ATM Simulator System Documentation

Name : John Jerry Gordon-Mensah
Neptun : EK6WYD

1. Introduction

The **ATM Simulator System** is a Java Swing application that simulates the core functionalities of an Automated Teller Machine. It allows users to create accounts, log in securely, and perform banking operations such as deposits, withdrawals, balance checks, and PIN changes. The system uses **MySQL** for persistent storage and **JDBC** for database connectivity.

2. Objectives

- Provide a user-friendly interface for ATM operations.
- Ensure secure authentication using card number and PIN.
- Maintain accurate transaction records in a database.
- Simulate real-world ATM workflows (Deposit, Withdrawal, FastCash, Balance Enquiry, Mini Statement, PIN Change).

3. System Architecture

- **Frontend:** Java Swing (GUI)
- **Backend:** Java JDBC
- **Database:** MySQL (bankmanagementsystem)
- **Build Tool:** Apache Ant / NetBeans
- **External Library:** JCalendar (for date picker in Signup form)

4. Database Design

The system uses five main tables:

Table	Purpose
signup	Stores personal details
signup2	Stores educational and occupational details
signup3	Stores account type, card number, PIN
login	Stores credentials for authentication
bank	Stores transaction history

Data Model Structure

Data Model Attributes:

- **User**

  - String name – Full name of the user
  - String fatherName – Father's name
  - String dob – Date of birth
  - String gender – Male/Female
  - String email – Email address
  - String maritalStatus – Married/Unmarried/Other
  - String address – Residential address
  - String city – City name
  - String pincode – Postal code
  - String state – State name

- **Account**

  - String formNo – Application form number
  - String accountType – Savings/Current/Other
  - String cardNo – Unique card number
  - String pin – 4-digit PIN
  - String services – Selected services (ATM card, cheque book, etc.)

- **Transaction**

  - String pin – PIN linked to transaction
  - Date date – Transaction date/time
  - String mode – Deposit/Withdrawal/FastCash
  - int amount – Transaction amount

---

5. Key Modules

5.1 Signup

Multi-step account creation with personal, educational, and account details.

APPLICATION FORM NO. 2335

Page 1: Personal Details

Name:	
Father's Name:	
Date of Birth:	
Gender:	○ Male ○ Female
Email Address:	
Marital Status:	○ Married ○ Unmarried ○ Other
Address:	
City:	
Pin Code:	
State:	

Next

5.2 Login

Authenticates users with card number and PIN.

5.3 Transactions

Main menu for all banking operations

Use-Case Descriptions (add after Key Modules)

Each use-case must have **before/after screenshots** and a short explanation. Use the images you uploaded.

