ATM CASE STUDY

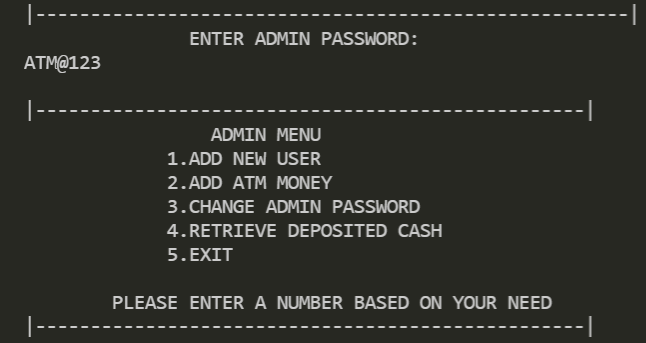
DONE BY: SIRISH SEKHAR

ROLL NUMBER: CS20B043

**DESIGN DESCISION:**

1.DUE TO MORE EMPHASIS ON THE OOPS PRINCIPLES IMPLEMENTATION OF DATABASE WITHIN TIMEFRAME WAS HARD SO TO COMPENSATE AN ADMIN FEATURE WAS ADDED TO INCORPORATE NEW USERS.

***(ADMIN PASSWORD-ATM@123)***

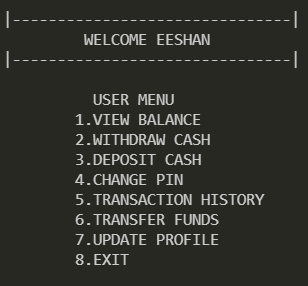


THIS IS TO MIMIC A DATABASE ENTRY SO THE ATM HAS ALL THE INFO REGARDING A USER’S DETAILS SUCH AS ACCOUNT NUMBER, PIN ETC: -

**ADD NEW USER** ALLOWS THE ADMIN TO ENTER THE DETAILS.

**ADD ATM MONEY** IS ALSO AN IMPORTANT UTILITY THAT ALLOWS ADMIN TO DEPOSIT CASH INTO THE ATM SO EVERY TIME MONEY RUNS OUT, THE NECESSARY AMOUNT CAN BE DEPOSITED THIS MIMICS THE REAL LIFE SCENARIO.

**RETRIEVE DEPOSITED CASH** REMOVES ALL THE USER DEPOSITS



2.USER CAN DO THE FOLLOWING FUNCTION, EACH OF THESE FUNCTIONS SERVE THE INTENDED PURPOSE AS NAME SUGGESTS.

3. A LIMIT OF 50000 WITHDRAWAL AND 50000 DEPOSIT HAS BEEN IMPLEMENTED, AS THIS IS STANDARD IN BANKS AND ATM CAN CONTAIN ONLY A LIMITED AMOUNT OF CASH IN OUR CASE 10,00,000 SO EACH PERSON WITHDRAWAL LIMIT IS 50000 LIKEWISE EACHPERSONS DEPOSIT LIMIT IS 50000 AS ATM CAN ONLY STORE UPTO 10,00,000.

4.FOLLOWING OOPS CONCEPTS HAVE BEEN IMPLEMENTED FOR DIFFERENT CLASSES.

