Mohamed Abouzaid Id: 501218839 Section:02

Use-case overview:

The bank system consists of two actors. The first actor is the manager. The manager can login into the system through the login page. Then the manager could create a new customer account. The manager first enters the username, password and account number. The system checks if the username or the account number are duplicated. Then if there is no duplication, the system creates a new account and creates a new file for him in the database. The manager also can display all the users in the database and see their usernames and account numbers only. The manager can delete a customer account. The manager selects the account and clicks on delete to delete the account. When the account is deleted, its file from the database is deleted also. The other actor is the customer. The customer enters the customer page after logging into the system. The customer can deposit an amount of money or withdraw from his account. Also, the customer can online purchase through the customer page. The customer chooses from various options and adds items to the cart then checks out and pays. The payment process requires and extends the check level use-case, because for each account level there is a payment fee. Both actors login to the system and the system authenticates their credentials.

The chosen use-case to be discussed in detail is createCustomer use-case.

Use case Name	createCustomer
Participating actor	Manager uses this use case
Entry condition	The manager logged in manager page
Flow of events	1- The manager enters the username, password and account number for the new customer. 2- The manager clicks on the create customer button. 3- The system checks if the username or the account number is being duplicated or already used. 4- If there is any duplication, the system shows a warning. If not, the customer is added.
Exit condition	The system shows a warning of wrong data entered Or The system successfully creates the customer.
Exceptions	The system displays a warning when duplicates happen.
Special Requirements	At the end of this use case, it includes the createAccountFile for the customer use case. The createAccountFile is invoked by the createCustomer, so that the database updates.

Class diagram overview:

The system first consists of Person class. Person class is an abstract class, which gives an insight about the Customer class and Manager class. Person class contains username which is string and unique for each object, password which is string, and role. The customer class inherits from the Person class and has the role as final static value of Customer. While the username and the password are entered while creating the customer object. The Customer also has a Bank Account and Level. The Manager class is inherited from the Person class and has role, username and password with final values of "Manager", "admin", and "admin" respectively. The Account class is class used to link the customer with his balance in the bank. The account has a unique account number and balance. The Account class has withdraw and deposit methods which acts on the balance. The Level class is an abstract class which implements the online purchasing and change level methods. Silver, Gold, and Platinum classes are inherited from the Level class. Each one withdraw a unique purchase fee. The relation between Customer and Manager with Person is inheritance. Also, the relationship between Level and Silver, Gold, Platinum is inheritance. The relation between Customer and Account, Level is has-a relation which is called aggregation relation.

State Diagram

The state diagram for user authentication starts at the Login Page, where users enter their credentials. Upon submission, the system verifies the credentials and directly transitions to either the ManagerPage or the CustomerPage, depending on the user's role. If authentication fails, the system returns to the Login Page with an error message. When the user logs out, the system moves to the Logout state and then returns to the Login Page for a new session. This diagram captures the key states and transitions from login to accessing the appropriate user page or handling authentication failures.

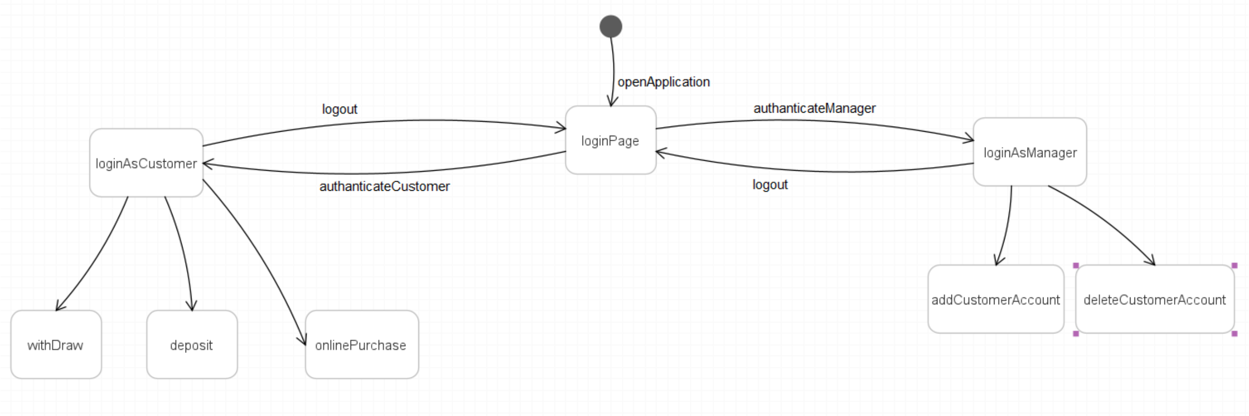

Figure 1 State Diagram for the Bank Project