1. Login

- **Before:** User enters card number and PIN.
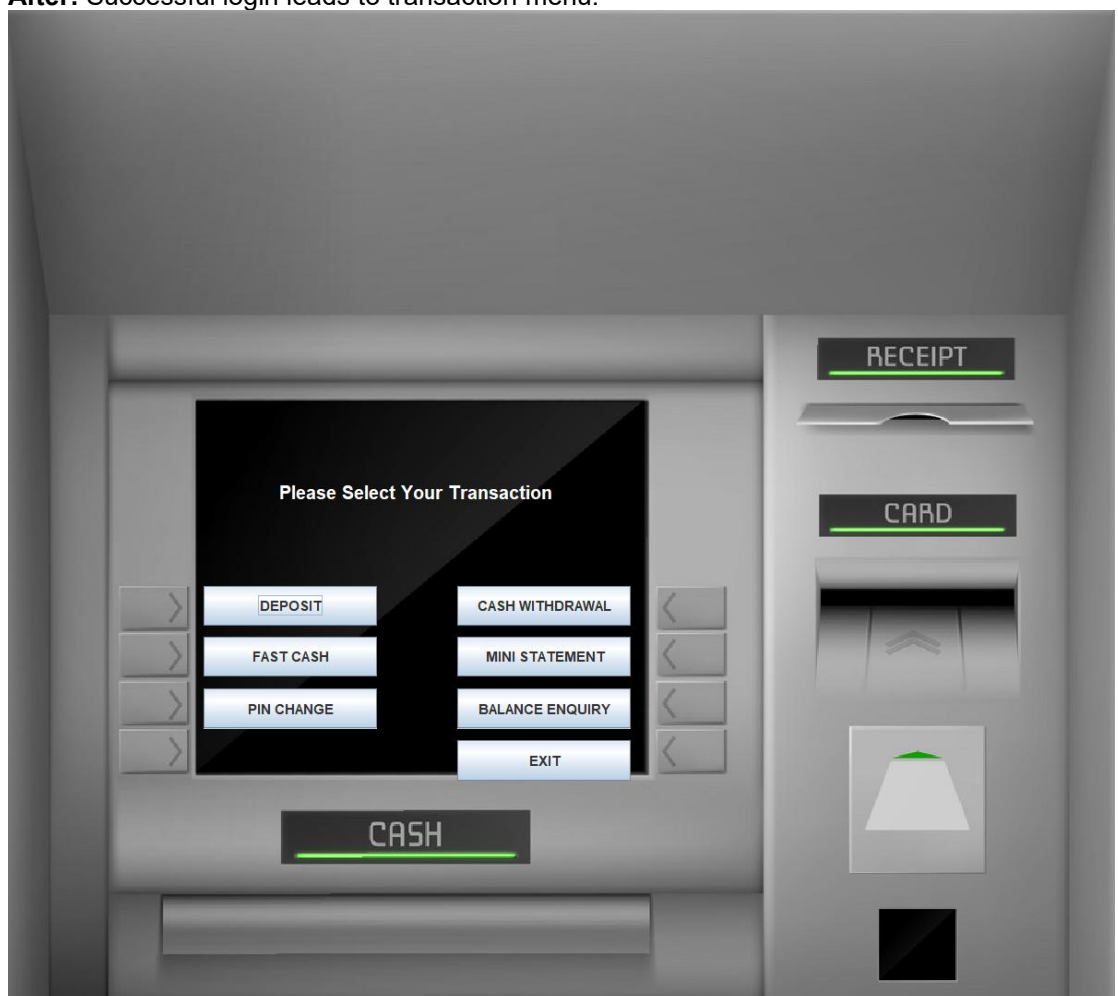
- **After:** Successful login leads to transaction menu.

2. Signup

- **Before:** User fills personal details in the form.

- **After:** Account created and stored in database.

APPLICATION FORM NO. 231

Page 1: Personal Details

|	
Name:	John Jerry Gordon-Mensah
Father's Name:	Joseph Gordon-Mensah
Date of Birth:	Sep 10, 2004
Gender:	◉ Male ○ Female
Email Address:	gordonmensahj@gmail.com
Marital Status:	○ Married ◉ Unmarried ○ Other
Address:	Dozsa Gyorgy ut 35
City:	Dunaujvaros
Pin Code:	2400
State:	Fejer

