JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Advanced Security System for ATM Software with Facial Recognition and OTP with Shuffle Keypad Feature

[1]Shubham Sambhaji Veer, [2]Omkar Hanumant Dalvi, [3]Omkar Pandurang Babar, [4]Pratik Shivaji Babar [1]
Department of Artificial Intelligence & Data Science,
[1]Suman Ramesh Tulsiani Technical Campus Faculty of Engineering, Pune, India.

Abstract : With increasing cases of ATM fraud and unauthorized access, this research presents an advanced security system incorporating Low Rank Representation Twin Tensor Kernel Deep Learning (LRR-TTK DL) and facial recognition. The proposed system integrates DeepFace-based authentication, Support Vector Machine (SVM) classification, and a shuffle keypad mechanism to enhance security. Additionally, an OTP verification module serves as a fallback measure. The architecture is designed to ensure robust authentication, fraud prevention, and user accessibility through a Tkinter-based graphical user interface (GUI) for both users and administrators. Experimental evaluation confirms the system's effectiveness in improving security measures for ATM transactions**.**

***IndexTerms* - Component,formatting,style,styling,insert.**

I. INTRODUCTION

Automated Teller Machines (ATMs) play a crucial role in financial transactions, yet they remain highly vulnerable to fraudulent activities such as card skimming, PIN theft, and unauthorized withdrawals. Traditional authentication methods, primarily reliant on PINs, have proven insufficient in countering these security threats. To address these challenges, this research proposes an AI-driven security system that enhances ATM authentication through facial recognition, machine learning algorithms, and advanced security measures. Automated Teller Machines (ATMs) are essential in modern banking, providing 24/7 financial services. However, conventional authentication methods such as PIN codes and magnetic stripe cards are increasingly vulnerable to fraudulent activities, including skimming and phishing attacks. This necessitates implementing sophisticated security measures to safeguard users. This study proposes a security model integrating biometric authentication through facial recognition, OTP verification, and a dynamic keypad system. Facial recognition offers biometric-based authentication, ensuring only legitimate users access their accounts. OTP verification adds an extra security layer, requiring users to input a time-sensitive code sent to their registered mobile numbers. Additionally, the dynamically shuffled keypad mitigates risks associated with visual hacking, further enhancing security. By combining these technologies, the proposed system creates a robust security framework that minimizes fraudulent transactions while maintaining user convenience.

This research focuses on developing a Advanced Security System for ATM Software with Facial Recognition and OTP with Shuffle Keypad Feature

Key Components And Working

ATM Protection: Security measures to prevent fraud, unauthorized access, and financial crimes at ATMs.

Biometric Verification: Authentication using unique biological traits like fingerprints or facial recognition for enhanced security.

Facial Recognition: AI-based identity verification using facial features to prevent unauthorized ATM access. The system should be able to authenticate the user based on their facial features. Upon initiating a transaction, the user's face will be captured by the ATM camera and sent to the backend for recognition. The system will compare the captured face against the stored face templates in the database. The system will employ facial recognition technologies like AWS Rekognition, Azure Face API, or OpenCV. Once the user's face is captured, the backend compares it to the face templates stored during the registration phase. If the system detects a match, it returns the corresponding user ID; otherwise, it prompts the user to retry facial recognition. This provides a contactless and secure means of verifying the user before proceeding with the transaction.

OTP Security: One-time passwords (OTPs) provide an extra authentication layer for secure transactions. Once the facial recognition successfully identifies the user, an OTP (One-Time Password) is generated and sent to the registered mobile number or email address of the user. The OTP will be time-sensitive and only valid for a limited time, typically 5 minutes. The OTP will be generated using a secure method like TOTP (Time-based One-Time Password), and services like PyOTP or OTPAuth will be used for generation. The system will use Twilio or SendGrid to send the OTP to the user via SMS or email. The user will enter the OTP into the ATM's shuffled numeric keypad interface. The backend will verify the entered OTP against the generated one. If it matches and is within the validity period, the user is granted access to the ATM functions.

User Session Management : After the user is authenticated via facial recognition and OTP verification, a secure session will be created, allowing the user to perform transactions. The session will be maintained and tracked using JWT (JSON Web Tokens). JWT will be used for session management to ensure that the user remains logged in for the duration of the transaction. The JWT will include an encrypted token containing user-specific data and an expiration timestamp. The backend will authenticate each subsequent request made by the user using this token to verify that the user is authorized to perform the transaction.

Dynamic Keypad: A shuffled on-screen keypad prevents PIN theft through visual hacking or keylogging. The system will display a randomized, dynamic numeric keypad for OTP entry. This is to prevent shoulder surfing and other security threats by ensuring the numbers on the keypad are different every time the user interacts with it. The keypad will be rendered using JavaScript or React.js, and each time the user accesses the keypad to enter their OTP, the numbers will be shuffled in a random order. This prevents any potential attacker from identifying which number corresponds to which key. Additionally, the input will be validated by the backend to ensure that only the correct OTP is accepted.

Transaction Processing and Logging : Once authentication is successful, the system should allow the user to perform various ATM functions such as withdrawing money, checking balance, transferring funds, etc. All actions performed by the user will be logged securely for audit purposes. After successful authentication, the ATM interface will allow the user to choose from a set of pre-defined functions like cash withdrawal, balance inquiry, or money transfer. Every user action will be recorded in the backend database for traceability and security. These logs will include information about the time, the type of transaction, and the amount involved. This ensures that the system maintains a high level of transparency and accountability for every transaction.

Secure Communication : All communication between the user's ATM interface and the backend system must be encrypted to protect sensitive data like OTPs and user credentials. The system will implement SSL/TLS encryption to ensure that all data transmitted between the ATM terminal and the backend is securely encrypted. This prevents eavesdropping, man-in-the-middle attacks, and ensures that sensitive data such as OTPs and user information remain confidential during transmission.

Fraud Prevention: Techniques and technologies to detect and prevent ATM fraud, ensuring secure financial transactions.

MODEL SELECTION: - The Software Development Life Cycle (SDLC) consists of different models used for structured software development. Here are key SDLC models:

Waterfall Model – A step-by-step, linear approach where each phase (planning, design, development, testing, deployment, maintenance) is completed sequentially.

Agile Model – An iterative model that emphasizes flexibility, collaboration, and continuous improvement through small, incremental releases.

V-Model (Validation & Verification) – A structured model where each development phase has a corresponding testing phase, ensuring high quality.

Spiral Model – A risk-focused model combining iterative development with systematic risk assessment and refinement in repeated cycles.

Iterative Model – Development occurs in multiple iterations, improving functionality progressively based on feedback.

RAD Model (Rapid Application Development) – Focuses on rapid prototyping and fast development with minimal planning.

Algorithms Used

1.Resnet : ResNet is used for facial feature extraction in the ATM security system, ensuring high accuracy even under different lighting conditions, angles, or occlusions. Its residual connections help in training deep networks without vanishing gradients, making it ideal for real-time facial authentication at ATMs. By leveraging ResNet-based facial recognition, the system adds a contactless and secure authentication layer, reducing reliance on easily compromised PIN-based security.

2. LRR-TTK DL (Low-Rank Representation with Tensor Train Kernel for Deep Learning) : LRR helps in reducing redundancy in high-dimensional data by representing features in a compact, low-rank form. TTK is used to capture complex relationships within the data while maintaining computational efficiency. This method is particularly useful in face recognition tasks, as it improves accuracy by preserving important facial features while reducing noise. By integrating LRR-TTK with deep learning, the model achieves higher classification accuracy and robustness in recognizing faces, making it suitable for ATM security systems where facial authentication is critical.

3. DeepPixBiS : Detects spoof attacks using pixel-wise supervision, preventing fraud via printed photos, masks, or deepfake videos. Uses CNN-based feature extraction to analyze texture, lighting inconsistencies, and depth cues before allowing authentication. Uses CNN-based feature extraction to analyze texture, lighting inconsistencies, and depth cues before allowing authentication. Runs before facial recognition to verify liveness; if a spoof is detected, access is denied immediately.

4. LSTMs: Detects suspicious activities by monitoring withdrawal amounts, frequency, and ATM locations over time. Identifies potential fraud, such as large withdrawals from an unusual location, and triggers additional authentication. Even if facial recognition is bypassed, suspicious transactions require extra verification (OTP, security questions, manual review). Continuously learns user transaction patterns to improve fraud detection over time, reducing false positives and enhancing security.

Activity Sequence :

- User Initiates Session
- ATM System: Start Session
- ATM System Requests Face Verification
- Facial Recognition API Verifies Face
- Face Verification Result Returned
- If Verification Fails: Notify Admin
- ATM System Requests OTP
- OTP Service Sends OTP
- User Receives and Inputs OTP
- ATM System Verifies OTP
- If OTP Fails: Notify Admin
- ATM System Displays Shuffled Keypad
- User Inputs PI
- ATM System Validates PIN
- If PIN Fails: Notify Admin
- ATM System Grants Access
- Session Ends

Activity Diagram: Facial Recognition & OTP Authentication for ATM Access

- **Start**
- **Capture User Face**
- **Verify Face Using API**
- **If valid** → proceed to **Validate Input**
- **If invalid →→ Send OTP to Registered Mobile**
- **Enter OTP**
- **Shuffle Keypad**
- **Validate Input**
- **Decision: Input Validation Outcome**
- **If valid → Grant Access**
- **If invalid → Access Denied**
- **End**

4.4 UML Unified Modeling Language Diagram

4.4.1 Use Case Diagram

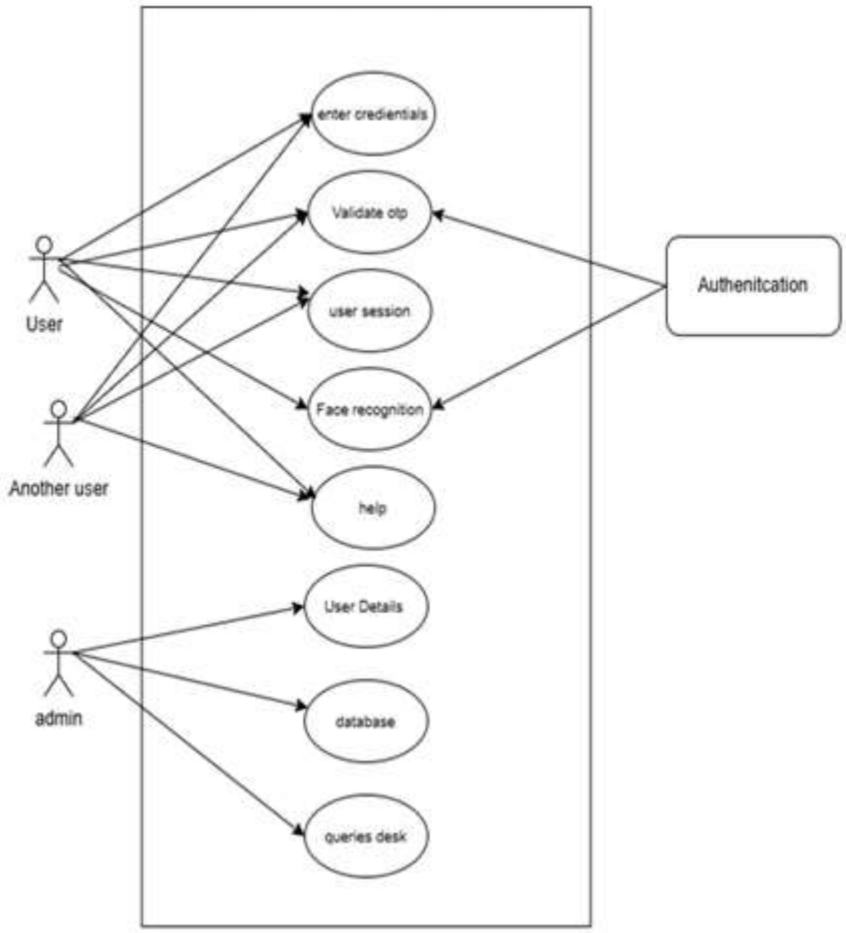


Figure 4.6:Use Case Diagram

Use Cases (Functionalities):

The use case diagram illustrates the interactions between different types of users—namely, the **User**, **Another user**, and the **Admin**—with the system's authentication and management functionalities. Both the User and Another user are able to initiate the authentication process by **entering credentials**, such as a username and password. Once the credentials are submitted, the system proceeds with an **OTP (One-Time Password) validation** step, which requires the users to input a code sent to their registered mobile number or email. An additional layer of security is provided through **face recognition**, where the system verifies the user's identity by analyzing facial features.

