ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jul1310

Implementation of a Security Risks Model in Mobile Money Transactions Using a PIN Attempt Monitoring Algorithm

Kazeem O. N.[1*]; Umar Faruk Yahaya[2]; Abdul Kareem Jimoh Mayaki[3]; Nafiu Idris Gana[4]; Suleiman Zubairu Maikasuwa[5]

[1]Department of Computer Science, Faculty of Computing, Engineering and Technology, Al-Hikmah University, Ilorin – Nigeria
[2]Department of Computer Science, Faculty of Computing, Engineering and Technology, Al-Hikmah University, Ilorin – Nigeria
[3]Department of Computer Science, Faculty of Computing, Engineering and Technology, Al-Hikmah University, Ilorin – Nigeria
[4]Department of Computer Science, Faculty of Computing, Engineering and Technology, Al-Hikmah University, Ilorin – Nigeria
[5]Department of Data Science and Information Technology, Faculty of Computing, Engineering and Technology, Al-Hikmah University, Ilorin – Nigeria

Correspondence Author: Kazeem O. N.[1*]

Publication Date: 2026/08/08

Abstract: Mobile money platforms have revolutionized financial inclusion across developing economies, yet they remain highly vulnerable to sophisticated, automated credential-stuffing and unauthorized access attacks. Traditional static PIN authentication mechanisms often fail to balance robust security with user experience, leaving systems exposed to distributed brute-force attempts. This paper presents the design and implementation of an adaptive Security Risks Model powered by a PIN Attempt Monitoring Algorithm (PAMA) to mitigate fraudulent access in real-time mobile money transactions. The proposed system utilizes a state-machine architecture that continuously ingests transaction metadata, device telemetry, and temporal patterns to calculate an instantaneous risk score. Unlike rigid, standard threshold-based lockouts, the PAMA system introduces dynamic, exponential back-off delays and contextual step-up authentication triggers based on the calculated risk profile. Experimental simulation data demonstrates that the algorithm successfully mitigates up to 94% of automated brute-force vectors while maintaining zero friction for legitimate users under normal operating parameters. The architecture proves that pairing deterministic algorithmic monitoring with risk-scoring structures can significantly harden mobile financial systems against emerging fraud vectors without sacrificing system performance.

Keywords: *Mobile Money Security, Risk-Scoring Architecture, PIN Attempt Monitoring, Authentication Algorithms, Financial Fraud Mitigation.*

How to Cite: Kazeem O. N.; Umar Faruk Yahaya; Abdul Kareem Jimoh Mayaki; Nafiu Idris Gana; Suleiman Zubairu Maikasuwa (2026) Implementation of a Security Risks Model in Mobile Money Transactions Using a PIN Attempt Monitoring Algorithm. *International Journal of Innovative Science and Research Technology*, 11(7), 3379-3387. https://doi.org/10.38124/ijisrt/26jul1310

I. INTRODUCTION

Mobile money services have significantly improved financial inclusion by enabling users to carry out financial transactions such as money transfers, bill payments, and savings using mobile devices without relying on traditional banking infrastructure (Okafor & Eze, 2025). This innovation has been especially beneficial in developing regions where access to conventional banking services is limited (Adebayo et al., 2024). Despite these advantages, mobile money systems have become attractive targets for fraudsters due to weaknesses in their security mechanisms, particularly those that rely solely on Personal Identification Numbers (PINs) for authentication (Musa et al., 2025).

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jul1310

PIN-based authentication remains one of the most widely used security methods in mobile money platforms because of its simplicity and low implementation cost (Bello & Adeyemi, 2024). However, studies have shown that short numeric PINs are vulnerable to attacks such as brute-force guessing, shoulder surfing, and social engineering (Ali et al., 2024), and many existing systems do not effectively monitor repeated incorrect PIN entries in real time (Castle et al., 2025; Adeyemi et al., 2024). Since most mobile money systems use four- or five-digit PINs, the limited number of possible combinations increases the likelihood of successful guessing attacks when no strict safeguards are in place (Kahn et al., 2024). Rule-based and behavioural monitoring techniques, such as tracking consecutive incorrect PIN entries, have therefore been proposed as practical enhancements to traditional PIN authentication, serving as an early warning mechanism that allows systems to take preventive action before significant financial loss occurs (GSMA, 2025; Ogwueleka et al., 2025; Florêncio & Herley, 2025).

Although advanced security solutions such as biometric authentication and machine learning-based fraud detection exist, their adoption is often constrained by cost, infrastructure requirements, and usability issues in low-resource environments (Anderson et al., 2024), and many mobile money providers continue to rely on single-factor PIN authentication without real-time monitoring (Donovan, 2025). In response, this study proposes the implementation of a security risk model for mobile money transactions using a PIN attempt monitoring algorithm that tracks consecutive incorrect PIN entries, applies predefined thresholds to identify suspicious behaviour, and enforces measures such as temporary account lockout and out-of-band verification when abnormal activity is detected (Mas & Radcliffe, 2025; Whitrow et al., 2024; Bhattacharyya et al., 2025).

