DOI: https://doi.org/10.52756/ijerr.2024.v45spl.012 *Int. J. Exp. Res. Rev.*, Vol. 45: 153-172 (2024)

Original Article | *Peer Reviewed* | *Open Access*

International Journal of Experimental Research and Review (IJERR)

ISSN: 2455-4855 (Online)
www.iaph.in
ISSN 2455-4855

AI-Enhanced Secure Mobile Banking System Utilizing Multi-Factor Authentication

Mohd. Salman[1]* and Rahul Kumar Mishra[2]

[1]Department Department of Computer Science and Engineering, IFTM University, Moradabad, UP-244001, India;
[2]Department of Computer Science and Engineering, IFTM University Moradabad, UP-244001, India

E-mail/Orcid Id:

MS, salmank64@gmail.com, https://orcid.org/0000-0002-5578-0601;
RKM, rahulmishra@iftmuniversity.ac.in, https://orcid.org/0000-0003-2274-4767

Article History:

Received: 17th June, 2024
Accepted: 30th Oct., 2024
Published: 30th Nov., 2024

Keywords:

AI-Powered Security, Biometric Authentication, Cyber-attacks, Machine learning, Mobile Banking Security, Multi-Factor Authentication.

How to cite this Article:

Mohd. Salman and Rahul Kumar Mishra (2024). AI-Enhanced Secure Mobile Banking System Utilizing Multi-Factor Authentication. *International Journal of Experimental Research and Review*, *45*, 153-172.

DOI: https://doi.org/10.52756/ijerr.2024.v45spl.012

Abstract: The increasing reliance on mobile banking has significantly heightened the need for robust security mechanisms to protect users from unauthorized access and fraudulent activities. As mobile banking continues to grow in popularity, safeguarding financial transactions and personal data becomes a top priority. This paper introduces an AI-enhanced secure mobile banking system that leverages Multi-Phase Authentication (MPA) to strengthen the authentication process. In this system, artificial intelligence is integrated with traditional authentication methods, creating a dynamic framework that assesses the risk level associated with each user interaction. Based on this real-time risk assessment, the system adjusts the authentication requirements, making them more stringent when higher risks are detected and more lenient when the risk is lower. This adaptive mechanism not only enhances the security of mobile banking by providing multiple layers of protection but also improves the user experience by reducing unnecessary authentication steps that can cause frustration and delay. The proposed system's effectiveness is validated through a series of simulations and case studies, which demonstrate significant improvements in key security metrics. These include a marked reduction in instances of fraud and lower false positive rates, which indicate that the system can accurately distinguish between legitimate and suspicious activities without imposing undue burden on users. Overall, the results of this study highlight the potential of AI-enhanced multi-phase authentication to provide a scalable and user-friendly solution for secure mobile banking. This approach represents a promising direction for the future of digital financial services, offering a balance between rigorous security and seamless user experience.

Introduction

The rapid advancement of technology has reshaped the banking industry, with mobile banking becoming a key component of financial services worldwide. Customers now have the ability to conduct banking transactions, manage accounts, transfer funds, and access various financial services directly from their mobile devices (Madasamy, 2022; Bhattacharya et al., 2024). This shift toward digital banking has been driven by the convenience, speed, and accessibility that mobile platforms offer. However, as the adoption of mobile banking continues to grow, so too does the potential for security vulnerabilities (Hassan et al., 2023; Sambrow and Iqbal, 2022). Mobile banking platforms, while convenient, have become prime targets for cybercriminals seeking to target weak points in security protocols to secure illegal access to sensitive financial information (Awad et al., 2024).

Mobile banking security has traditionally relied heavily on single-level authentication approaches, like passwords, to verify user identities. While these methods offer a basic level of security, they are increasingly inadequate in the face of sophisticated cyber threats (Gautam, 2023; Cole et al., 2009; Batra and Kalra, 2016). Passwords are vulnerable to phishing attacks, social engineering, or exhaustive search techniques. Once compromised, unauthorized

***Corresponding Author:** salmank64@gmail.com

access to mobile banking accounts can lead to significant financial loss, identity theft, and erosion of customer trust (Hassan and Shukur, 2019; Omariba et al., 2012; Hassan et al., 2020).

In response to these challenges, Multi-phase authentication has emerged as a more robust security measure (Ometov et al., 2018). Multi-phase authentication ensures users must offer two or more verification factors before gaining account access. Generally, these factors are comprised of something the user knows, something the user possesses and something the user is (Fan et al., 2017; Shaju and Panchami, 2016; Okpara and Bekaroo, 2017). By demanding several verification factors (Harish et al., 2019),

While multi-phase authentication enhances security, it is not without its challenges. The implementation of Multi-Phase authentication can sometimes be cumbersome for users, leading to potential friction in the user experience (Huseynov and Seigneur, 2019). Additionally, as cyber threats continue to evolve, even multi-phase authentication systems can be targeted by advanced attacks, such as SIM swapping or biometric spoofing. To address these challenges, there is a growing interest in integrating Artificial Intelligence (AI) into mobile banking security systems (Kaur and Devgan, 2015; Emeka and Liu, 2017).

AI has the potential to revolutionize cybersecurity by providing dynamic, real-time analysis and response capabilities (Huseynov and Seigneur, 2019). AI-driven algorithms can analyze vast amounts of data to detect patterns and anomalies that may indicate a security breach. For instance, AI can monitor user behavior to identify unusual login attempts, flag potentially fraudulent transactions, or adapt authentication requirements based on the perceived risk level. When combined with Multi-Phase authentication, AI can enhance the security of mobile banking systems by providing an additional layer of defense that is both adaptive and proactive (Hassan and Shukur, 2021; Sudar et al., 2017; Kogan et al., 2017).