After successful validation, the system establishes a **user session**, allowing the user to access the ATM functions securely for the duration of the session. If any issues arise during the process, users can access the **help** feature for guidance or support. On the administrative side, the **Admin** has distinct responsibilities and privileges. The Admin can manage **user details**, access and maintain the **database**, and address customer issues through the **queries desk**.

The system boundary, labeled **Authentication**, encapsulates all these use cases, defining the scope of functionalities handled by the authentication system. The diagram clearly represents how users interact with the system during the authentication process, while administrators are responsible for managing backend operations and supporting user activities. Overall, this use case diagram provides a clear overview of the system's workflow and highlights the various roles and interactions involved in ensuring secure and efficient ATM access.

Overall Flow:

- Users begin by **entering credentials**.
- The system requires **OTP validation** and **face recognition** for enhanced security.
- Once authenticated, a **user session** is created, allowing them access to services.
- If needed, users can access **help**.
- Admins have additional privileges for **user management**, **database control**, and **query resolution**.

IMPLEMENTATION :

User Panel Flowchart :

```
Start
  ├───> Face Recognition
  |     ├───> Recognized ───> Transaction Menu
  |     └───> Not Recognized ───> OTP Verification
  |           ├───> Enter OTP
  |           |     ├───> Correct OTP ───> Transaction Menu
  |           |     ├───> Incorrect OTP ───> Retry (Max 3)
  |           |     └───> Timeout ───> Return to Start
  |
  ├───> Transaction Menu
  |     ├───> Deposit
  |     |     ├───> Enter Amount
  |     |     ├───> Confirm & Update Balance
  |     |     ├───> Send SMS Notification
  |     |     └───> Return to Menu
  |     |
  |     ├───> Withdrawal
  |     |     ├───> Enter Amount
  |     |     ├───> Check Balance
  |     |     ├───> Confirm Withdrawal
  |     |     ├───> Dispense Cash
  |     |     ├───> Send SMS Notification
  |     |     └───> Return to Menu
  |     |
  |     ├───> Check Balance
  |     |     ├───> Display Current Balance
  |     |     └───> Return to Menu
  |     |
  |     ├───> PIN Change Request
  |     |     ├───> User Enters New PIN
  |     |     ├───> Confirm PIN Change
  |     |     ├───> Send Request to Admin
  |     |     └───> Return to Menu
  |     |
  |     └───> Exit ───> Return to Start
```

Admin Panel Flowchart :

```
Admin Login
  ├───> Admin Dashboard
  |     ├───> View Users
  |     |     ├───> Search User
  |     |     ├───> Select User
  |     |     |     ├───> Edit Details ───> Update Database ───> Return
  |     |     |     ├───> Block User ───> Confirm ───> Return
  |     |     |     ├───> Delete User ───> Confirm ───> Return
  |     |     └───> Return
  |     |
  |     ├───> Add User
  |     |     ├───> Enter User Details
  |     |     ├───> Capture Face Data
  |     |     ├───> Store in Database
  |     |     └───> Return to Dashboard
  |     |
  |     ├───> Complaints/Support
  |     |     ├───> View Complaints
  |     |     ├───> Respond or Mark Resolved
  |     |     └───> Return to Dashboard
  |     |
  |     ├───> System Logs
```

```
    |     |       ├───> View Login/Transaction Logs
    |     |       ├───> Filter/Search Logs
    |     |       └───> Return to Dashboard
    |     |
 |      └───> Exit ───> Logout
```

Functions & Workflows Breakdown :

1)User Authentication Workflow : This workflow ensures secure access using **Face Recognition, OTP, and PIN verification**.

1.1 Start Screen (ATMApp.create_main_screen)

- Displays **Welcome Message**
- Options:
  - **Start Face Recognition**
  - **Guide (Help)**
  - **Support (Contact Admin)**

1.2 Face Recognition (authenticate_user)

- Captures the **user's live image** using OpenCV.
- **Detects face and checks liveness** (blink detection).
- If **recognized**, → **proceed to transaction menu**.
- If **not recognized**, → **OTP Verification**.

1.3 OTP Verification (show_phone_verification_panel)

- **User enters registered phone number**.
- **OTP is generated & sent via Twilio** (send_otp).
- **User enters OTP**:
  - ☑ **Correct OTP** → Proceed to transaction menu.
  - ✕ **Incorrect OTP** → Retry (Max: 3 attempts).
  - ⌛ **Timeout** → Restart process.

1.4 PIN Authentication (show_atm_pin_panel)

- **User enters a 6-digit PIN** (shuffled keypad).
- **PIN is verified** against the **hashed PIN** in the database.
  - ☑ **Correct PIN** → Allow transaction.
  - ✕ **Incorrect PIN** → Retry (Max: 3 attempts, then block account for 24 hours).

2) Transaction Workflow

Once authenticated, the user can **Deposit, Withdraw, or Check Balance**.

2.1 Transaction Menu (show_transaction_menu)

- Options:
  - **Deposit**
  - **Withdrawal**
  - **Check Balance**
  - **PIN Change Request**

2.2 Deposit (show_deposit_panel)

- **User enters amount** (Multiples of ₹100, Max: ₹10,000).
- **Balance is updated** in the database.
- **SMS Notification** is sent confirming the deposit.

Withdrawal (show_withdraw_panel)

- **User enters amount**.
- **Balance is checked** before approval.
- If **sufficient balance**:
  - Amount is **deducted**.
  - **Cash is dispensed**.
  - **SMS Notification** is sent.
- If **insufficient balance**, show error.

2.4 Check Balance (check_balance_action)

- Retrieves and **displays the current balance** from the database.

Check Balance (check_balance_action)

- Retrieves and **displays the current balance** from the database.

2.5 PIN Change Request (show_transaction_menu)

- **User requests PIN change**.
- **Admin manually approves and updates** the PIN.

3) Admin Panel Workflow

Admins manage users, logs, and system settings.

3.1 Admin Login (admin_panel)

- Verifies admin credentials before allowing access.

3.2 Dashboard (admin_panel.show_main_menu)

- Options:
  - **View Users**
  - **Add User**
  - **Update User**
  - **Block/Delete User**
  - **Complaints/Support Panel**
  - **System Logs**

3.3 User Management

- **View Users (get_registered_users)**
  - **Search users by name or phone.**
  - **View user ID, name, phone, status.**
- **Add User (register_user)**
  - **Enter details (Name, Phone, ATM PIN).**
  - **Capture face data.**
  - **Store face encoding in database.**
  - **Save hashed ATM PIN.**
- **Update User (update_user_details)**
  - **Modify name, phone, PIN.**
  - **Option to re-register face.**
- **Block/Delete User (block_user, delete_user)**
  - **Blocked users cannot access the ATM.**
  - **Deleted users are removed permanently.**

3.4 Complaints & Support (open_complaints_panel)

- **Users can submit complaints**.
- Admin can **view, respond, and resolve** issues.

3.5 System Logs (open_user_checker)

- **Logs unknown face recognition attempts**.
- **Logs failed login attempts & transactions**.

Final Workflow Summary

Technologies Used

Face Recognition & Liveness Detection

Library:face_ recognition, dlib, OpenCV, torch

Model:

LRR-TTK(Low-Rank, Representation, with Twin Tensor Kernel)

Purpose:

- Detect and recognize faces in real-time.
- Ensure **liveness detection** using **blink detection** (avoids spoofing with images).
- Use **Deep Learning (DL) to improve accuracy** and reduce false positives.

📌 **LRR-TTK (Low-Rank Representation with Twin Tensor Kernel)**

- **Low-Rank Representation (LRR):** Improves feature extraction for face encoding.
- **Twin Tensor Kernel (TTK):** Enhances **generalization** of face recognition, even in **low-light** and **angle-variation** scenarios.
- **Usage:** This DL-based model helps **store more accurate face encodings** and improves **authentication security**.

2)Database & Data Storage

Database: MySQL (XAMPP Server)

Tables Used:

- users_detail: Stores **user information (face encoding, PIN, phone, balance, status)**.
- transaction_logs: Stores **deposits, withdrawals, and balance updates**.
- unknown_entries: Logs **unrecognized face attempts**.

Security Measures: ✓ ATM PINs stored using bcrypt hashing (no plaintext storage).
✓ **Face encodings stored securely** as JSON-serialized vectors.

3) OTP-Based Authentication

Library: Twilio

Purpose:

- **Backup authentication method** if face recognition fails.
- **15-second OTP validity** with **resend option**.
- **Auto-expiry** and **retry limit (3 attempts)** to prevent abuse.

4)GUI & User Interface

Library: Tkinter (Python GUI)**Features:**

- **Admin Panel:** Manage users, logs, complaints.
- **User Panel:** ATM interface for authentication & transactions.
- **Shuffled Keypad:** Prevents PIN entry pattern detection.

GUI Components: ✓ **Face Recognition Window** → Detects face & confirms identity.
✓ **OTP Entry Window** → Takes OTP input with **15s timer**.
✓ **Shuffled PIN Keypad** → Prevents shoulder surfing attacks.
✓ **Transaction Menu** → Options for deposit, withdrawal, and balance check.

5) Deep Learning & Machine Learning

Model: LRR-TTK Face Recognition Model (Implemented in torch, numpy)

Purpose:

- **Enhance ATM authentication accuracy**.
- **Reduce false positives** with low-rank feature extraction.
- **Handle multiple face angles & real-time detection**.

📌 **Why LRR-TTK?** ✓ **Better recognition accuracy** than HOG/CNN-based models.
✓ **Handles varied lighting & occlusions well**.
✓ **Prevents fraud via liveness checks (blinking detection)**.

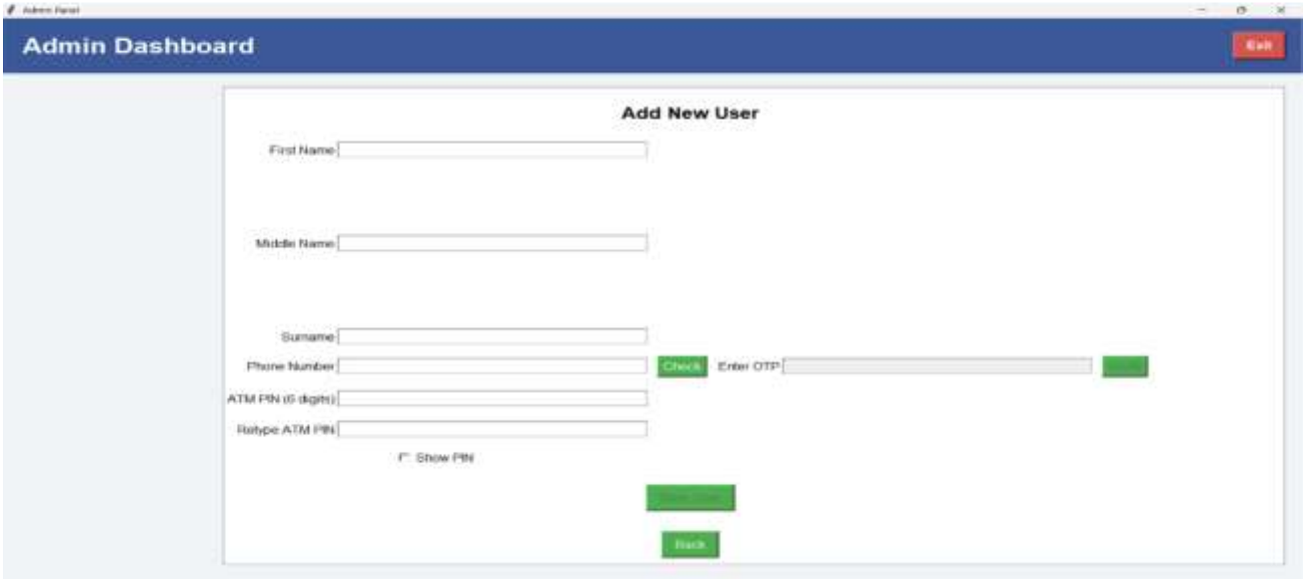
6) Security & Encryption

Library: bcrypt (For PIN encryption)

Security Measures:

- **ATM PINs are hashed** (bcrypt, salt added).
- **Users are blocked after 3 incorrect PIN attempts** (24-hour ban).
- **Transaction limits set** (Max ₹10,000, Multiples of ₹100).
- **Database secured with role-based access** (Admin vs User).