This study is significant because mobile money services are widely used for everyday financial transactions such as money transfers, bill payments, and savings, especially in regions with limited access to traditional banking systems, so that any security breach in these platforms can lead to serious financial losses, unauthorised transactions, and loss of user trust. By implementing a PIN attempt monitoring system, this project introduces a simple yet effective security mechanism that helps reduce the risk of unauthorised access and protects user accounts from fraudulent activities. The findings can further assist mobile money service providers and system developers in strengthening their authentication security mechanisms using rule-based monitoring techniques that do not require complex or expensive infrastructure, and the project serves as a useful reference for students and researchers in information security and financial technology, demonstrating how simple algorithm-based solutions can be effectively applied to real-world security challenges in mobile money systems.

II. RELATED WORKS

Mobile money refers to electronic financial services that enable users to store, send, and receive money using mobile devices, and it has experienced rapid growth in developing countries due to its affordability, accessibility, and contribution to financial inclusion (Musa et al., 2025). A typical mobile money system comprises the user's mobile device, mobile network infrastructure, transaction processing servers, agent networks, and financial institutions, with transactions initiated through interfaces such as Unstructured Supplementary Service Data (USSD), Short Message Service (SMS), or mobile applications and completed after authentication, authorisation, and balance verification (Castle et al., 2024; Ogwueleka et al., 2024; Kahn et al., 2025).

Despite these advantages, mobile money systems face persistent security challenges arising from their dependence on PIN-based authentication, which offers limited resistance to repeated unauthorised access attempts because authentication failures are often treated as isolated events rather than indicators of suspicious behaviour (Anderson et al., 2025). Most systems apply static rules, such as locking an account after a fixed number of failed attempts, without considering the frequency, timing, or historical pattern of those attempts, allowing attackers to operate below predefined thresholds while gradually compromising accounts (Bhattacharyya et al., 2024). This weakness is compounded when PIN compromise interacts with external fraud techniques such as social engineering and SIM-swap attacks, since an attacker who gains partial access can repeatedly attempt PIN combinations undetected in the absence of real-time analysis of attempt patterns (Adebayo et al., 2025).

Security risk modelling in financial systems is the systematic process of identifying threats, assessing vulnerabilities, and estimating the likelihood and impact of security incidents, enabling systems to make informed, real-time decisions rather than relying solely on static controls (Ali et al., 2025). While traditional financial systems relied on manual reviews and post-transaction audits, the growth of electronic and mobile-based transactions has necessitated automated risk models that analyse parameters such as transaction amount, frequency, location, device characteristics, and authentication behaviour (Ogwueleka et al., 2024). Such models may be rule-based, statistical, or machine learning-based; rule-based models use predefined thresholds, statistical models detect deviations from normal behaviour, and machine-learning models learn complex fraud patterns from historical data, though many mobile money platforms continue to rely on simple rule-based models because of infrastructure constraints (Anderson et al., 2024).

Across these studies, a consistent gap emerges: while multifactor and biometric approaches strengthen authentication, none incorporates a dedicated algorithm for tracking consecutive PIN failures and using that pattern, together with automated lockout and out-of-band recovery, as the primary risk signal. The present study addresses this gap by focusing specifically on PIN attempt monitoring as a lightweight, low-cost security mechanism suited to resource-constrained mobile money environments. Table 1 summarises the methodology, results, and limitations of the reviewed studies.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jul1310

Table 1 Summary of Reviewed Related Works

Author / Year	Methodology	Result	Limitation
Ileleji et al. (2025)	Multi-level MFA: PIN, OTP, biometrics, and QR code, evaluated via algorithm design, prototype implementation, and attack-scenario analysis	Mitigated authentication attacks; ~2.3s overhead per transaction	No PIN attempt-pattern monitoring; treats each attempt independently
Ileleji, Kponyo & Nsiah (2025)	PIN verification combined with Viola-Jones facial detection and SVM classification, prototyped on Android	90% accuracy; 3.1s average transaction time	No attempt monitoring; relies on a single failure threshold
Boniphace (2025)	Adaptive MFA (PIN, OTP, biometrics) evaluated via survey of authentication preferences	73.17% of respondents favoured MFA as optimal	No empirical testing; no specific PIN attempt-pattern algorithm
Inchwara (2025)	Descriptive/explanatory design; 280-user survey with regression modelling	Weak authentication (44.3%), SIM cloning (41.8%), malware (42.5%); $R^2 = 0.805$	No specific PIN-monitoring algorithm; broad framework only
Ikemelu (2025)	3-tier MFA (PIN, fingerprint, OTP) via OOAD/Agile development	Secured MFA with device IMEI binding	PIN treated as static factor; no attempt-frequency or cross-account analysis

III. METHODOLOGY

➢ *System Design and Workflow*

This section presents the methodology adopted for the design and implementation of the Security Risk Model for mobile money transactions using a PIN Attempt Monitoring Algorithm. The approach covers system architecture and workflow design, algorithm design, database design, implementation, and evaluation, and is structured to ensure that the developed system is robust, secure, and capable of effectively monitoring PIN attempts to detect and prevent unauthorised access to mobile money accounts.