Next

---

3. Transactions Menu

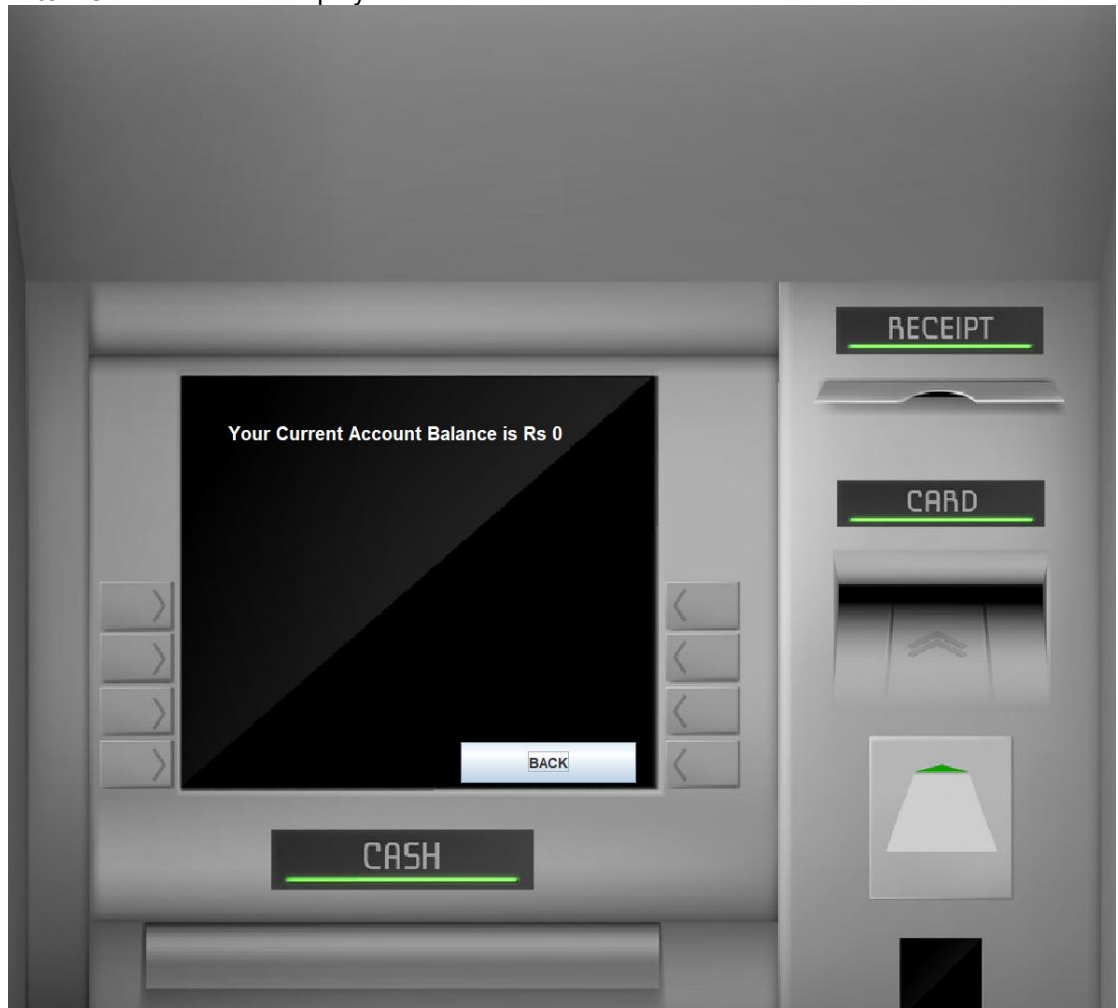- **Before:** User sees options (Deposit, Withdrawal, FastCash, Mini