Final System Flow : Face Recognition → OTP (If needed) → PIN Entry → Transaction → Balance Update → Secure Logging



Add New User

First Name

Middle Name

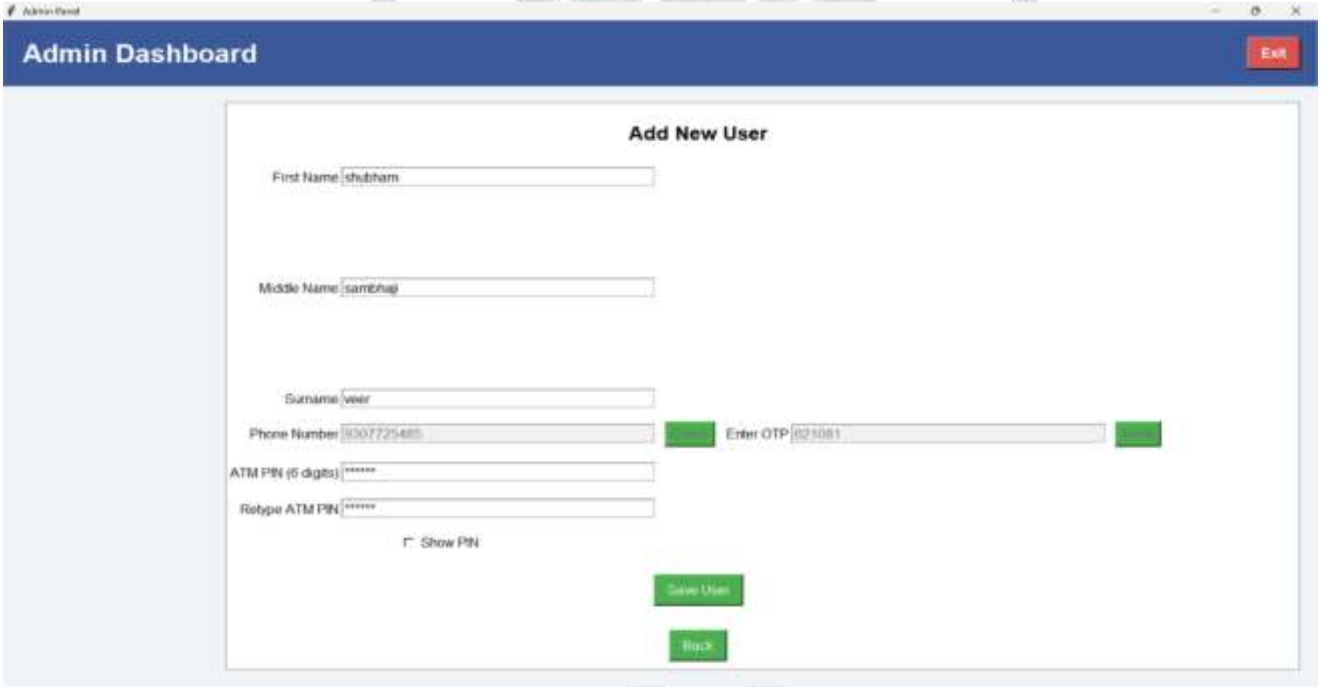
Surname

Phone Number

Enter OTP

ATM PIN (6 digits)