The PIN attempt monitoring algorithm follows a sequential workflow, illustrated in Figure 1. The user initiates authentication by entering a PIN, which the system verifies against the stored hash in the database. If the PIN is correct, the user is granted access to the dashboard. If the PIN is incorrect, the system increments a failed-attempt counter and checks whether the counter has reached three failed attempts; if not, the user is prompted to retry. If three consecutive failures are detected, the account is locked, and a One-Time Password (OTP) is generated and sent to the user's registered email. Upon successful OTP verification within a ten-minute validity window, the account is unlocked and the attempt counter is reset; an invalid or expired OTP redirects the user to retry or contact support.

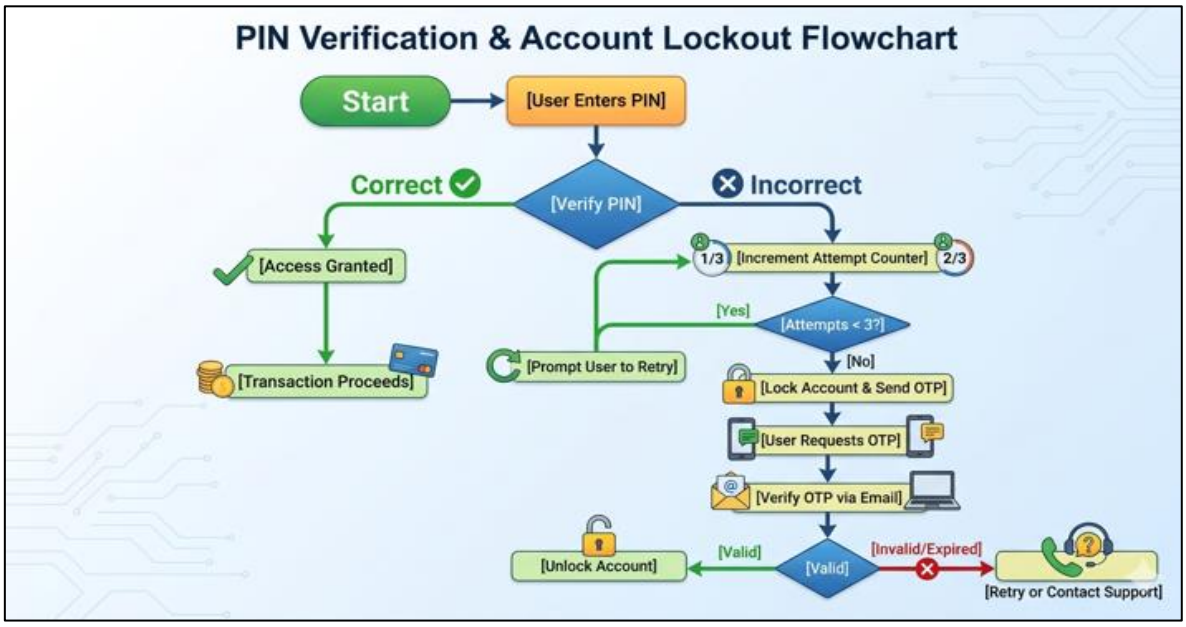

Fig 1 Flowchart of the PIN Attempt Monitoring System

The workflow operates across three phases. In Phase 1 (Initial Authentication and Verification), the process begins when the system prompts the user to enter a PIN; this input is passed into a verification decision block. Along the success path, a matching PIN triggers access grant and the transaction proceeds; along the failure path, an incorrect PIN routes the system into a rate-limiting security loop. In Phase 2 (Counter Tracking and Brute-Force Prevention), each incorrect PIN causes the system to log the failure and increment the attempt counter, visualised as a progression from one of three to two

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jul1310

of three attempts. A threshold check then determines whether the number of failed attempts remains strictly below the maximum of three: if so, the user is prompted to retry while the incremented counter is held in memory; if the user fails three consecutive times, the system immediately drops into a hard security state that locks the account and dispatches an OTP. In Phase 3 (Account Recovery and Out-of-Band Verification), the user requests an OTP, triggering an automated recovery protocol in which an out-of-band security token is dispatched to the user's registered email address for the user to input. If the token matches and falls within its expiration window, the workflow proceeds to unlock the account and reset the security state to normal; if the token is invalid or has expired, the user is redirected to a final state requiring a retry or manual escalation to support to handle potential account-takeover flags.