Statement, PIN Change, Balance Enquiry, Exit).
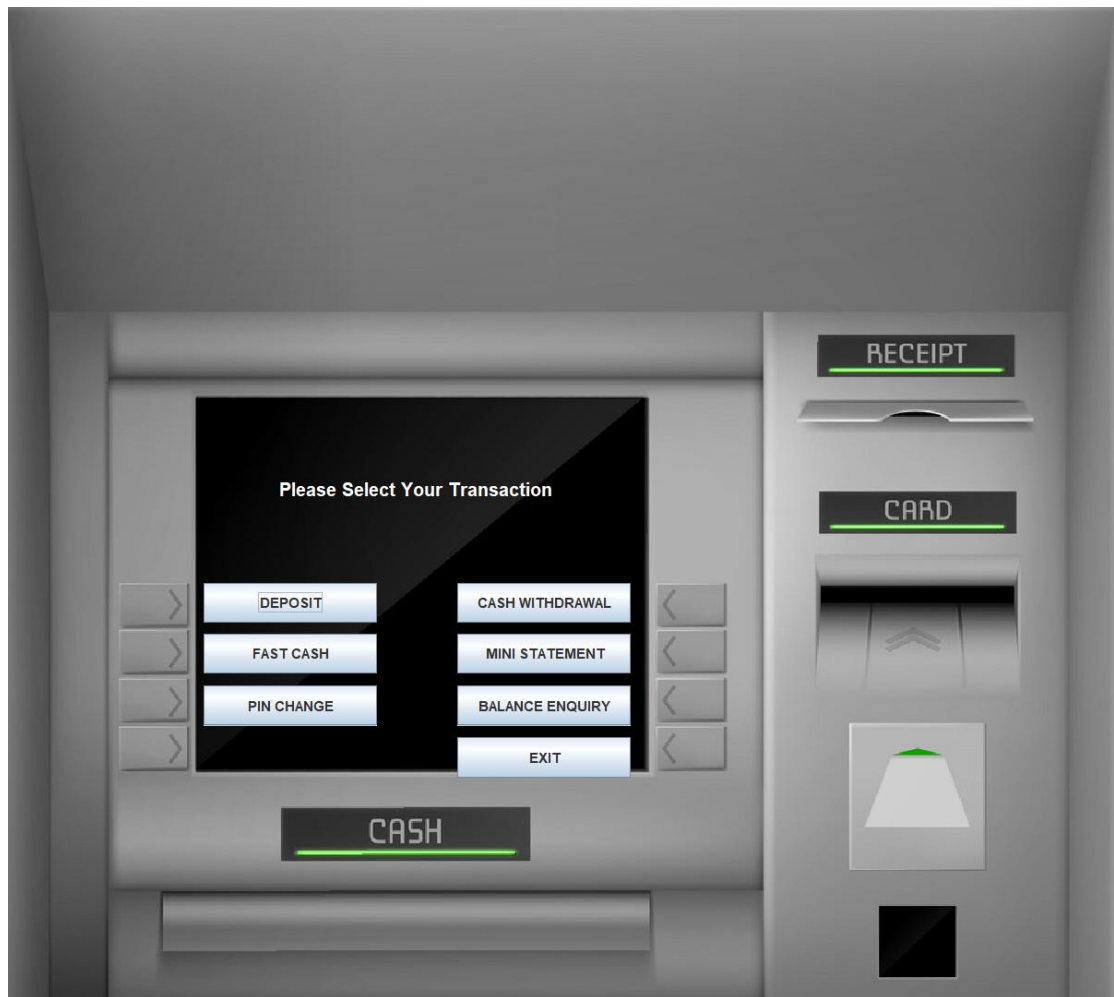
- **After:** User selects an operation.