Retype ATM PIN



Add New User

First Name shubham

Middle Name sambhaji

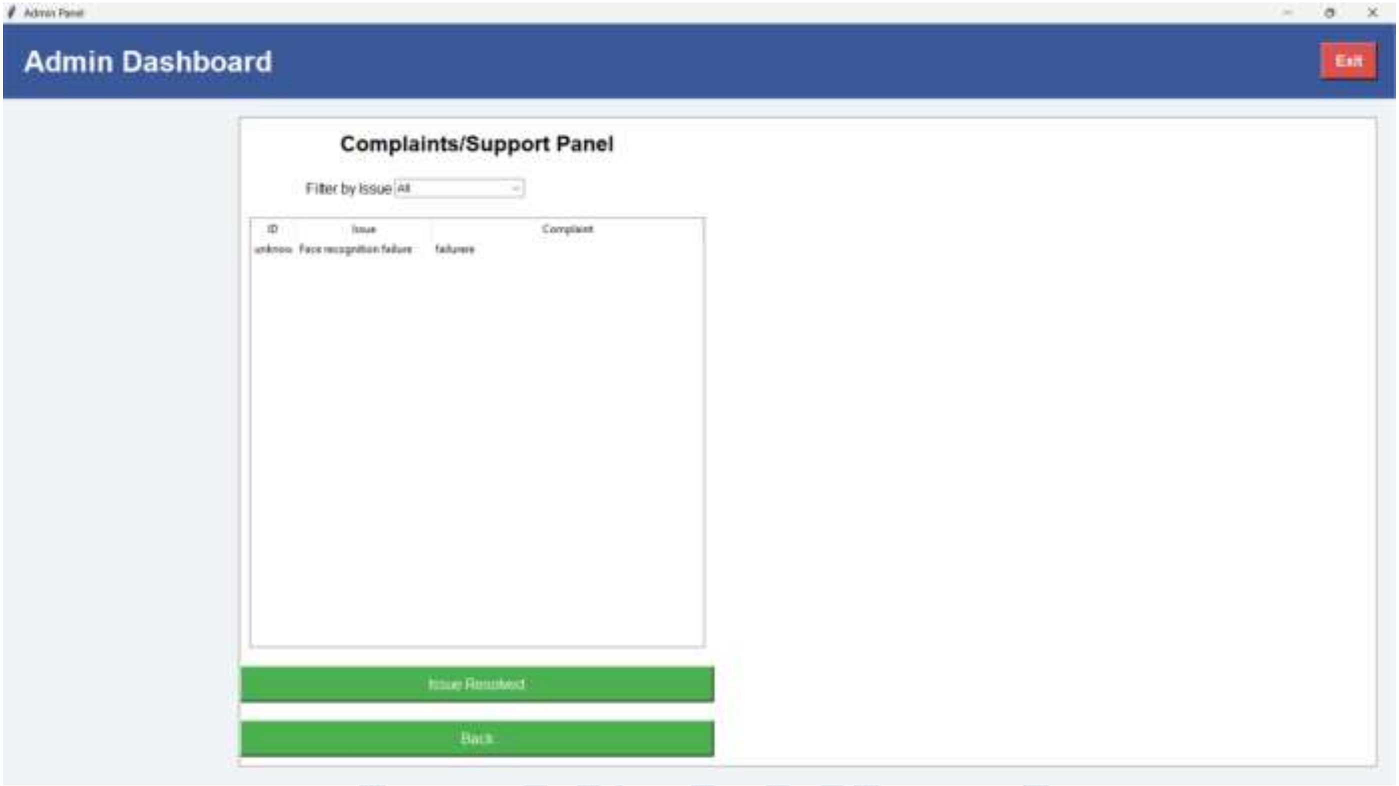
Surname veer

Phone Number 9307725485

Enter OTP 621081

ATM PIN (6 digits) ******

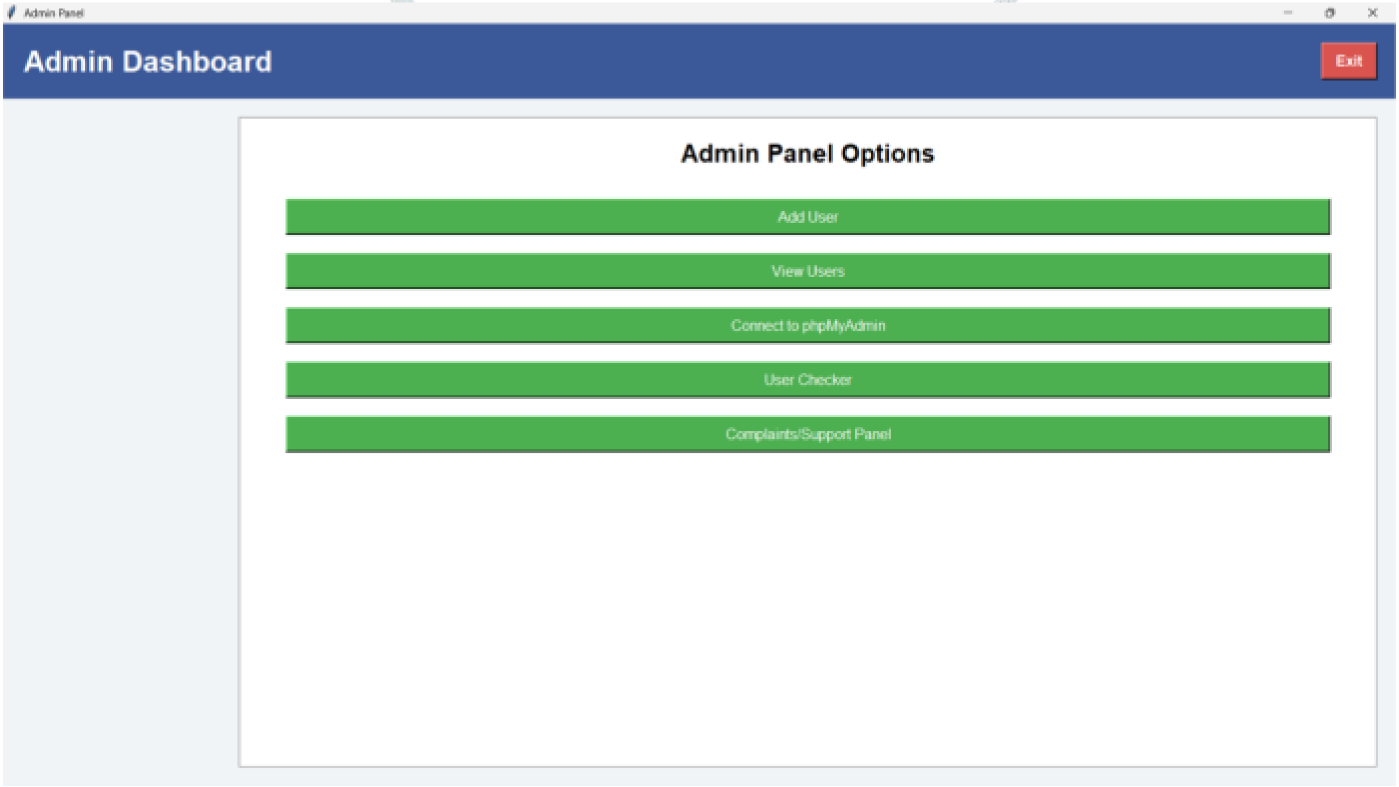
Retype ATM PIN ******

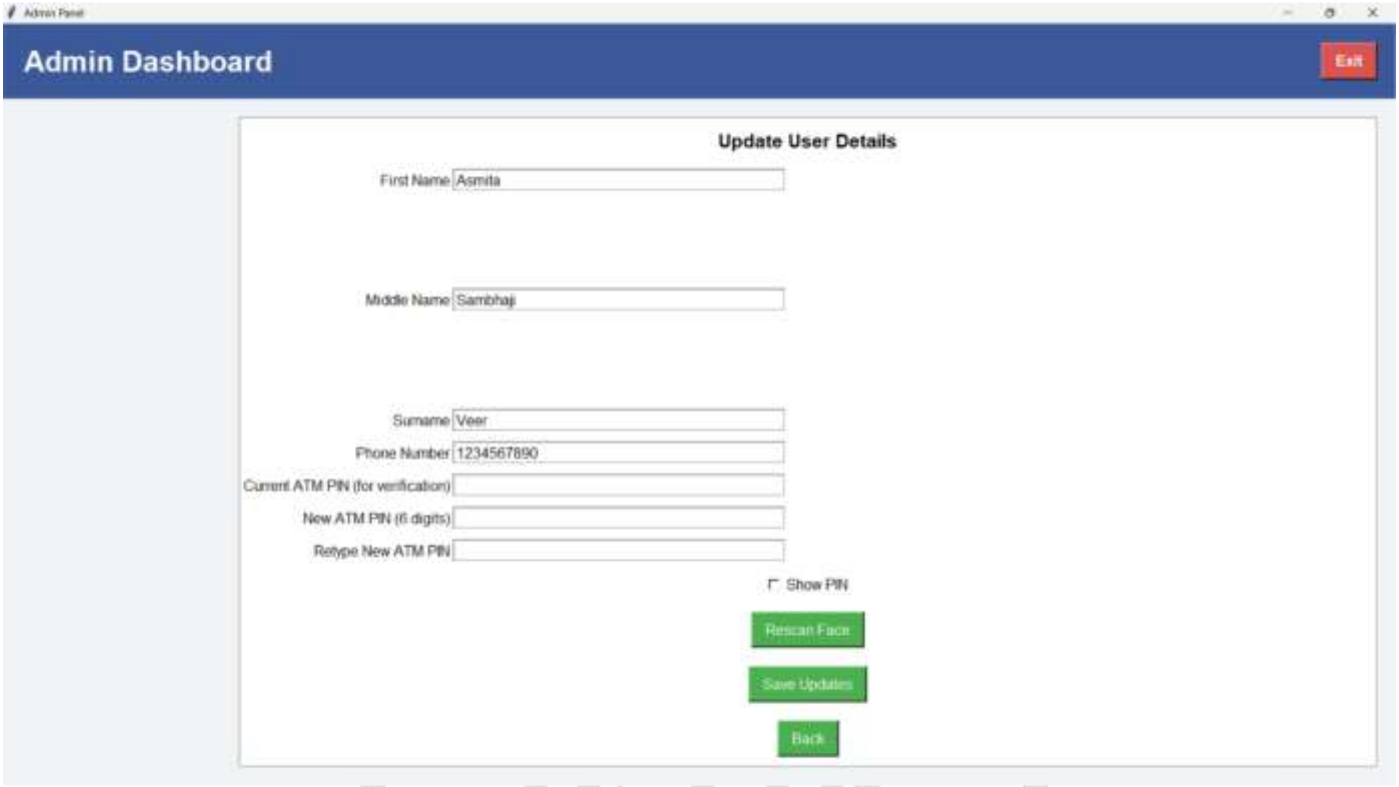


Complaints/Support Panel

Filter by Issue All

ID	Issue	Complaint
unknow	Face recognition failure	failurere

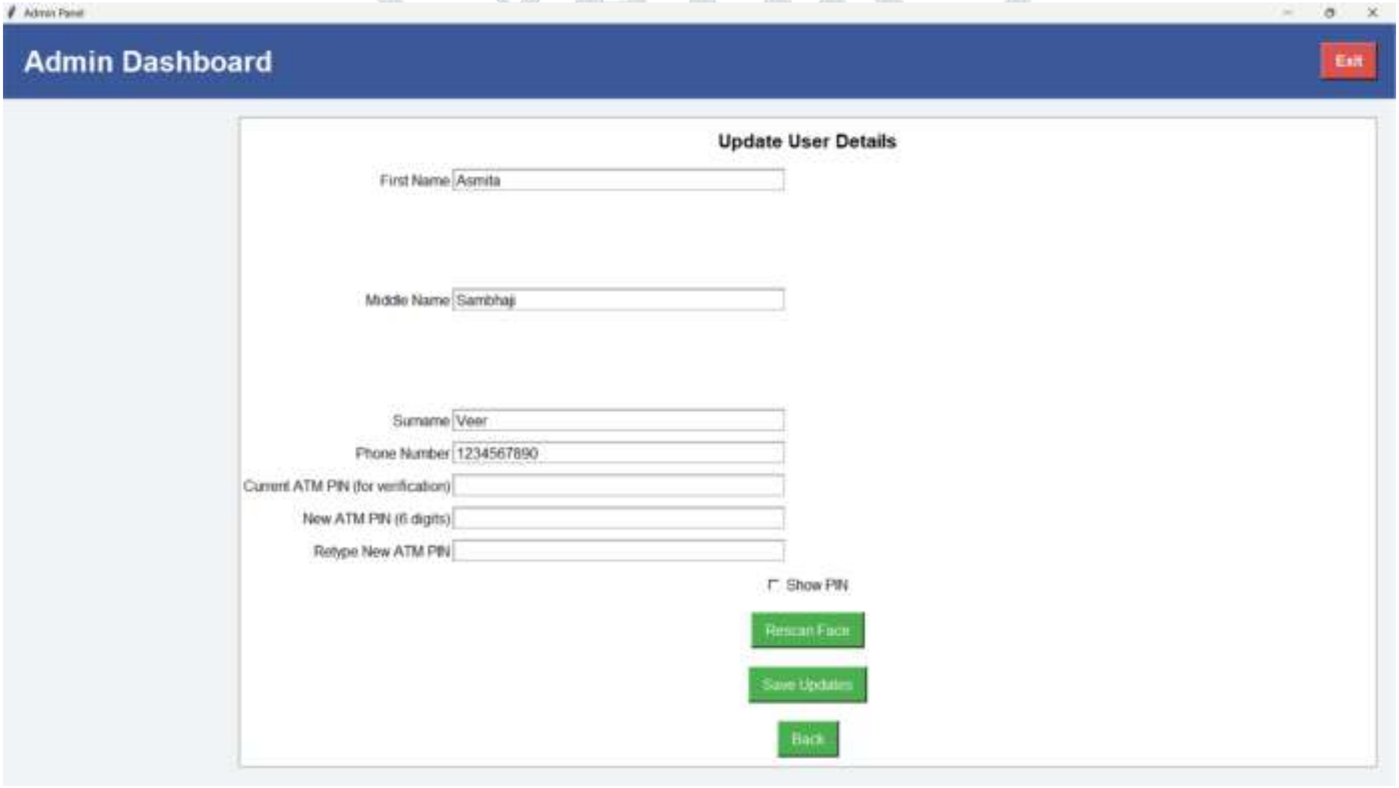




Update User Details

First Name	Asmita
Middle Name	Sambhaji
Surname	Veer
Phone Number	1234567890
Current ATM PIN (for verification)	
New ATM PIN (6 digits)	
Retype New ATM PIN	

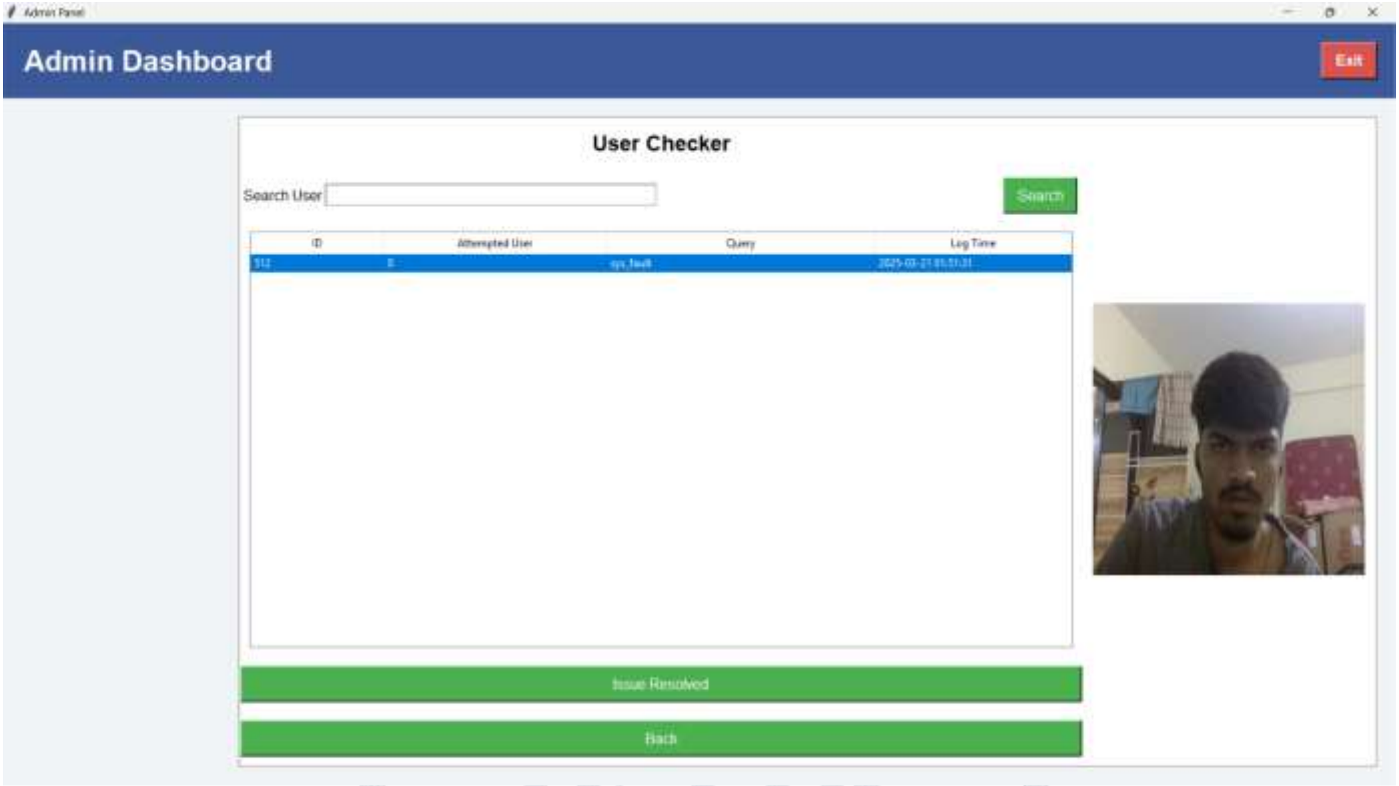
□ Show PIN



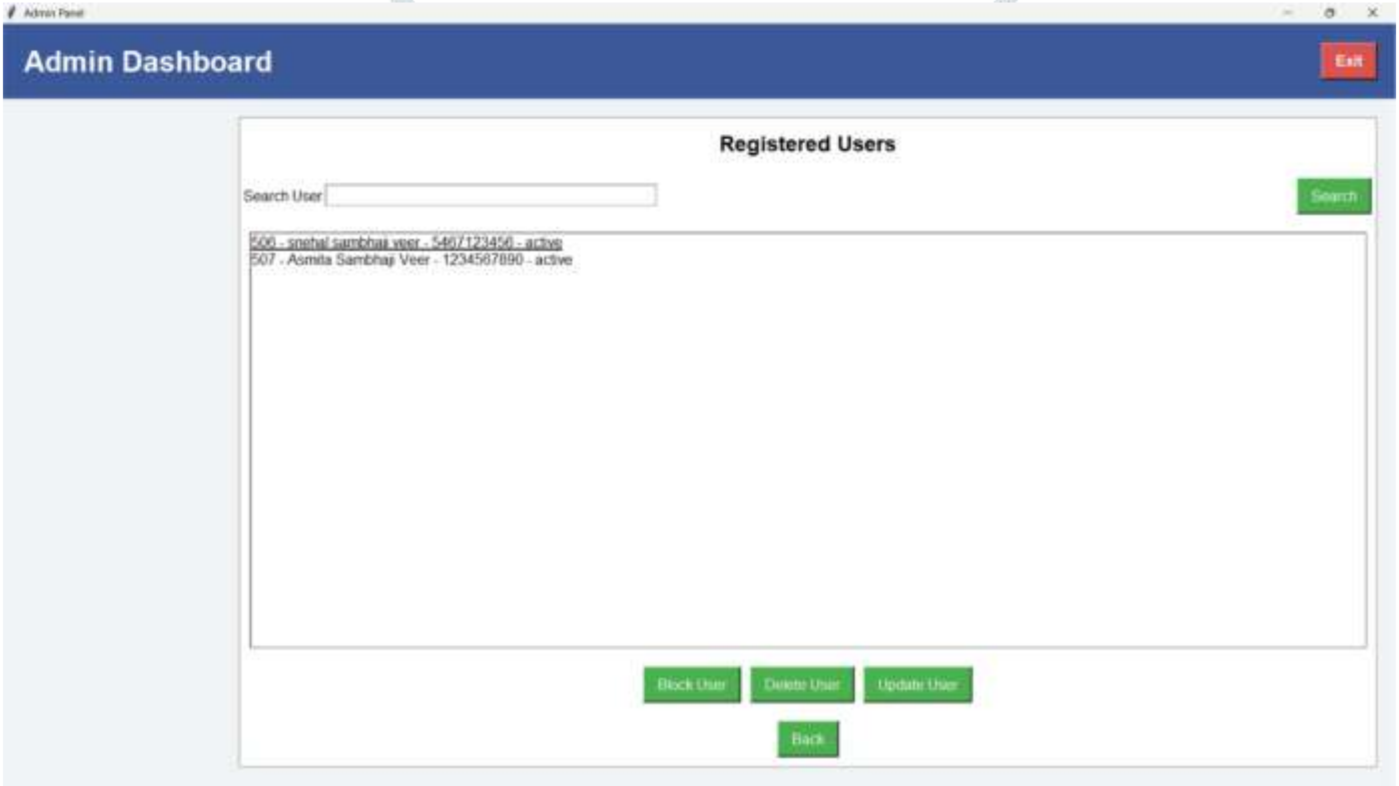
Update User Details

First Name	Asmita
Middle Name	Sambhaji
Surname	Veer
Phone Number	1234567890
Current ATM PIN (for verification)	
New ATM PIN (6 digits)	
Retype New ATM PIN	