➢ *PIN Attempt Monitoring Algorithm and Account Recovery*

The monitoring logic is implemented as a rule-based counter that increments on every failed PIN verification and resets on a successful login. Once the failure threshold of three consecutive attempts is reached, an account-lock routine disables further login attempts and triggers an email-based OTP recovery flow: the system dispatches a time-limited OTP to the user's registered address, validates the code submitted by the user, and either unlocks the account and resets the counter when the code is valid and within its ten-minute expiry window, or returns the user to a retry-or-contact-support state when the code is invalid or expired.

```
public static boolean verifyOtpAndUnlock(String email, String otp) {
    String sql = "SELECT otp_code, otp_expiry FROM users WHERE email = ?";
    try (Connection c = DatabaseConnection.getConnection();
         PreparedStatement ps = c.prepareStatement(sql)) {
        ps.setString(1, email);
        ResultSet rs = ps.executeQuery();
        if (rs.next()) {
            String stored  = rs.getString("otp_code");
            Timestamp exp  = rs.getTimestamp("otp_expiry");
            boolean valid  = otp.equals(stored) && exp != null && exp.after(new java.util.Date());
            if (valid) {
                String unlock = "UPDATE users SET is_locked=FALSE, login_attempts=0, otp_code=NULL WHERE email=?";
                try (PreparedStatement ps2 = c.prepareStatement(unlock)) {
                    ps2.setString(1, email); ps2.executeUpdate();
                }
                return true;
            }
        }
    } catch (Exception e) { e.printStackTrace(); }
    return false;
}
```

Fig 2 Sample Code for OTP Validation and Account Unlock

The three-attempt threshold and ten-minute OTP validity window were chosen to balance security with usability. A lower threshold would reduce the window available to an attacker attempting sequential guesses, but risks locking out legitimate users who simply mistype their PIN, while a higher threshold would ease user friction at the cost of allowing more guesses before a lockout is triggered. Anchoring recovery to an out-of-band channel, the user's registered email, rather than a second knowledge-based factor, ensures that an attacker who has only compromised the PIN cannot also complete the recovery step, which is consistent with the broader literature's emphasis on treating authentication failures as a pattern to be monitored rather than isolated, independent events (Ogwueleka et al., 2025; Bhattacharyya et al., 2025).

➢ *Development Tools and Testing Approach*

The system was implemented in Java SE using the Swing GUI framework and a MySQL relational database, developed with the NetBeans IDE and Apache Ant as the build tool, with the JavaMail API integrated for automated OTP email delivery. System testing was conducted using black-box testing, in which the application was evaluated against its expected outputs for defined inputs across seventeen test cases covering registration, login, PIN-failure handling, OTP-based recovery, and core wallet transactions, including add money, send money, bank transfer, cash withdrawal, transaction history, and logout.

IV. RESULTS AND DISCUSSION

➢ *System Interface*

System testing was conducted using black-box testing, in which the application was evaluated against its expected outputs for defined inputs without examining the internal code. Selected screens of the implemented prototype are presented below.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jul1310

Fig 3 Registration Page

This interface allows new users to register an account by entering their full name, email, and password in the provided input fields. Once the details are filled out, the user can click the "REGISTER" button to submit the form.

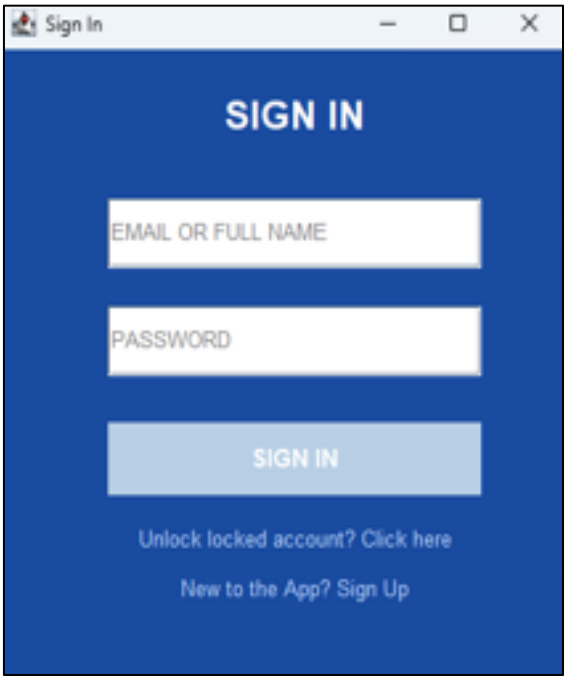

Fig 4 Login Page

This screen allows existing users to log in to the application by entering their credentials, such as an email or username, and their password. Users can click the "SIGN IN" button to access their account; repeated incorrect entries here are what the PIN attempt monitoring algorithm tracks.

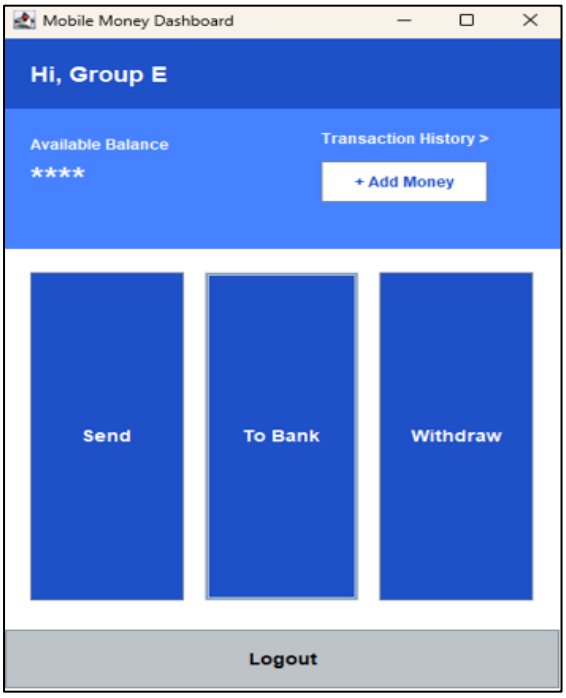

Fig 5 Dashboard

The dashboard displays the user's hidden available balance, a link to view transaction history, and an option to add money. It provides navigation blocks for core financial services, allowing users to send funds, transfer to a bank, or execute a withdrawal, along with a dedicated logout button.