a.) ENCAPSULATION

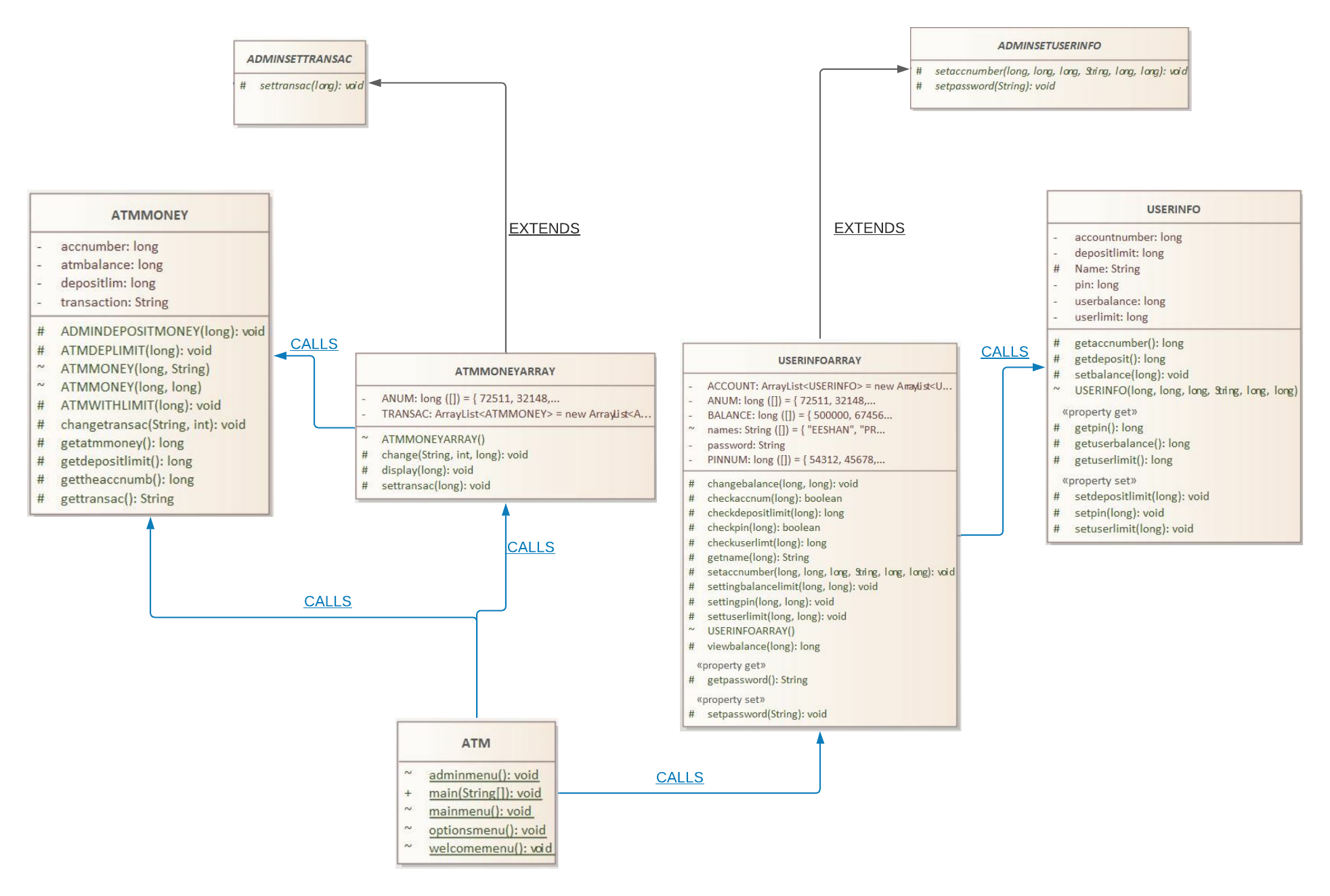
b.) POLYMORPHISM

c.) ABTRACTION

d.) INHERITANCE

***(COMMENTS IN CODE TELL WHERE EACH IS USED)***

***INTERFACES WERE NOT USED AS THEY ARE DEFAULT SET AS PUBLIC AND COULD POSE A SECURITY RISK***

CLASS DIAGRAM

THE ABOVE DIGRAM DISPLAYS ALL THE CLASSES AND THE PICTURE CAN BE

ZOOOMED IN FOR BETTER VIEW

**EACH CLASS SERVES A SPECIFIC PURPOSE**

**ATM** --->MAIN CLASS WHICH CALLS EVERY OTHER CLASS.

**USERINFO** ---> CLASS THAT CONTAINS NAME, ACCOUNTNUMBER, PIN BALANCE ETC: - **(ENCAPSULATION USED HERE)**

**USERINFOARRAY** --->HAS AN ARRAYLIST OF TYPE USERINFO TO DEAL WITH ALL USERINFO RELATED QUERIES SUCH AS BALANCE, PIN CHANGE ETC: -**(ABSTRACTION AND INHERITANCE USED HERE)**

**ATMMONEY** ---> CLASS FOR INFO RELATED TO AMOUNT OF MONEY IN ATM AND TRANSACTIONS OF ALL INDIVIDUALS. **(ENCAPSULATION AND POLYMMORPHISM USED HERE)**

**ATMMONEYARRAY** ---> HAS AN ARRAYLIST OF TYPE ATMMONEY TO DEAL WITH ALL USERINFO RELATED QUERIES SUCH AS TRANSACTION HISTORY AMOUNT OF MONEY LEFT IN ATM ETC: - **(ABSTRACTION AND INHERITANCE USED HERE)**

**ADMINSETTRANSAC** **AND ADMINSETUSERINFO** ---> ABSTRACT CLASSES TO HELP WITH ADMIN RELATED FUNCTIONALITIES.

***DETAILS OF EACH FUNCTION WITHIN THE CLASSES GIVEN IN COMMENTS***

**FAILED ASSUMPTIONS:**

1. KEEPING TRACK OF EACH TRANSACTION WITH HELP OF 2D ARRAY WAS MEMORY CONSUMING.

***Solution: -*** SO ALL TRANSACTIONS PERTAINING TO A CERTAIN USER WAS PUT INTO A STRING.

1. ADMIN FUNCTION WAS INITIALLY NOT INCLUDED SO AFTER MONEY IN ATM FINISHED NO WAY TO PUT IN MORE.
2. EVERYTHING WAS MADE PUBLIC INITIALLY WASN’T A GOOD PRACTICE AND EASILY MANIPULATION POSED A SECURITY RISK

**ADVANATGES OF USING THIS ATM**

1. USER FRIENDLY AND ALLOWS TRANSFER OF MONEY FROM ONE ACCOUNT TO ANOTHER
2. ADMIN FEATURE ALLOWS FOR EASY AND SECURE DEPOSITION OF CASH ONCE IT’S BEEN ALL USED UP.
3. SECURE AND SAFE AS EVERYTHING HAS BEEN MADE PRIVATE WITH REGARDS TO OOPS CONCEPT
4. ALL TRANSACTIONS ARE STORED IN A STRING! SO, MEMORY USAGE IS VERY LESS AND HENCE VERY EFFICIENT
5. ATM WITHDRAWAL AND DEPOSIT LIMITS ENSURE EVERY USER GETS A EQUAL AMOUNT OF CASH FROM THE ATM AND ONE PERSON CANT ENTIRELY TAKE OUT ALL CASH IN THE ATM.

**DISADVANATGES OF USING THIS ATM**

1. THE NEW USERS HAVE TO BE MANUALLY ADDED THOUGH DOESN’T HAPPEN IN REAL LIFE DUE TO NON-IMPLEMENTATION OF DATABASE
2. DOES NOT INCORPORATE SMALL DENOMINATIONS LIKE 10 OR 20 RUPPEES AND STARTS FROM A MINIMAL OF 100
3. PASSWORD SHOULD NOT BE SHARED OR ELSE IT MAY POSE A SECURITY RISK