□ Show PIN

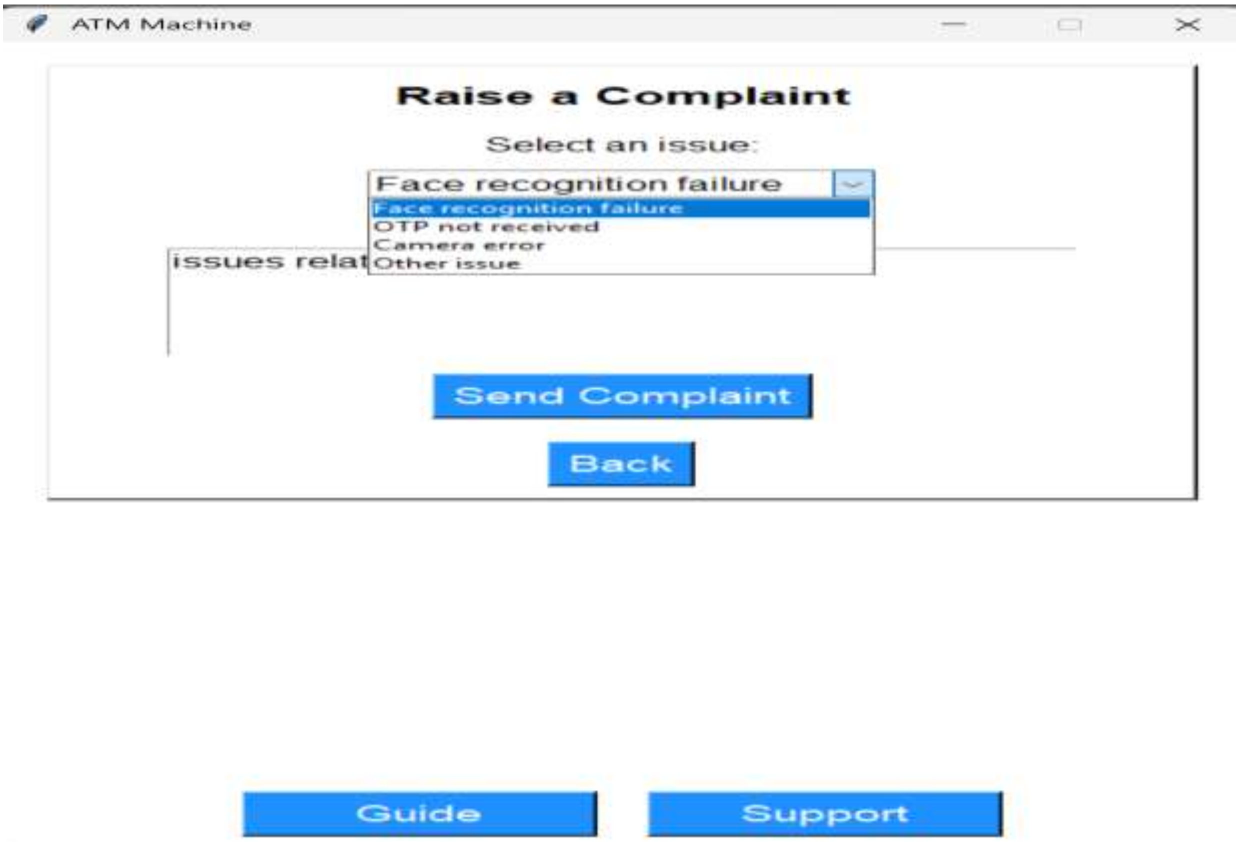


ID	Attempted User	Query	Log Time
512	8	sys_fault	2025-03-21 01:51:31



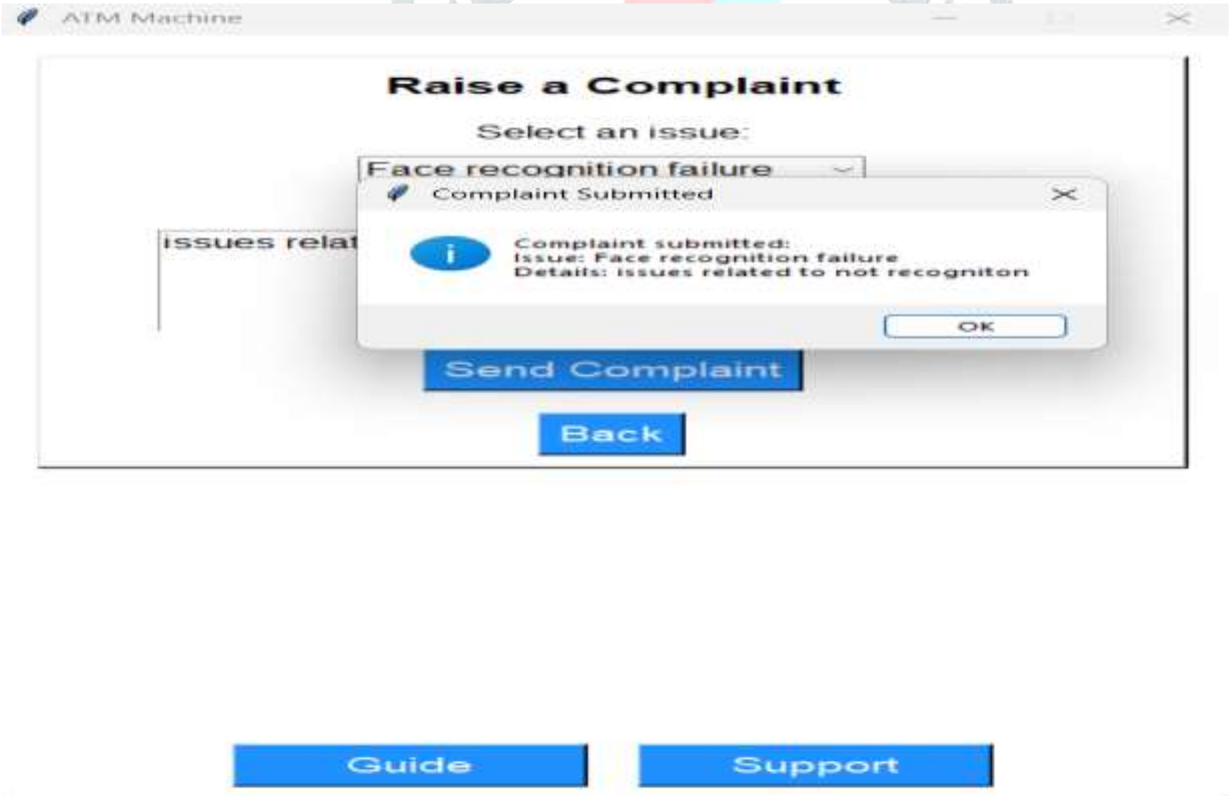
506 - snehal sambhaji veer - 5467123456 - active
507 - Asmita Sambhaji Veer - 1234567890 - active

User Dashbord :