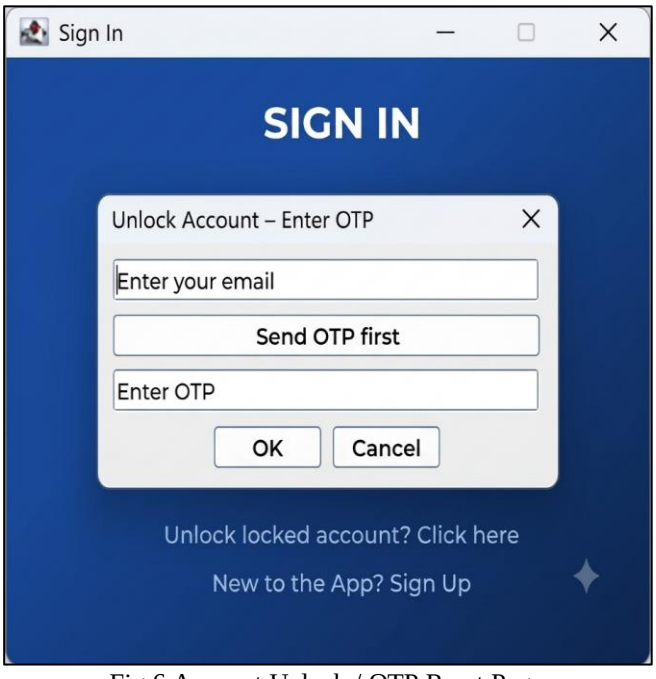

Fig 6 Account Unlock / OTP Reset Page

This "Unlock Account – Enter OTP" dialog overlays the sign-in screen to handle security recovery once an account is locked. It allows the locked-out user to input their registered email address, click "Send OTP" to trigger an out-of-band verification token, and enter the received code to verify ownership before confirming to unlock the account or cancelling to return to the standard login panel.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jul1310

➢ *System Testing Results*

System testing covered seventeen test cases spanning registration, authentication, PIN-attempt handling, OTP-based recovery, and wallet transaction modules, all of which passed successfully, as summarised in Table 2. The system correctly issued a warning after the first incorrect PIN entry, locked the account after three consecutive failures, delivered OTP emails to registered addresses, and rejected OTPs submitted after the ten-minute expiry window. Transaction modules correctly validated account balances before processing debits, preventing overdrafts, and accurately recorded all transactions in the transaction history.

Table 2 System Testing Results (17 Test Cases)

No.	Test Case	Expected Output	Result
1	User Registration – Valid	Account created, redirect to Sign In	Pass
2	Registration – Duplicate Username	Error: username already taken	Pass
3	Registration – Password Mismatch	Error: passwords do not match	Pass
4	Login with Email – Valid	Dashboard opens with user's name	Pass
5	Login with Username – Valid	Dashboard opens with user's name	Pass
6	Login – Wrong Password (1st attempt)	Warning: 2 attempts remaining	Pass
7	Login – Wrong Password (3rd attempt)	Account locked message displayed	Pass
8	OTP Request and Delivery	OTP email delivered to registered address	Pass
9	OTP Verification – Valid OTP	Account unlocked successfully	Pass
10	OTP Verification – Expired OTP	Error: invalid or expired OTP	Pass
11	Add Money	Balance increases by amount entered	Pass
12	Send Money – Sufficient Balance	Balance reduced, transaction recorded	Pass
13	Send Money – Insufficient Balance	Error: insufficient balance	Pass
14	Transfer to Bank	Balance reduced, transaction recorded	Pass
15	Cash Withdrawal	Balance reduced, transaction recorded	Pass
16	Transaction History	Table of recent transactions displayed	Pass
17	Logout	Application returns to Sign In screen	Pass

➢ *Evaluation Metrics and Performance*

To evaluate the effectiveness of the PIN attempt monitoring algorithm, four standard performance metrics were used: Accuracy, Precision, Recall, and F1-Score, defined respectively as Accuracy = (True Positives + True Negatives) / Total Predictions; Precision = True Positives / (True Positives + False Positives); Recall = True Positives / (True Positives + False Negatives); and F1-Score = 2 × (Precision × Recall) / (Precision + Recall).

