Introduction to Entrepreneurship Development

(IED001)

B. TECH CSE

2nd YEAR

SEMESTER: 3rd

SESSION – 2024-25

Submitted By:

Sarthak (2023356277)

Vaibhavdeep Bhatt (2023523190)

Siddharth Singh (2023411968)

Section: CSR(G1)

Submitted To:

Mr. Himanshu Sharma

Assistant Professor

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Sharda School of Engineering and Technology (SSET)

Sharda University Greater Noida

TITLE – MULTI LAYER AUTHENTICATION FOR ATM'S

The project titled "**Multi-Layer Authentication for ATMs**" aims to enhance the security of Automated Teller Machines (ATMs) by implementing a multi-layer authentication system. The overall purpose of this project is to improve the security and user verification process to reduce the risk of fraudulent activities and unauthorized access.

Here's a brief overview of how such a system typically works:

1. **Traditional Authentication**: The project would start with the usual PIN-based authentication, where users enter their personal identification number (PIN) to access their accounts.

2. **Additional Authentication Layers**: To increase security, additional layers of authentication are introduced. These include:

- **Biometric Verification**: Incorporating fingerprint scanning, facial recognition, or iris scanning to verify the user's identity.

- **Two-Factor Authentication (2FA)**: Sending a one-time password (OTP) to the user's registered mobile number or email, which they need to enter to complete the authentication process.

- **Security Questions**: Security questions are used to verify a user's identity by asking questions that only the user should know the answers to.

4. **User Experience**: While adding multiple layers of security, the project also focuses on maintaining a user-friendly experience, ensuring that the additional security measures do not overly complicate the transaction process.

Objectives/Goals

1. **Enhance Security**: The primary goal is to strengthen the security of ATM transactions by implementing multiple layers of authentication. This reduces the risk of unauthorized access and fraud.

2. **Integrate Advanced Authentication Methods**: Incorporate various authentication technologies such as biometric verification (fingerprint, facial recognition), two-factor authentication (2FA), and security questions to provide a robust security framework.

3. **Reduce Fraudulent Activities**: By making it more difficult for unauthorized users to gain access, the system aims to minimize incidents of card skimming, identity theft, and other forms of ATM fraud.

4. **Improve User Verification**: Ensure that only legitimate users can access their accounts by verifying their identity through multiple means, enhancing trust in the ATM system.

5. **Maintain User Experience**: While adding layers of security, the system should be designed to be user-friendly and not overly cumbersome, ensuring a smooth transaction process for legitimate users.