Select an issue: Face recognition failure

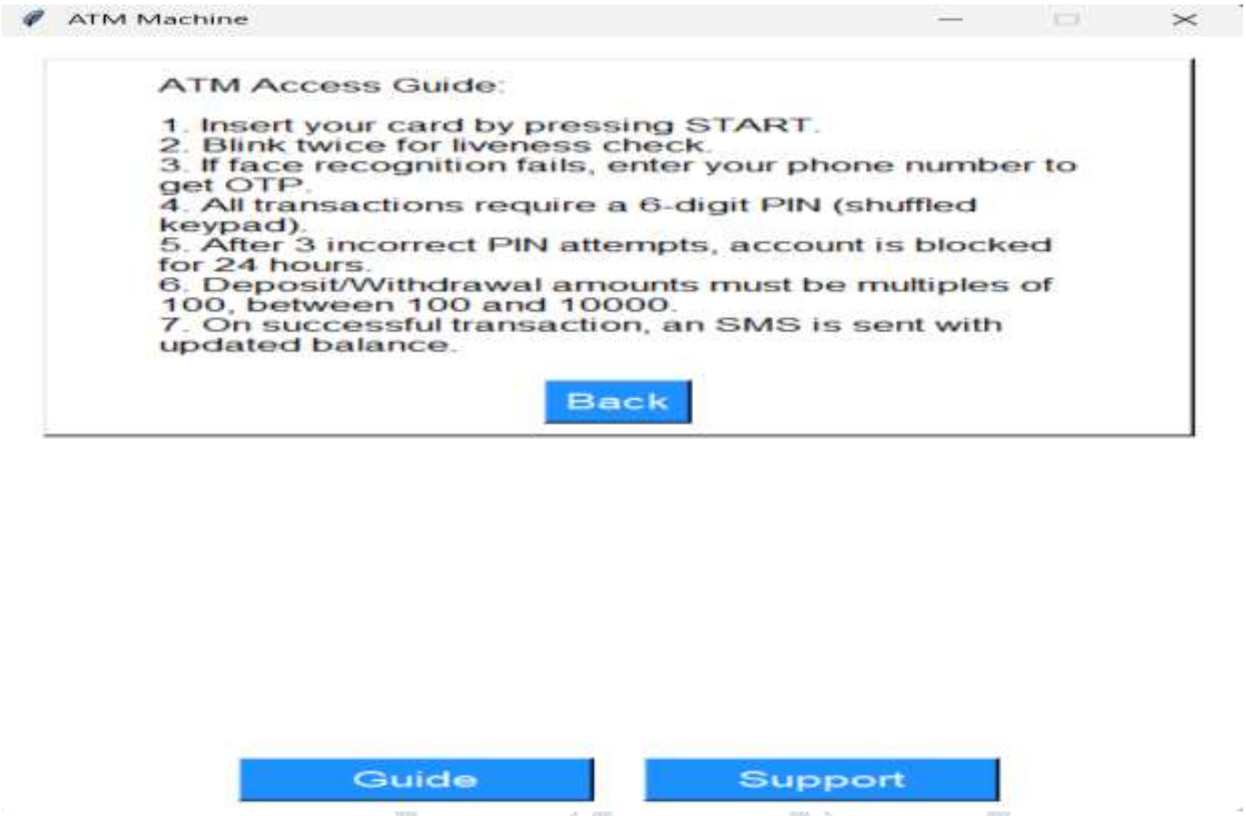
issues relat



Select an issue: Face recognition failure

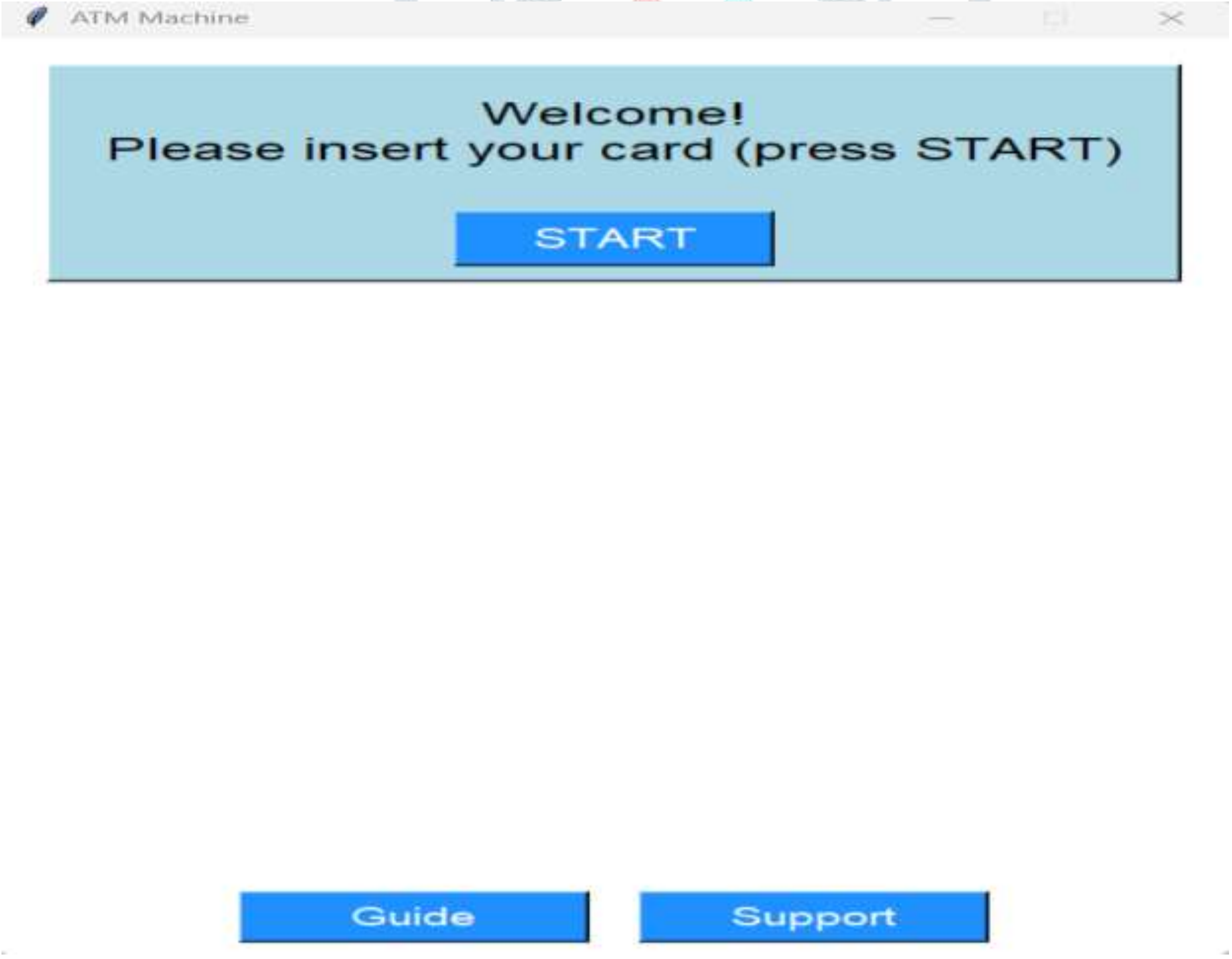
issues relat

Complaint submitted:
Issue: Face recognition failure
Details: issues related to not recogniton

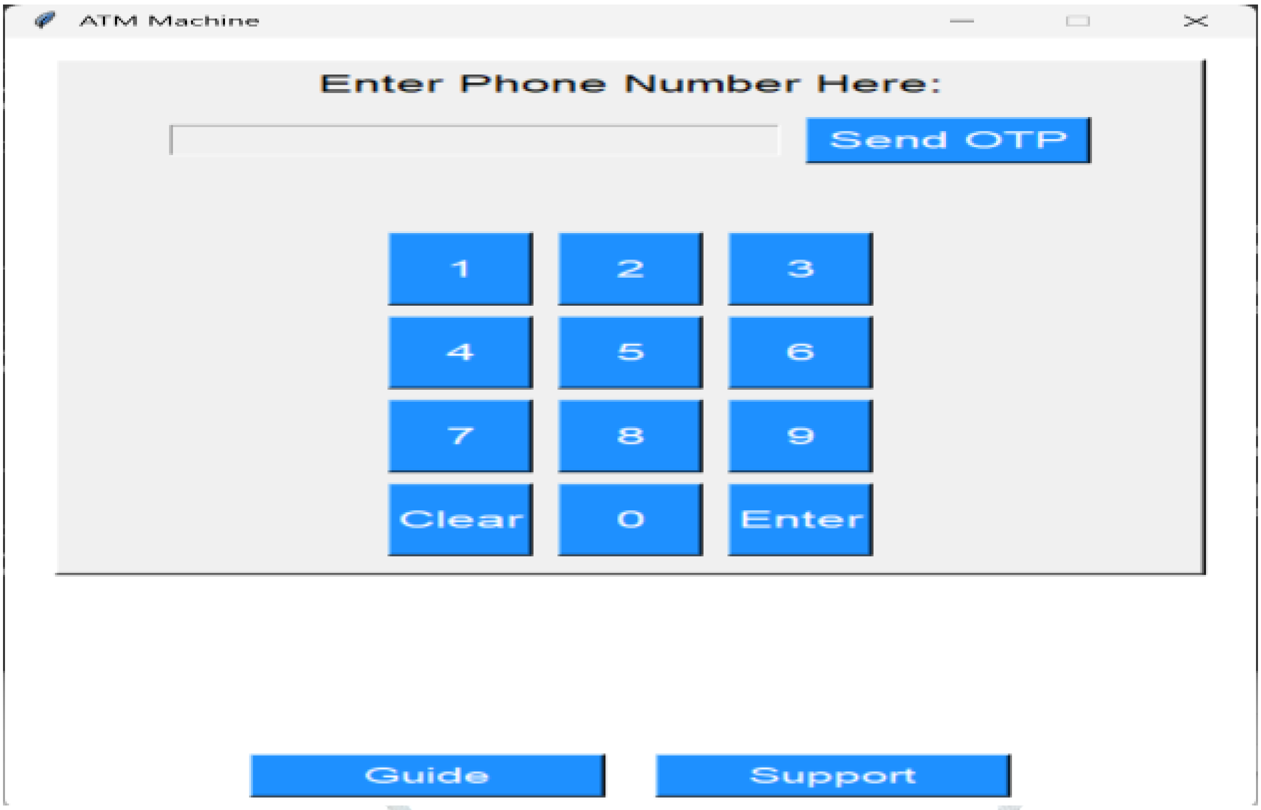


ATM Access Guide:

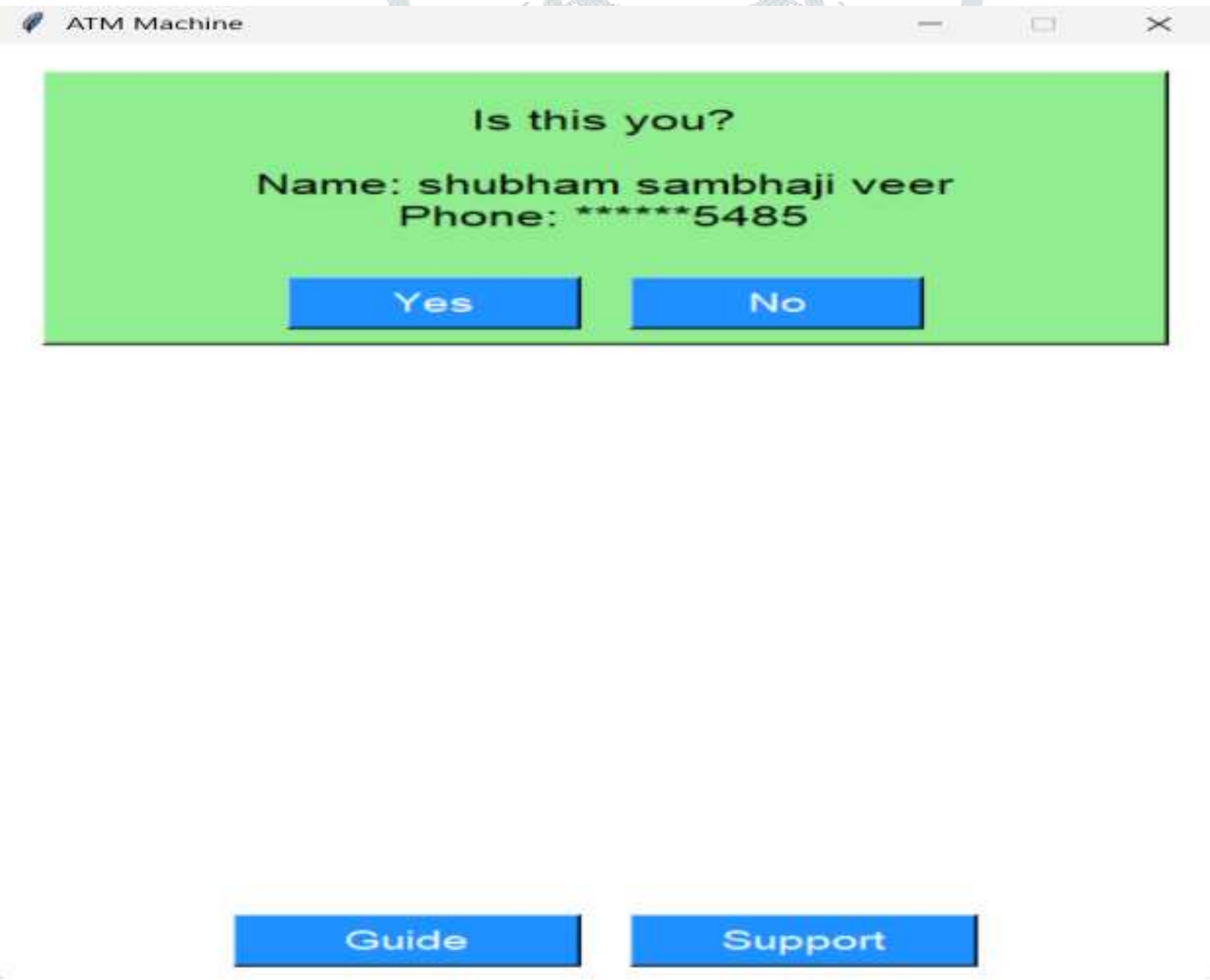
1. Insert your card by pressing START.
2. Blink twice for liveness check.
3. If face recognition fails, enter your phone number to get OTP.
4. All transactions require a 6-digit PIN (shuffled keypad).
5. After 3 incorrect PIN attempts, account is blocked for 24 hours.
6. Deposit/Withdrawal amounts must be multiples of 100, between 100 and 10000.
7. On successful transaction, an SMS is sent with updated balance.