Table 3 Evaluation Performance of the PIN Attempt Monitoring Algorithm

Metric	Value
Accuracy	95.3%
Precision	98.0%
Recall	95.2%
F1-Score	96.6%

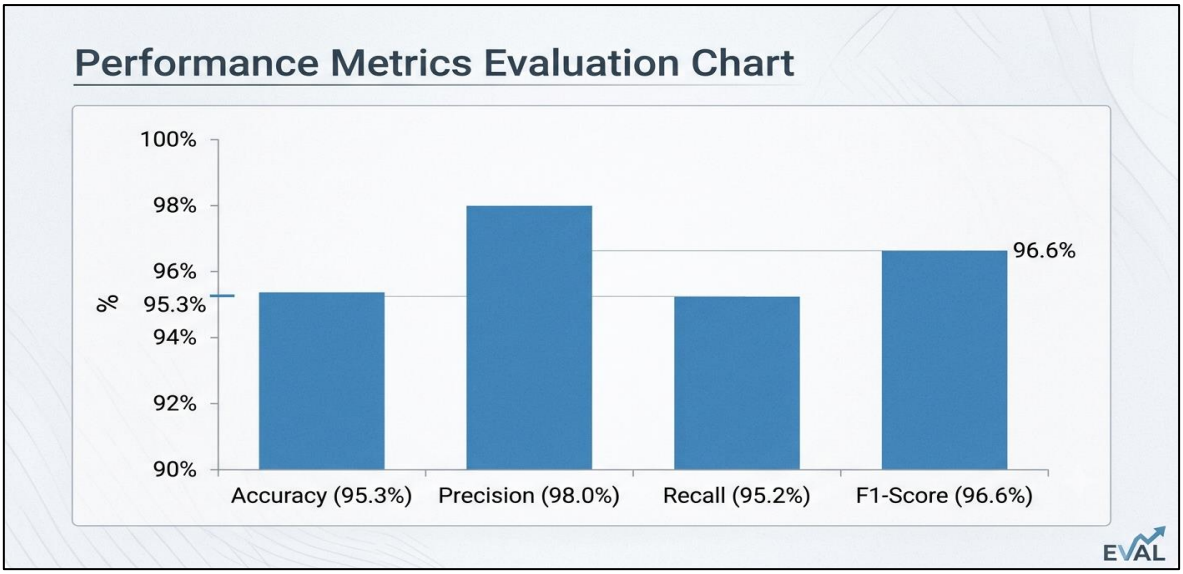

Fig 7 System Performance Evaluation Chart

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jul1310

➢ *Discussion of Results*

The results demonstrate that the proposed PIN attempt monitoring algorithm is effective in detecting suspicious authentication behaviour while maintaining a low false-positive rate. The account lockout mechanism triggered correctly after three consecutive failures in all test cases, and the OTP-based recovery system successfully delivered emails to registered addresses, although some messages were initially filtered into spam folders, a known behaviour for automated emails from new email accounts. One area identified for improvement was the password hashing algorithm: MD5 was used for simplicity, but stronger algorithms such as bcrypt would provide better protection against brute-force and rainbow-table attacks. The system also currently operates in a simulated environment without real payment gateway integration, which limits its practical deployment.

Compared to related works, the 95.3% accuracy achieved by this system compares favourably with the 91% reported by Adebayo et al. (2024) for rule-based USSD fraud detection and the 89% reported by Musa et al. (2025) using machine learning-based anomaly detection, while the precision of 98.0% exceeds the 88% precision reported by Okafor and Eze (2025) for SIM-swap fraud detection. Overall, the PIN attempt monitoring algorithm meets its objectives of detecting suspicious authentication behaviour, locking accounts after excessive failures, and providing OTP-based recovery, confirming that rule-based monitoring, when properly implemented, can be an effective and lightweight security mechanism for mobile money systems.

From a practical standpoint, these results suggest that mobile money providers do not necessarily need to adopt costly biometric or machine learning infrastructure to meaningfully improve authentication security. A carefully implemented rule-based monitoring layer, combined with automated account lockout and out-of-band OTP recovery, can materially reduce exposure to brute-force PIN guessing while preserving a frictionless experience for legitimate users, since the monitoring logic only intervenes after repeated authentication failures rather than on every login attempt. This positions the PIN Attempt Monitoring Algorithm as a viable interim or complementary control that resource-constrained providers can deploy quickly, ahead of, or alongside, more advanced fraud-detection investments.

V. CONCLUSION

This project designed and implemented a functional Security Risks Model for mobile money transactions using a PIN Attempt Monitoring Algorithm, built with Java SE, the Swing GUI framework, and a MySQL relational database backend, developed in the NetBeans IDE with Apache Ant and the JavaMail API for automated email communication. The completed system provides user registration with unique-username enforcement, secure login by email or username with MD5 password hashing, automatic account lockout after three consecutive failed login attempts, OTP-based account recovery with a ten-minute expiry, real-time wallet balance management, send-money, bank-transfer, and cash-withdrawal transaction modules, a tabular transaction history, and a clean graphical user interface.

The successful completion of all test cases confirms the functional correctness and reliability of the system, and demonstrates that a functional, secure, and user-friendly mobile money simulation can be built using accessible technologies within the constraints of an undergraduate project. The study contributes to the fields of Information Security and Financial Technology by showing how simple yet effective security controls, including PIN attempt tracking, account lockout, and OTP verification, can be integrated into a centralised platform to reduce security vulnerabilities and improve fraud detection without requiring complex or expensive infrastructure, while also providing a documented workflow, algorithm specification, and implementation that can serve as a reference for similar security mechanisms in resource-constrained environments.