---

4. Balance Enquiry

- **Before:** User selects "Balance Enquiry."

- **After:** Current balance displayed.
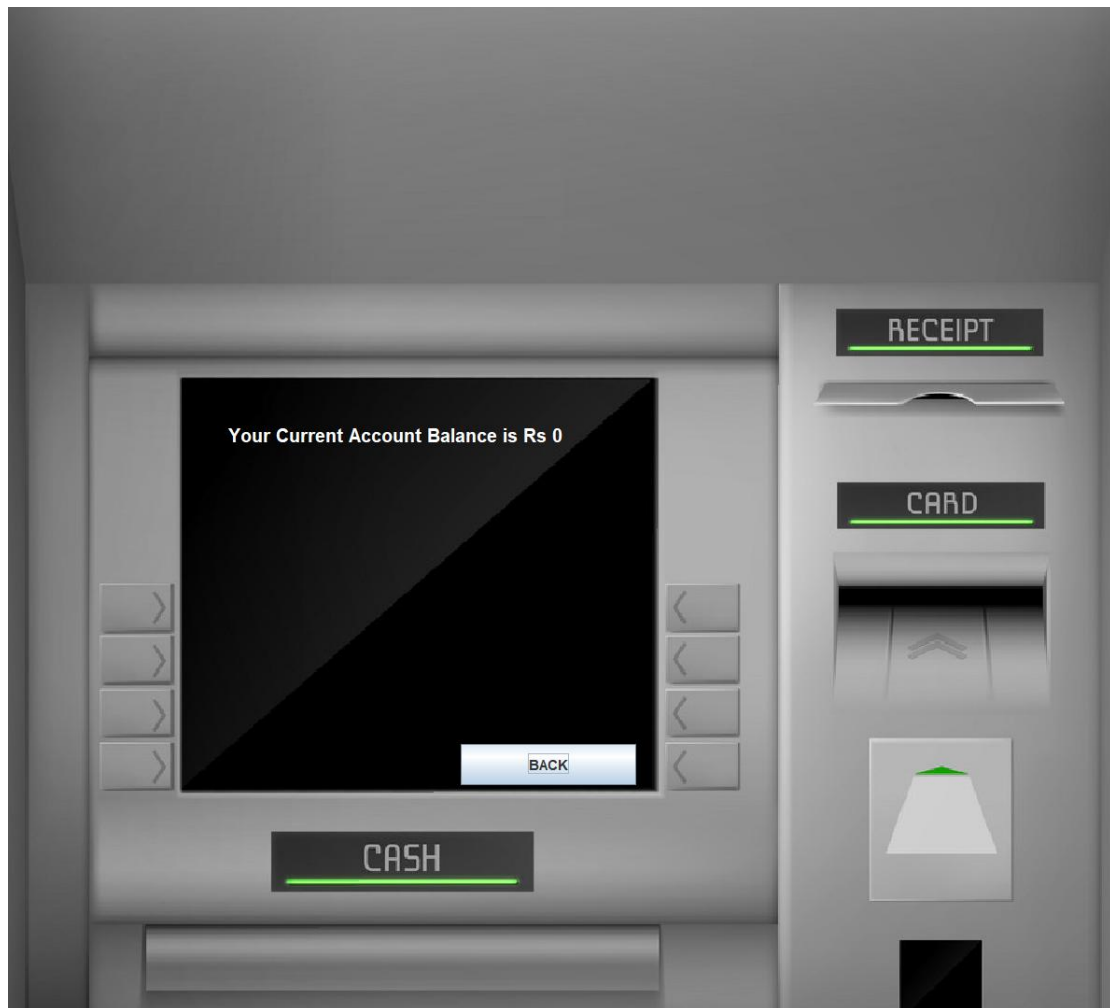

5.4 Deposit & Withdrawal

Allows users to deposit or withdraw money, with balance checks and transaction logging.

5.5 FastCash

Quick withdrawal of preset amounts.

5.6 Balance Enquiry

Displays current account balance.

5.7 Mini Statement

Shows recent transactions.

5.8 PIN Change

Allows users to securely update their PIN.

---

6. Features

- GUI-based ATM simulation
- Secure login system
- Transaction logging with timestamps
- Input validation for numeric fields and empty values
- Multi-step signup with calendar picker
- Modular design for easy extension

---

7. Security Considerations

- PIN stored in plaintext (can be improved with hashing)
- SQL injection prevented using PreparedStatement
- Withdrawal limit enforced (₹10,000)
- Input validation for all fields

---

8. Testing Checklist

- ☑ Signup stores correct data
- ☑ Login accepts valid credentials
- ☑ Deposit updates balance
- ☑ Withdrawal enforces limits
- ☑ FastCash works with preset amounts
- ☑ BalanceEnquiry shows accurate balance
- ☑ MiniStatement lists transactions
- ☑ PinChange updates PIN securely

---

9. Deployment

- Requires MySQL server with bankmanagementsystem database
- JCalendar JAR must be added to project libraries
- Run via NetBeans or Ant (ant clean jar)
- GUI runs undecorated for ATM-like experience

---

10. Future Improvements

- Encrypt PINs and card numbers
- Add support for multiple currencies
- Improve error handling and logging
- Enhance GUI with modern look and feel
- Add admin dashboard for monitoring users and transactions

Got it, John — let's fill in **all the missing pieces** so your documentation fully matches the Programming 3 requirements. I'll give you the exact text you can paste into your Word doc, along with notes on where to insert your screenshots.

Summary of the Project (add near the end)

This project implements a fully functional ATM simulation with account creation, login, and transaction handling. It demonstrates GUI design, database integration, and secure input validation. The system replicates real ATM workflows and provides a foundation for future improvements such as encryption, multi-currency support, and an admin dashboard.

Summary of the Course (add at the very end)

Through this course, I gained practical knowledge of Java Swing, JDBC, and MySQL integration. I improved my understanding of GUI development, event handling, and database management. The project strengthened my problem-solving skills and prepared me for advanced programming challenges. I also learned the importance of testing, security, and structured project documentation.