Welcome!
Please insert your card (press START)

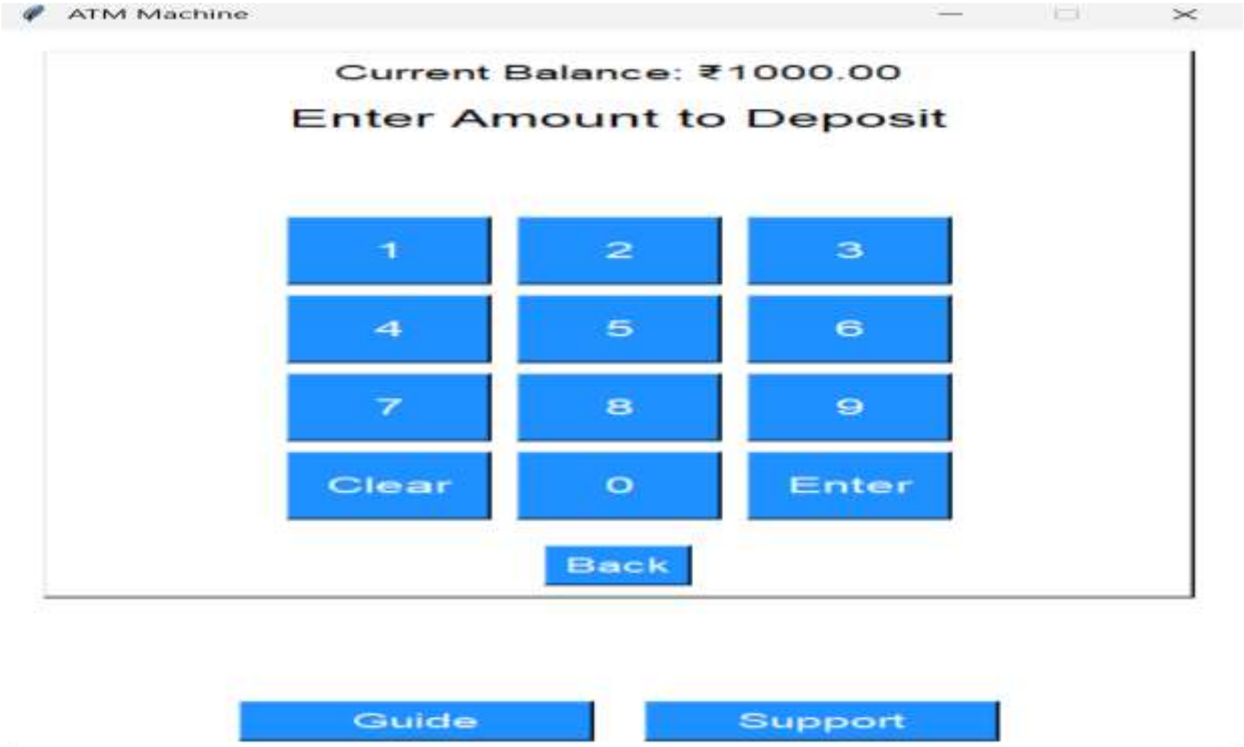


Enter Phone Number Here:



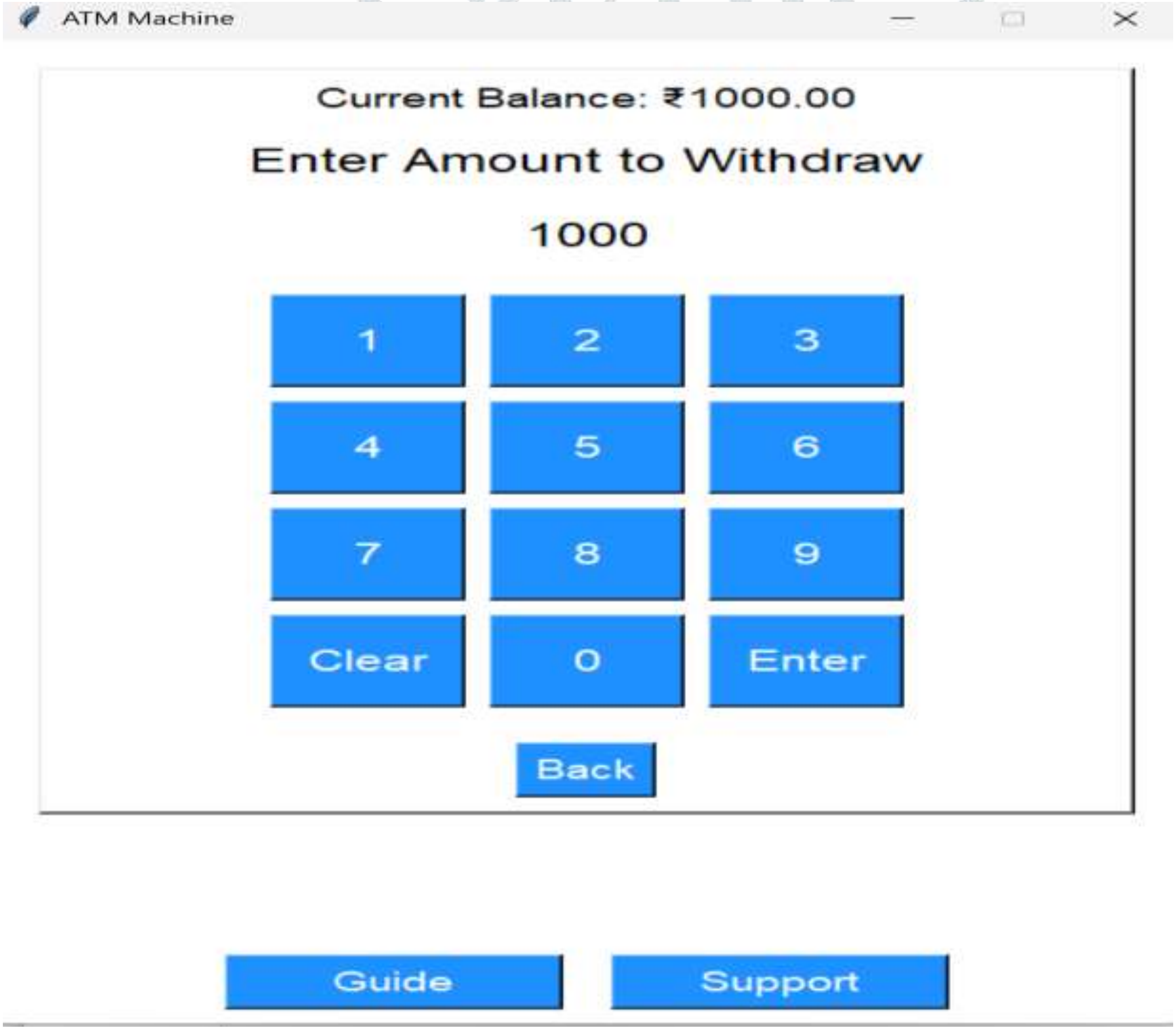
Is this you?

Name: shubham sambhaji veer
Phone: *******5485



Current Balance: ₹1000.00

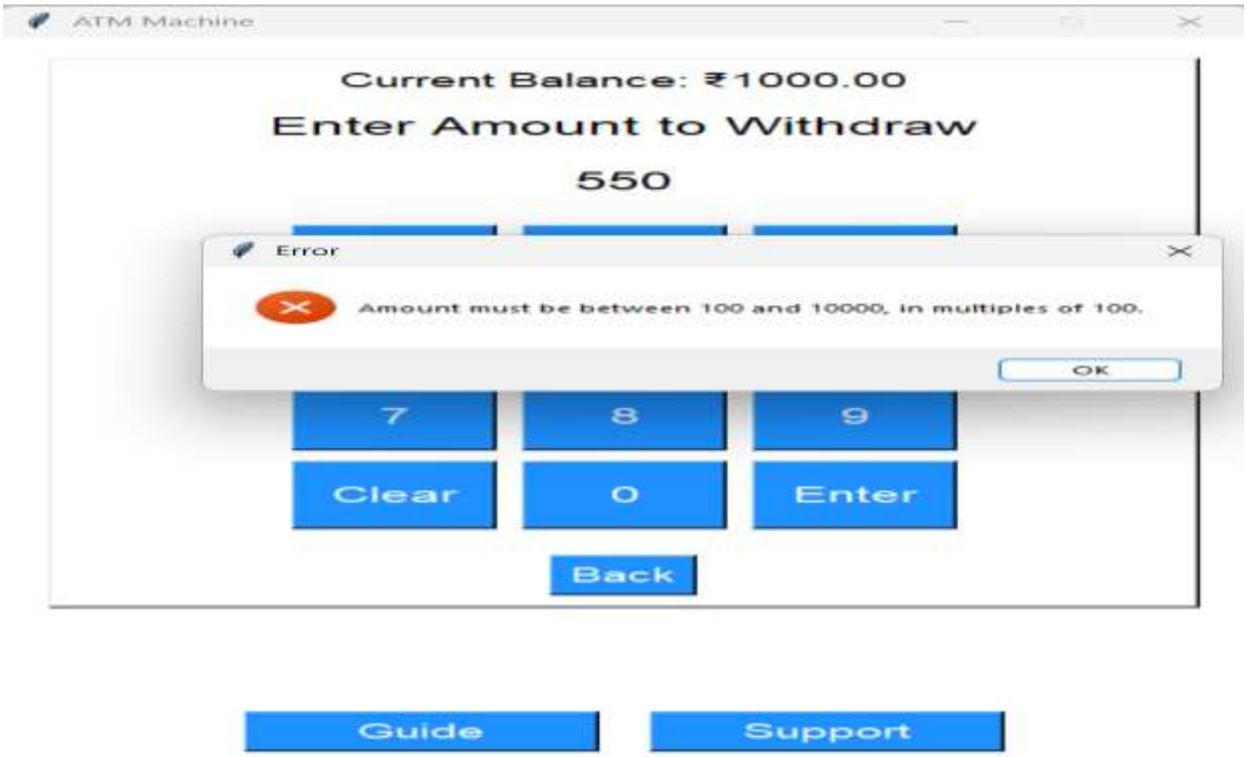
Enter Amount to Deposit



Current Balance: ₹1000.00

Enter Amount to Withdraw

1000



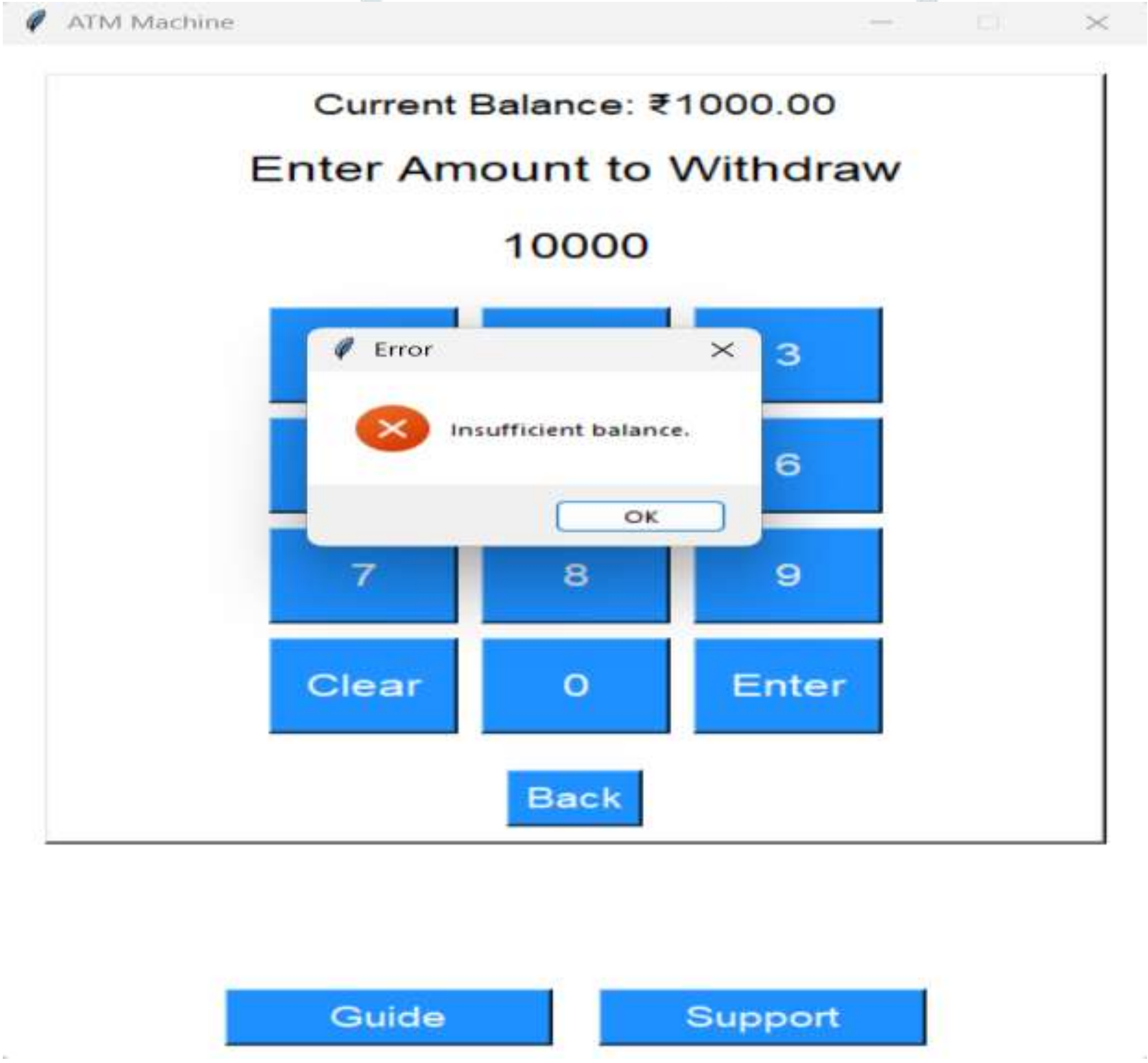
Current Balance: ₹1000.00

Enter Amount to Withdraw

550

Error

Amount must be between 100 and 10000, in multiples of 100.