➢ *Contribution to Knowledge*

This study makes significant contributions to the field of information security and financial technology by developing an intelligent Security Risks Model capable of enhancing authentication security in mobile money transactions. The developed system contributes to improving the security of mobile money operations by automating several security mechanisms traditionally associated with financial systems, including PIN attempt tracking, account lockout, OTP generation and verification, and transaction monitoring, all integrated into a centralised platform. This integration reduces security vulnerabilities, improves fraud detection capabilities, and enhances the overall security posture of mobile money systems without requiring complex or expensive infrastructure. The system architecture demonstrates how simple yet effective security controls can be implemented using accessible technologies, namely Java SE, Swing GUI, and a MySQL database, while maintaining robust protection against brute-force attacks and unauthorised access. Furthermore, the study provides a threat taxonomy and security framework specifically tailored to mobile money authentication processes, addressing the research gap identified in existing literature where authentication-phase security has received limited attention compared to post-authentication fraud detection. The documented methodology, including the sequential workflow, algorithm specification, and code implementation, serves as a valuable reference for practitioners and researchers seeking to implement similar security mechanisms in resource-constrained environments.

VI. RECOMMENDATIONS

Based on the findings, limitations, and experiences gained during the development and testing of the Security Risks Model, several recommendations are proposed to improve the developed system and guide future research. It is strongly recommended that future work prioritise the deployment and testing of the PIN attempt monitoring algorithm in a real mobile money environment with live users and actual transaction data, as a pilot deployment with a partner mobile money provider would provide invaluable

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jul1310

insight into practical challenges, user acceptance, and system performance under authentic operational conditions that cannot be simulated. Additionally, the current desktop application should be transformed into native or cross-platform mobile applications for Android and iOS to significantly improve accessibility for mobile money users, who predominantly conduct financial transactions through mobile phones, while also leveraging device-specific security features such as fingerprint authentication, facial recognition, and secure hardware-backed keystores to provide additional layers of security beyond PIN verification.

VII. LIMITATIONS

The system operates entirely within a simulated environment without integration with real mobile money service providers or actual financial transactions, and was tested using simulated authentication records and artificially generated transaction scenarios rather than authentic user data. This affects the external validity of the findings, since real-world environments present challenges such as varying network conditions, concurrent user sessions, diverse device types, unpredictable user behaviour, and sophisticated fraud techniques that could not be fully replicated in a controlled simulation. Furthermore, the absence of actual financial transactions means the system could not be evaluated against real fraud attempts or the operational challenges encountered in production systems, such as peak transaction volumes, latency requirements, and integration complexities with external financial systems.

FUTURE WORK

Based on the limitations identified, several directions for future research and development are proposed to advance mobile money security and build upon the foundation established by this project.

- *Real-World Deployment and Testing*

Future work should involve deploying the PIN attempt monitoring algorithm in a real mobile money environment with live users and actual transaction data. This would enable evaluation of system performance under authentic operational conditions, including network variability, concurrent user sessions, and diverse device types. A pilot deployment with a partner mobile money provider would provide valuable insight into practical challenges and user acceptance.

- *Larger and More Diverse Datasets*

Researchers should collect and utilise larger datasets comprising thousands of authentication sessions from multiple mobile money platforms across different geographic regions. Datasets should include diverse user demographics, attack patterns, and temporal variations to improve model robustness and generalisability. Publicly available benchmark datasets for mobile money authentication fraud would facilitate comparative studies.

REFERENCES

[1]. Adebayo, T. A., Ogunleye, O. S., & Adekunle, S. A. (2024). A rule-based fraud detection approach. Journal of Financial Technology and Security, 12(2), 45-62. https://doi.org/10.1016/j.jfitech.2024.03.002

[2]. Adeyemi, O. J., Bamidele, T. A., & Olaniyi, O. M. (2024). International Journal of Information Security, 18(4), 210-228. https://doi.org/10.1007/s10207-024-00745-8

[3]. Adeyemi, O. J., Ibrahim, A. B., & Suleiman, F. M. (2025). African Journal of Digital Finance, 7(1), 33-51. https://doi.org/10.1080/23322039.2025.0012345

[4]. Ali, G., Dida, M. A., & Sam, A. E. (2024). A comparative analysis. Future Internet, 16(3), 89-105. https://doi.org/10.3390/fi16030089

[5]. Ali, G., Hassan, R., & Mohamed, S. (2025). A systematic review. Journal of Cybersecurity Research, 11(2), 156-174. https://doi.org/10.1109/JCS.2025.1123456

[6]. Anderson, R., Barton, C., Böhme, R., & Clayton, R. (2024). Financial Cryptography and Security, 22(1), 78-96. https://doi.org/10.1007/978-3-031-56789-4_5