Importance of the Work

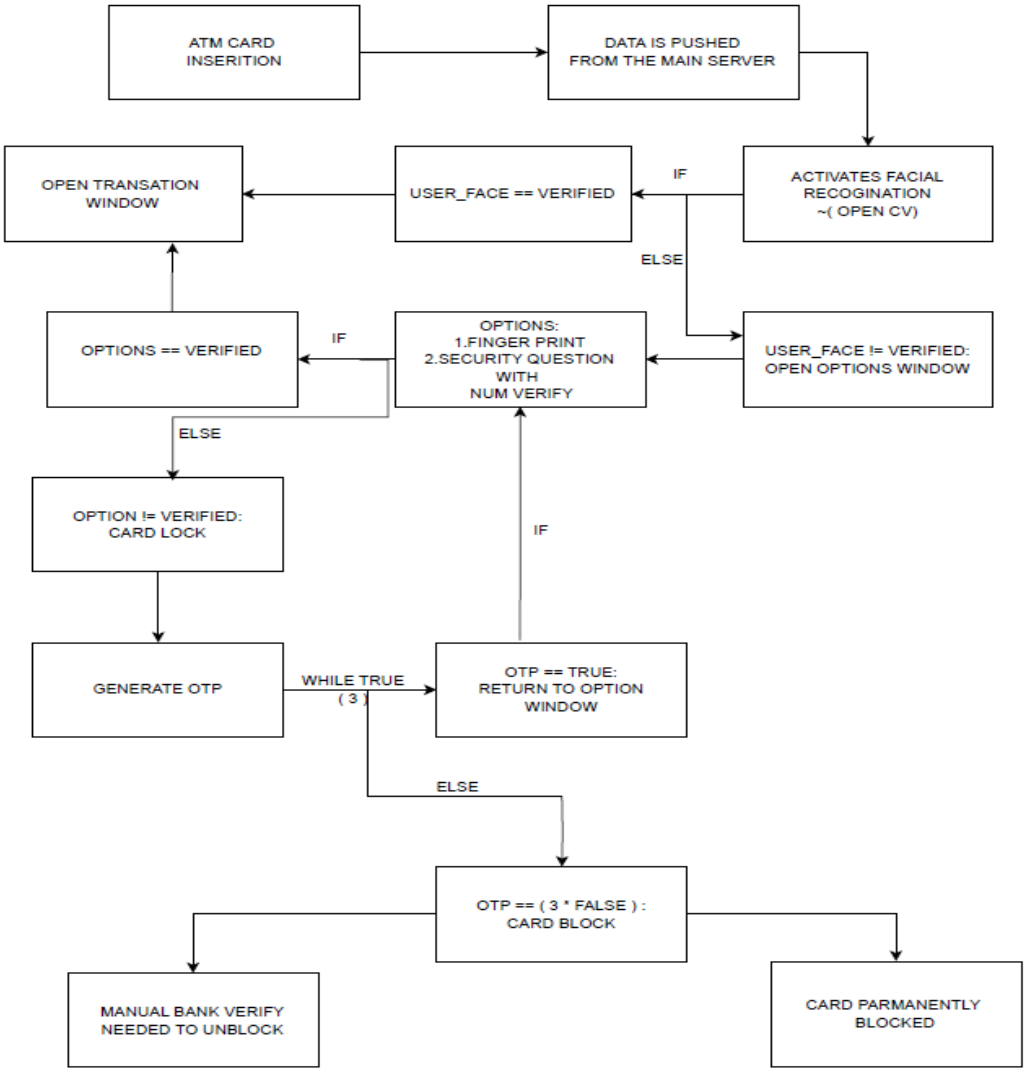
1. **Enhanced Security**: With rising concerns about cybercrime and financial fraud, strengthening ATM security is crucial. Multi-layer authentication provides a significant boost in protecting user accounts from various types of attacks.

2. **Reduction in Financial Losses**: By preventing unauthorized access and fraud, financial institutions can reduce potential losses associated with fraudulent transactions and identity theft.

3. **Adaptation to Evolving Threats**: As fraud tactics become more sophisticated, continuously updating security measures is essential to stay ahead of potential threats. Multi-layer authentication provides a flexible and adaptive approach to security.

In summary, the project's goals are to enhance security through a multi-faceted approach, ultimately leading to increased protection for users and reduced fraud. Its importance lies in addressing current security challenges and building a more secure and trustworthy ATM system.

ARCHITECTURAL DIAGRAM



Functioning -

1. **ATM Card Insertion**: When a user inserts their ATM card, it triggers the next steps.

2. **Data Push from the Main Server**: After the Card is inserted the data is pushed from the main server to the ATM system.

3. **Facial Recognition Activation (using OpenCV)**: The system activates facial recognition software (OpenCV) to verify the identity of the user based on their face.

- If the user's face is verified (matches the stored data), the system proceeds.

- If the user's face is not verified, the system provides alternative authentication options.

4. **Alternative Options if Facial Recognition Fails:**

- **Fingerprint Verification**: The user can opt to use fingerprint scanning for verification.

- **Security Question with Number Verification**: Another option is answering security questions and verifying with a numerical code.

- If any of these alternative methods are verified, the system proceeds.

- If verification fails, the card gets **locked**.

5. **Transaction Window**: Once the user is authenticated, the transaction window opens for normal operations.

6. **OTP Generation**: The system generates a One-Time Password (OTP) as an additional layer of security during transactions.