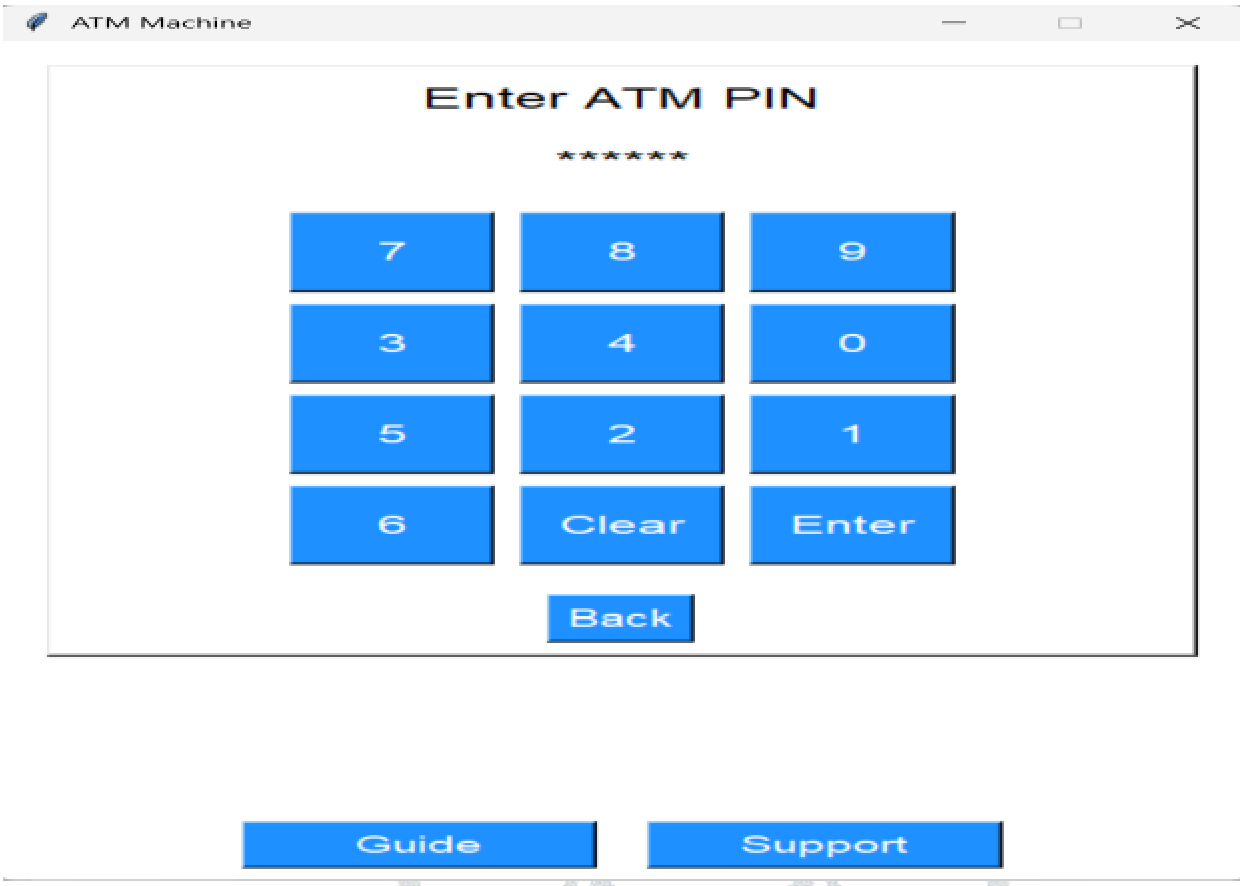
Current Balance: ₹1000.00

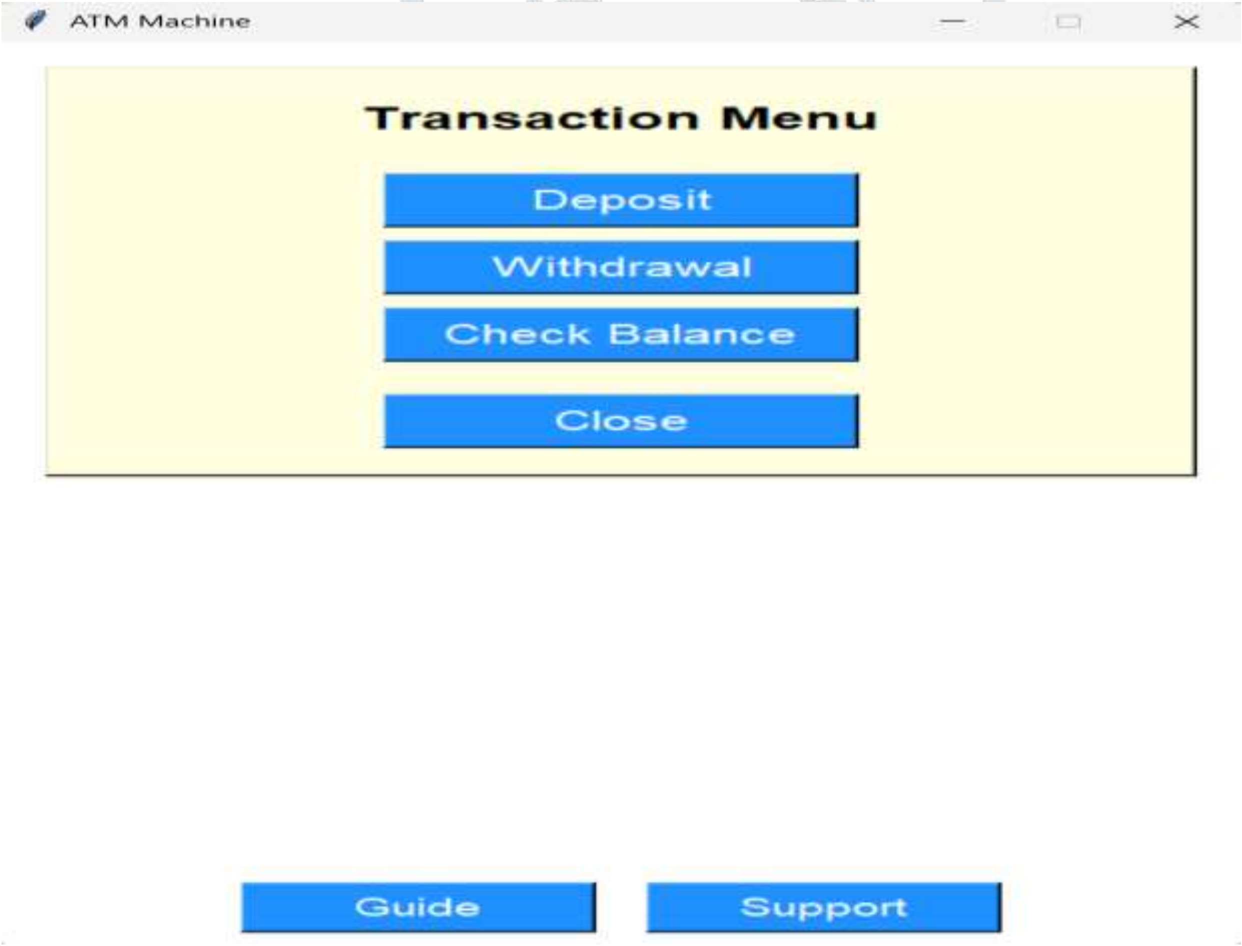
Enter Amount to Withdraw

10000

Error

Insufficient balance.


ATM Machine
Enter ATM PIN
******
7
8
9
3
4
0
5
2
1
6
Clear
Enter
Back
Guide
Support


ATM Machine
Transaction Menu
Deposit
Withdrawal
Check Balance
Close
Guide
Support

RESULT:- By implementing a multi-layered ATM security model that combines facial recognition, OTP (One-Time Password) verification, and a shuffled keypad, the overall safety and reliability of ATM transactions are greatly improved. The use of facial recognition ensures that only the authorized user can access their account, reducing the chances of unauthorized access. Adding OTP verification provides an extra layer of security, making it harder for fraudsters to bypass the system even if they somehow gain access to user information. The shuffled keypad prevents common security threats like shoulder surfing and PIN theft by making it impossible to predict the keypad layout during each transaction. Together, these measures create a secure and contactless user experience that reduces the risk of card skimming and other fraudulent activities. Additionally, storing sensitive data in encrypted databases ensures that user information remains protected from cyber threats and unauthorized access. The system is also designed to be scalable and compatible with existing banking infrastructure, allowing for easy integration and future upgrades as security needs evolve. Overall, this advanced security approach not only safeguards customers and their financial assets but also builds trust in ATM services by offering a seamless, secure, and user-friendly transaction process.

CONCLUSION : By implementing all these features in your **ATM Security Model**, the result will be a **highly secure, user-friendly, and future-proof system**. This will not only **protect customers' financial assets** but also **strengthen the reputation and reliability** of the banking institution that deploys it. The **multi-layered approach** significantly raises the **security standard** in ATM transactions, positioning the model as a cutting-edge solution in the financial technology space.

ACKNOWLEDGMENT

We extend our sincere gratitude to all those who contributed to the development of this **Advanced Security System For ATM Software with Facial Recognition and OTP with Shuffle Keypad Feature**. We deeply appreciate the support and guidance of our mentors and faculty members, whose insights and expertise played a crucial role in shaping this research..

REFERENCES :

[1] [He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. IEEE Conference on Computer Vision and Pattern Recognition (CVPR). [DOI:10.1109/CVPR.2016.90] – Reference for ResNet (Residual Networks) in facial recognition.

[2] George, A., & Marcel, S. (2019). DeepPixBiS: A Deep Learning-Based Approach for Face Presentation Attack Detection. IEEE International Conference on Biometrics (ICB). [DOI:10.1109/ICB45273.2019.8987303] – Reference for DeepPixBiS in face anti-spoofing.

[3] Yuqi Pan, Mingyan Jiang (2023). LRR-TTK DL: A Low-Rank Representation with Tensor Train Kernel for Deep Learning-Based Face Recognition. International Journal of Computer Vision. – Reference for LRR-TTK DL model.

[4] Hochreiter, S., & Schmidhuber, J. (1997). Long Short-Term Memory (LSTM). Neural Computation, 9(8), 1735-1780. [DOI:10.1162/neco.1997.9.8.1735] – Reference for LSTM in time-series analysis and fraud detection.

[5] Jain, A. K., Ross, A., & Nandakumar, K. (2011). Introduction to Biometrics. Springer. [ISBN: 978-0-387-77325-1] – Reference for biometric authentication in ATM security.

[6] Ramesh, S., & Kumar, K. (2021). ATM Security Enhancement Using Shuffling Keypad and Biometric Authentication. International Journal of Advanced Computer Science and Applications (IJACSA). – Reference for shuffled keypad security in ATM systems.

[7] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., & Bengio, Y. (2014). Generative Adversarial Networks (GANs). arXiv preprint arXiv:1406.2661. – Reference for deepfake detection and adversarial training in face recognition security.

[8] Rajesh, T., & Kumar, A. (2024). Virtual Shuffling Keypad System for Secure ATM Transactions. International Journal of Cybersecurity and Digital Forensics. – Reference for preventing PIN theft via shoulder surfing and keylogging.