**ATM CASE STUDY**

**SARTHAK GIROTRA**

**CS20B037**

* **DESIGN DECISIONS AND APPLICATION OF OOPS**

Classes:

1. Admin
2. ATM
3. Conversion
4. Transaction
5. User\_Data

Interface:

1. ATM\_Setup.
2. Admin(extends User\_Data and implements ATM\_Setup):
   * Class where admin can make changes to the ATM system
   * Includes the instance variables ID,password,dailyLimit,depositLimit
   * Contains the methods to display the functionalities for the admin,setter and getter methods for dailyLimit and depositLimit,the checkDetails method for validating the login details of the admin, a method to return the details of customers
   * The above methods have been created keeping in mind the functionalities an admin has in a bank
   * All the data members and the methods have been binded together(encapsulation)
   * All the implementation is performed in these functions and very few details are visible to the user(abstraction)
   * The class extends a class and implements an interface(inheritance)
3. ATM:
   * interconnects all the classes and runs an infinite loop(which can be terminated only by the admin) for conduction all the transactions
4. Conversion:
   * Contains an array which stores the conversion rates from various currencies to rupees
   * Now with this class the functionality for deposits in various currencies could be added
5. Transaction(extends User\_Data and implements ATM\_Setup):
   * Represents the actual transaction
   * Contains data members which store the account number of the current user
   * Contains methods which resemble an actual transaction such as displaying the functionalities, selecting the type user at time of login,validating login vis Accountnumber password and also after that the OTP system,loggin out.
   * All the data members and the methods have been binded together(encapsulation)
   * All the implementation is performed in these functions and very few details are visible to the user(abstraction)
6. User\_Data:
   * Has the data members to store all the data of the customers which acts as a database
   * Includes methods to retrieve the data of the users update all the details including updating the bank balance
   * All the data members and the methods have been binded together(encapsulation)
   * All the implementation is performed in these functions and very few details are visible to the user(abstraction)

ATM\_System(interface):

* Contains methods which need to be implemented in Transaction class and Admin class.

ADVANTAGES:

* Performs all the functionalities of an atm machine such as deposit,transfer,withdrawal,view balance etc.
* Enables user to change some of the details related to their accounts
* The transaction limits can be set by the admin
* Money can be deposited in various form
* Money can be deposited in various currencies
* OTP system for double verification

DISADVANTAGES:

* Some part which is hard coded could have been done by creating some more methods in appropriate places

Diagram

Description automatically generated