# *ATM Program*

Student: Nativida Muhammad

SDEV 200 – Software Using Java

Instructor: Adam Bumgardner

Ivy Tech Community College

# Intro

Introducing the bank’s new web-based ATM application! It's the new one-stop solution for managing banking transactions online. Now, with just a few clicks, you have the capability to check balances, withdraw or deposit money, transfer funds, and view transaction history. The best part is that it is accessible from anywhere with an internet connection.  Our user-friendly interface makes banking hassle-free. Plus, you can always rest assured, your security is our #1 priority. Say goodbye to long queues, fuhgeddaboudit… say HELLO instead to convenient banking at your fingertips!

# Classes

### User Class:

*Attributes:*
username: String (Primary Key)
password: String
name: String
email: String
accountBalance: double

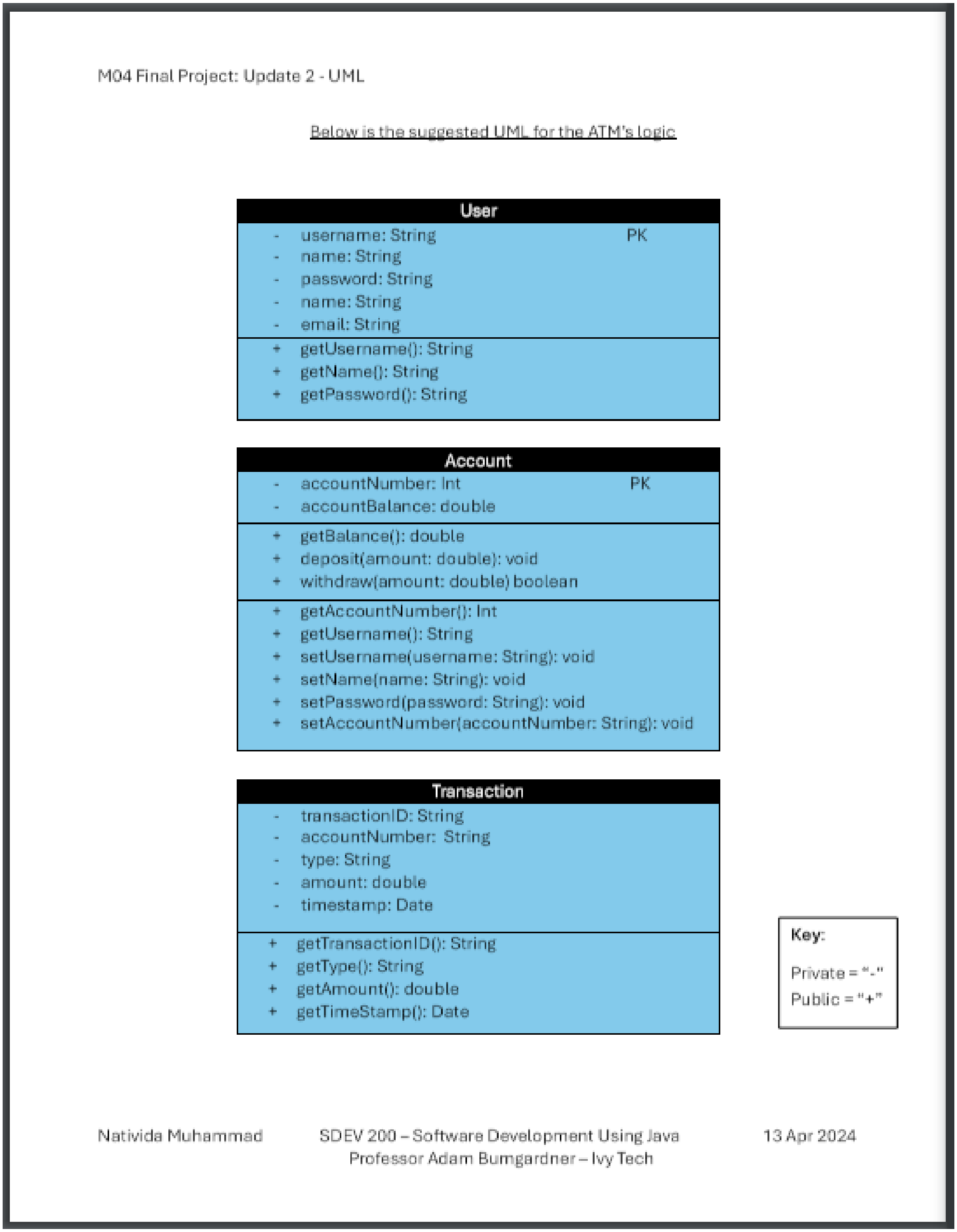
### Account Class:

*Attributes:*
accountNumber: int (Primary Key)
accountType: String
accountBalance: double
*Relationships:*
Each User can have multiple Accounts associated with their profile.

### Transaction Class:

*Attributes:*
transactionType: String
amount: double
timeStamp: Date
sourceAccount: Account
destinationAccount: Account
*Relationships:*
Loosely associated with both User and Account classes, as it records transactions made by users involving their accounts.

## UML Diagram



M04 Final Project: Update 2 - UML

Below is the suggested UML for the ATM's logic

| User | | |
|---|---|---|
| - | username: String | PK |
| - | name: String | |
| - | password: String | |
| - | name: String | |
| - | email: String | |
| + | getUsername(): String | |
| + | getName(): String | |
| + | getPassword(): String | |

| Account | | |
|---|---|---|
| - | accountNumber: Int | PK |
| - | accountBalance: double | |
| + | getBalance(): double | |
| + | deposit(amount: double): void | |
| + | withdraw(amount: double) boolean | |
| + | getAccountNumber(): Int | |
| + | getUsername(): String | |
| + | setUsername(username: String): void | |
| + | setName(name: String): void | |
| + | setPassword(password: String): void | |
| + | setAccountNumber(accountNumber: String): void | |

| Transaction | |
|---|---|
| - | transactionID: String |
| - | accountNumber:  String |
| - | type: String |
| - | amount: double |
| - | timestamp: Date |
| + | getTransactionID(): String |
| + | getType(): String |
| + | getAmount(): double |
| + | getTimeStamp(): Date |

Key:

Private = "-"
Public = "+"

Natividad Muhammad SDEV 200 – Software Development Using Java 13 Apr 2024
Professor Adam Bumgardner – Ivy Tech

## Testing

Functionality: This ATM Java program allows users to perform tasks such as logging in, checking their account balances, withdrawing and/or depositing money, transferring funds between accounts, as well as viewing transaction history. Finally, each functionality should be implemented *correctly*.

Security: Because this is a banking application, security is priority. The ATM program securely handles user authentication (e.g., password hashing) and transactions (e.g., validation, encryption).

User Interface: The GUI provides a user-friendly experience with all necessary navigation elements and buttons for different actions. It is intuitive and responsive.

Data Management: The program manages user data (e.g., username, password, account details) accurately and securely. It handles transactions and updates account balances accordingly.

Error Handling: The program handles errors gracefully, providing informative messages to users in case of invalid input or failed transactions.

Scalability and Performance: The program is designed to handle potential scalability requirements and perform efficiently even with a large number of users and transactions.

Testing: Thorough testing that was conducted, users can rest assured that the program will remain correct and robust under various scenarios.

## GitHub ATM Repository

https://github.com/Tiv-Barlow/Java---M08-Final-Project-Submission/tree/main