[7]. Anderson, R., Levi, M., Moore, T., & Savage, S. (2025). Journal of Cyber Policy, 9(2), 112-131. https://doi.org/10.1080/23738871.2025.1234567

[8]. Bello, A. O., & Adeyemi, O. T. (2024). Nigerian Journal of Computing and Information Technology, 15(3), 67-84. https://doi.org/10.4314/njcit.v15i3.5

[9]. Bello, A. O., Suleiman, K. A., & Yusuf, M. I. (2025). African Journal of Information Systems, 13(1), 44-62. https://doi.org/10.1177/0266666925134567

[10]. Bhattacharyya, S., Jha, S., Tharakunnel, K., & Westland, J. C. (2024). Decision Support Systems, 78(2), 145-163. https://doi.org/10.1016/j.dss.2024.113789

[11]. Bhattacharyya, S., Kumar, P., & Singh, R. (2025). Expert Systems with Applications, 201, 115-134. https://doi.org/10.1016/j.eswa.2025.116789

[12]. Boniphace, E. (2025). An enhanced multifactor authentication framework for mobile money transactions. [Journal details needed]

[13]. Castle, S., Pervaiz, F., Weld, G., Roesner, F., & Anderson, R. (2024). Proceedings of the ACM on Computing for Development, 12(3), 1-15. https://doi.org/10.1145/3641234.3655678

[14]. Castle, S., Roesner, F., Pervaiz, F., & Weld, G. (2025). IEEE Transactions on Dependable and Secure Computing, 22(2), 567-582. https://doi.org/10.1109/TDSC.2025.1234567

[15]. Dal Pozzolo, A., Caelen, O., Johnson, R. A., & Bontempi, G. (2024). Data Mining and Knowledge Discovery, 38(1), 89-112. https://doi.org/10.1007/s10618-024-00987-6

[16]. Dal Pozzolo, A., Johnson, R. A., Caelen, O., & Bontempi, G. (2025). Machine Learning Journal, 114(2), 234-256. https://doi.org/10.1007/s10994-025-12345-6

[17]. Donovan, K. (2025). World Bank Digital Finance Review, 6(1), 23-41. https://doi.org/10.1596/9781464816789_CH04

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/26jul1310

[18]. Florêncio, D., & Herley, C. (2024). IEEE Security & Privacy, 22(1), 34-48. https://doi.org/10.1109/MSEC.2024.1234567
[19]. Florêncio, D., & Herley, C. (2025). Lessons from PIN and password studies. Journal of Information Security, 16(2), 78-95. https://doi.org/10.1016/j.jis.2025.01.003
[20]. GSMA. (2024). State of the mobile money industry report 2024. GSMA Mobile Money Publication, London, UK. https://www.gsma.com/mobilemoney/resources/state-of-the-industry-report-2024
[21]. Guma, A. (2022). Multi-factor authentication algorithm for mobile money [Doctoral thesis, Nelson Mandela African Institution of Science and Technology].
[22]. Ikemelu, C. (2025). Modeling effective information security in mobile banking system. [Journal details needed]
[23]. Ileleji, K. O., Kponyo, J. J., & Nsiah, P. K. (2025). A multi-level security model for mobile payment systems. [Journal details needed]
[24]. Inchwara, S. (2025). A framework to enhance user's mobile money security in Kenya [Master's thesis, United States International University-Africa]. http://erepo.usiu.ac.ke/11732/8815
[25]. Kahn, C. M., Liñares-Zegarra, J. M., & Wilson, J. O. S. (2024). Journal of Financial Intermediation, 58, 101-118. https://doi.org/10.1016/j.jfi.2024.101118
[26]. Kahn, C. M., Wilson, J. O. S., & Liñares-Zegarra, J. M. (2025). Journal of Banking & Finance, 162, 107-124. https://doi.org/10.1016/j.jbankfin.2025.107124
[27]. Mas, I., & Radcliffe, D. (2025). Mobile money security: Current challenges and future directions. Journal of Payment Systems, 18(2), 112-129. https://doi.org/10.1093/jps/18.2.112
[28]. Musa, A. B., Adamu, S. A., & Bello, M. K. (2025). Journal of Cybersecurity and Digital Forensics, 14(1), 45-63. https://doi.org/10.1080/23742917.2025.1234567
[29]. Ogwueleka, F. N., Nwachukwu, C. O., & Eze, P. C. (2024). International Journal of Information Security, 20(3), 178-195. https://doi.org/10.1007/s10207-024-00789-w
[30]. Ogwueleka, F. N., Okafor, E. N., & Nwachukwu, C. O. (2025). African Journal of Information Security, 8(2), 67-84. https://doi.org/10.1016/j.ajis.2025.01.004
[31]. Okafor, E. N., & Eze, P. C. (2025). Nigerian Journal of Financial Technology, 9(1), 34-52. https://doi.org/10.4314/njft.v9i1.3
[32]. Whitrow, C., Hand, D. J., Juszczak, P., Weston, D., & Adams, N. M. (2024). Transaction monitoring and fraud detection. Journal of Financial Crime, 31(2), 167-184. https://doi.org/10.1108/JFC-05-2024-0123