The over all task is to work as a group to be able to split workloads and build a functional application that will be easy enough to use that a brief explanation is all that is required of the developer.

User Guide

To get started a user will need to talk to a teller in order to create an account. Upon opening the application the user is prompted to log in using their account credentials. If they do not have an account then they must have a Teller create one for them. Once logged in the user must select that they are a customer and not an employee as employees could be a customer as well which requires being able to select what view they wanted. Once the user has logged in they have a menu to choose options from. They can simply view their accounts and how much money is in them, or they can make changes to their account. These changes include a button to allow the name change of the bank accounts. Another button is to change the address of the user, and there is a button to change the name of the user as well. There is also a menu to transfer money between accounts as well as create of remove accounts. When removing an account, the user will be prompted asking where they want the money from the account they are closing to go. For an employee, they must know or have a manager make sure that the database is correctly set up as the password for the database is located within the script and is not a menu to enter in the database credentials. For an employee, after logging in they can select Teller View which will allow them to create new customer accounts, which requires that the SSN field be unique, otherwise they will be updating an existing account. Tellers also have the ability to withdraw and deposit money into a customers account. The customer must be able to tell the Teller their SSN as this feature only works if the SSN has been entered into the provided field, otherwise the buttons don’t do anything and there is no error message at this time. A Managers only ability is to log into the Manager View and see statistics of the Credit Union database.

UML Diagrams

Use Case Diagrams



Testing Overview

Known Bugs

There are a few bugs like being able to manipulate the database from within any text entry field that would be stored into the database after execution. This lack of sanitizing the inputs will create a security risk as anyone could flat out delete the entire database if they knew how to. There is also the issue of closing a window on any of the extended views like creating a customer account will result in the entire program closing and not just that window, so for now some windows have a button that is to close just that window rather than the whole program. There are many public methods that could pose as a security risk and should be accessed immediately to ensure that no Teller can access information of a customer without needing to edit or help a customer, and a customer should never be able to see what the Teller is doing, as well as what other customers accounts are like.

Possible Feature Additions

We had already planned on building an ATM application that would be for customers to enter in their credit or debit card and be able to access their funds from a remote location. The only thing we would need to do is implement a way for Tellers to issue bank cards for the customers bank accounts. Each card would connect to only one bank account so any customer could have one or multiple cards. Then when a customer entered in the card information into the ATM they could then use this application like a normal ATM, no reason to reinvent an ATM.