Report
on
Title: Graphical User Interface-Based ATM Simulation using Java Swing

Abstract
This project presents a secure, desktop-based ATM simulation system developed using Java Swing. The system mimics core ATM functionalities such as user login authentication, balance inquiry, money deposit, and withdrawal. It is designed with an intuitive graphical user interface (GUI), ensuring ease of use and interaction for users. The aim is to simulate real-world banking operations within a simple yet extensible software environment. This system is built entirely using the Java Standard Edition platform, utilizing built-in libraries without the need for external dependencies.

Objectives
- To design a desktop-based ATM interface using Java Swing.
- To provide user authentication based on ID and PIN entry.
- To implement basic ATM functions like balance check, deposit, and withdrawal.
- To enhance GUI usability while ensuring safe transactions.
- To demonstrate core object-oriented and event-driven programming concepts in Java.

Tools and Technologies Used

Technology	Purpose
Java (JDK 8+)	Core programming language
Java Swing	GUI development toolkit
IDE (IntelliJ/Eclipse/VS Code)	Code writing and debugging
Command Prompt/Terminal	Compiling and executing Java files

System Design
1. Architecture Overview
- **Input Layer:** User credentials and transaction inputs via GUI components.
- **Logic Layer:** Authentication logic, transaction processing, and balance updates.
- **Presentation Layer:** Swing components for all user interactions.

2. Components Used
- JFrame: Main window container
- JLabel, JTextField, JPasswordField: For input forms
- JButton: For interaction
- JOptionPane: For popup dialogues and messages
- Event Listeners for button actions

User Authentication
Upon launching the application, the user is prompted to enter a valid **User ID** and **PIN**. These credentials are hardcoded for demonstration purposes. Only authenticated users can proceed to the ATM operations panel.

Functional Modules

1. Login Screen

- Accepts user ID and PIN
- Verifies credentials
- Redirects to the main ATM menu on success

2. Main ATM Menu

- **Check Balance:** Displays the current account balance.
- **Deposit Money:** Prompts the user to enter an amount, adds it to the balance.
- **Withdraw Money:** Requests withdrawal amount, deducts it from the balance if sufficient.
- **Exit:** Closes the application with a farewell message.

Sample Workflow

1. User opens the ATM program.
2. Login screen appears.
3. User enters User ID: 123456, PIN: 7890.
4. Upon success, the main menu displays options.
5. User selects any desired transaction.
6. System processes and shows confirmation via dialog boxes.
7. User exits the session.

How to Run

1. Save the file as ATMGUI.java
2. Compile: javac ATMGUI.java
3. Run: java ATMGUI
Make sure Java is installed and added to your system path.

Key Learning Outcomes

- Developed a deeper understanding of GUI programming in Java.
- Applied object-oriented programming principles.
- Learned how to handle user events and input validation.
- Simulated a real-world banking scenario using code.

Future Scope

- Add support for multiple user accounts using file or database storage.
- Implement transaction history and mini statements.
- Integrate PIN change and account settings options.
- Enhance security with hashing and session timeouts.

Limitations

- Single user support only (static credentials)
- No persistent data storage (balance resets after program ends)
- No encryption for credentials (not production secure)

Conclusion

This Java-based GUI ATM system serves as a foundational project demonstrating how banking systems operate at a basic level. It highlights the capabilities of Java Swing in building functional desktop applications with proper flow control, GUI design, and user interaction. While it's designed for educational purposes, it offers a clear path for future enhancements and practical implementation.

Submitted By:
Name: ASHIRWAD SHUKLA
Branch: *B.Tech Computer Science & Engineering Specialization in Cyber Security*
Year: *Pre-Final*
Institution: *Dev Bhoomi Uttarakhand University, Dehradun*