- If the **OTP** is correctly entered within **three** attempts, the system proceeds with the transaction.

- If the **OTP** is entered incorrectly **three** times, the card is **blocked**.

7. **Post-OTP Card Block:**

- If the card gets blocked, manual bank verification is required to unblock it.

- In case of repeated failures, the card may be permanently blocked, requiring further actions from the bank for recovery.

In summary, the system uses multiple layers of security (facial recognition, fingerprint, security questions, OTP) to authenticate the user, ensuring that the card is locked or blocked only after several failed attempts, protecting the account from unauthorized access.

Challenges

1. **Facial Recognition Accuracy**: The system relies on OpenCV for facial recognition, which may face challenges like false positives (verifying the wrong person) or false negatives (failing to recognize a legitimate user). Variations in lighting, user appearance (e.g., wearing glasses or masks), or camera quality could impact its effectiveness.

2. **Fallback Authentication Methods:**

- If facial recognition fails, the system offers fingerprint or security question verification. However, **fingerprint scanners** might face issues with damaged or unclear fingerprints, and **security questions** may not be secure enough.

- Additionally, users who forget security question answers or face fingerprint scanner malfunctions may experience frustration.

3. **Multiple Points of Failure:**

- Each authentication step introduces a point of potential failure. If any of the systems (facial recognition, fingerprint scanning, OTP generation) experience technical issues, users may not be able to authenticate themselves.

- This could lead to **unnecessary card locking or blocking**.

4. **User Experience Issues:**

- The overall user experience may suffer if users face delays or repeated authentication failures. Having to switch between multiple authentication methods may frustrate users, especially if they feel the system is overly complex.

5. **Manual Bank Verification Process:**

- Once a card is blocked, the system requires **manual bank verification** to unblock it. This could be a lengthy process and may inconvenience users, especially in critical situations where they need immediate access to funds.

Completed Tasks -

1. Read and Store Face Data: The system captures the user's face, converts it into a `.jpg` file, and stores it in a directory named `face_data` for future use.

2. Initialize Face Unlock: During authentication, the system compares the captured face with stored images and, upon a match, displays a message "Welcome, User."

3. Store Data for a Face: Each new user's face data is saved in the `face_data` directory, ensuring it is available for future logins.

4. Face Unlock and Registration Accessible via Tkinter GUI: The face registration and face unlocking functions are integrated into a Tkinter GUI accessible from the `main.py` file.

5. Login Success Message: Upon successful face match, the system shows a confirmation message, "Login Successful," on the GUI.

Team Contributions -

1. Siddharth - Integrated the face registration and unlocking functionality into a Tkinter GUI, ensuring it was accessible from the `main.py` file. Managed the overall coordination of the project.
2. Vaibhavdeep - implemented the logic for comparing the captured face with stored images and displaying the "Welcome, User" message upon a match.
3. Sarthak - Worked on capturing the user's face, converting it into `.jpg` format, and saving it to the `face_data` directory.

Remaining Tasks -

1. Implement Fingerprint Authentication

- Integrate a **fingerprint scanner** module to capture and store fingerprints during the registration phase.
- Implement a fingerprint matching algorithm for authentication when facial recognition fails.

2. Implement Security Question Authentication

- During registration, ask the user to set up **security questions** and store the answers securely (preferably encrypted).
- Implement a logic to present security questions when both facial and fingerprint recognition fail.
- Add this as an option in the Tkinter GUI, allowing users to authenticate by answering security questions.

3. OTP Generation and Validation

- Generate and send an OTP to the user via email or SMS as an additional verification layer.
- Add this OTP validation step after face or fingerprint authentication for enhanced security.
- Display the OTP entry option in the Tkinter GUI.

4. Test and Debug Authentication Flow

- Thoroughly test the **entire authentication flow**, including facial recognition, fingerprint scanning, security questions, and OTP generation.
- Handle edge cases, like when a user fails multiple authentication attempts or when hardware (fingerprint scanner) is unavailable.