COE 528 Project Report
Instructor: Boujemaa Guermazi
TA: Jeanne Alcantara

Name: Kashish Patel
Student ID: 501050150

Section: 05

CASE DIAGRAM DESCRIPTION:

User Case Name:

Banking System

Participating Actors:

Customer, Manager, and System

The class diagram for the given code would have five classes: BankAccount, Customer, AccountState, Platinum, and Gold. The class BankAccount has a reference to the AccountState interface, which is implemented by the Platinum and Gold classes. Each BankAccount object can have a state of either Platinum or Gold. The Customer class has a reference to the BankAccount class, indicating that a customer can have a bank account. The BankAccount class has two attributes, balance and accountState, and methods for getting and setting the balance and updating the account state. The Customer class has a method for getting the account and a method for setting the account state. The AccountState interface has methods for performing online purchases, describing the account type, checking the balance, depositing, withdrawing, and updating the balance. The Platinum class and Gold class implement the AccountState interface and provide their own implementation for the methods. The Gold class has a static attribute for the service charge, but the Platinum class does not have any attributes.

The class that I have selected to address point number 2 is the BankAccount class.

The BankAccount class has the following attributes:

- balance: A private instance variable of the double data type, which represents the balance of the bank account.
- accountState: A private instance variable of the AccountState data type, which represents the account state of the bank account.

The BankAccount class has the following methods:

- getBalance(): A public method that returns the balance of the bank account.
- setBalance(int newBalance): A public method that sets the balance of the bank account.
- getAccountState(): A public method that returns the account state of the bank account.
- updateAccountState(AccountState newState): A public method that updates the account state of the bank account.

Flow of Events:

1. Manager Authentication: Upon login, the Manager gains access to functions such as adding or deleting customers, followed by logging out.
2. Customer Authentication: Customers authenticate themselves to access functionalities including making deposits or withdrawals, purchasing items online, and subsequently logging out.
3. Withdrawal Verification: The system verifies withdrawal requests against the available account balance that is if the withdrawal is greater than the balance in the account, it won't work.
4. Online Purchase: Customers can execute online purchases.
5. Online Purchase Fees: Cost of the item is deducted from the account balance. Additionally, an online purchase fee, based on level of customers (GOLD, SILVER, or PLATINUM) is applied.
6. Final Balance: The system gives the final account balance after all transactions are completed.
7. Conclusion: Use case concludes with successful transactions and customer logout/Manager logouts.

Exit Conditions:

All transactions are successful; customer logs out or the manager logouts after adding or deleting customers.

Reference

TMU Project Manual for COE 528 by Faculty of Engineering and Architectural Science Introduction to Java Programming Comprehensive version, 1OTH EDITION by Y. Daniel Liang.

Lecture 6_ Modeling with UML Toronto Metropolitan University,

https://courses.torontomu.ca/d2l/le/content/839005/viewContent/5647654/View