Despite the advancements in security technologies, mobile banking systems continue to face significant security challenges. The increasing sophistication of cyberattacks, coupled with the limitations of traditional authentication methods, presents a serious risk to the safety and integrity of mobile banking platforms (Isaac and Sherali, 2014). Phishing attacks, for example, can trick users into divulging their credentials, while man-in-the-middle attacks can intercept and alter communications between the user and the banking server. Credential stuffing, where attackers use stolen credentials from other breaches to gain access to accounts, is also a growing concern (Dwivedi et al., 2013).

While multiple-phase verification provides an extended level of security, it is not immune to attacks. Cybercriminals have developed methods to bypass multi-phase authentication, such as through social engineering techniques or exploiting weaknesses in the multi-phase authentication process itself. Additionally, the use of static authentication factors, such as passwords and even biometrics, can still be vulnerable to certain types of attacks (Yang et al., 2019; Gualdoni et al., 2017).

There is a critical need for a more advanced security framework that can address these challenges. The integration of AI into mobile banking security, combined with multi-phase authentication, presents a promising solution. An AI-enhanced secure mobile banking system could dynamically assess the risk of each transaction or login attempt and adjust the verification criteria accordingly. This method could strengthen security and improve the user experience by reducing unnecessary friction during the authentication process (Venugopal and Viswanath, 2016; Roy and Venkateswaran, 2014; Hassan et al., 2020a).

This research aims to develop an AI-enhanced secure mobile banking system that utilizes multi-factor authentication to guard t against a wide range of cyber-attacks. The proposed system will leverage AI to regularly check and analyze user behavior, detect potential security threats in real-time and adjust authentication requirements based on the degree of risk. By doing so, the system will offer a more secure and user-friendly mobile banking experience (Ataya and Ali, 2019; Hassan et al., 2020b).

Motivation

The motivation behind the development of the AI-enhanced secure mobile banking system utilizing

DOI: https://doi.org/10.52756/ijerr.2024.v45spl.012

multi-phase authentication stems from several key factors:

Rising Security Threats in Mobile Banking

Limitations of Traditional Authentication Methods

#Enhancing User Experience without Compromising Security

Leveraging AI for Real-Time Risk Assessment

Scalability and Adaptability for Modern Banking:

#Addressing Regulatory and Compliance Requirements:

In summary, this work aims to address the growing security challenges in mobile banking by developing an advanced, AI-driven multi-phase authentication system that enhances both security and user experience. The system aims to be a scalable, adaptable, and forward-looking solution that meets the needs of modern banking environments while providing a robust defense against emerging cyber threats.

Literature Review

Mobile Banking Security Current mobile banking security measures include encryption, secure communication protocols, and user authentication. Despite these measures, the increasing complexity of cyber-attacks necessitates more advanced solutions Multi-Phase authentication involves using multiple verification methods, such as passwords, tokens, and biometrics. While Multi-Phase authentication enhances security, it can also be cumbersome for users (Chaudhry et al., 2016). This paper explores how AI can streamline and strengthen multi-phase authentication processes. AI in Cybersecurity AI technologies, including machine learning and NLP, are increasingly used in cybersecurity to detect and respond to threats in real-time. These technologies can identify patterns and anomalies that may indicate security breaches. AI and Multi-Phase authentication in Financial Services Previous research has highlighted the potential of AI and Multi-Phase authentication in enhancing security within financial services (Skračić et al., 2017; Ibrahim, 2018). This paper builds on existing research by proposing an integrated system for mobile banking.

The rapid evolution of mobile banking has transformed financial transactions, but it has also introduced new security risks. Integrating multi-phase authentication and artificial intelligence (AI) is becoming increasingly important in countering these risks. This literature review explores the current research and developments in AI-enhanced secure mobile banking systems utilizing Multi-Phase authentication (Elliot and Talent, 2018; Kanimozhi and Kamatchi, 2017; Tan and Samsudin, 2018).

Multi-Phase Authentication

Concept and Importance

Multi-Phase authentication is enhanced by enforcing security measures that require users to authenticate their credentials using various verification factors. The three primary types of factors include:

#Factors Based on Knowledge: Passwords, PINs.

Factors Based on Possession: Security tokens, smartphones.

Factors Based on Inherence: Biometrics such as fingerprints and facial recognition.

Effectiveness

Multi-Phase authentication significantly enhances enhancing security by incorporating additional layers of protection, increasing the difficulty of unauthorized access. It is particularly important in mobile banking, where the likelihood of unauthorized access is high due to the nature of mobile devices and the internet (Tellini and Vargas, 2017).

Artificial Intelligence (AI) in Security

Applications of AI and Machine Learning: AI and ML enhance security systems by analyzing vast amounts of data to detect anomalies and predict threats(Huseynov & Seigneur, 2017). Key applications include:

 # Fraud Detection: Identifying suspicious transactions through pattern recognition.

 # Behavioral Analysis: Monitoring user behavior to detect deviations from normal patterns.

Benefits: AI improves security by offering real-time, adaptive responses to emerging threats. It enables more sophisticated analysis and prediction capabilities compared to traditional methods.

DOI: https://doi.org/10.52756/ijerr.2024.v45spl.012

AI-Enhanced Multi-Phase Authentication Systems

Integration of AI with Multi-Phase Authentication: The integration of AI with Multi-Phase authentication systems enhances traditional authentication by:

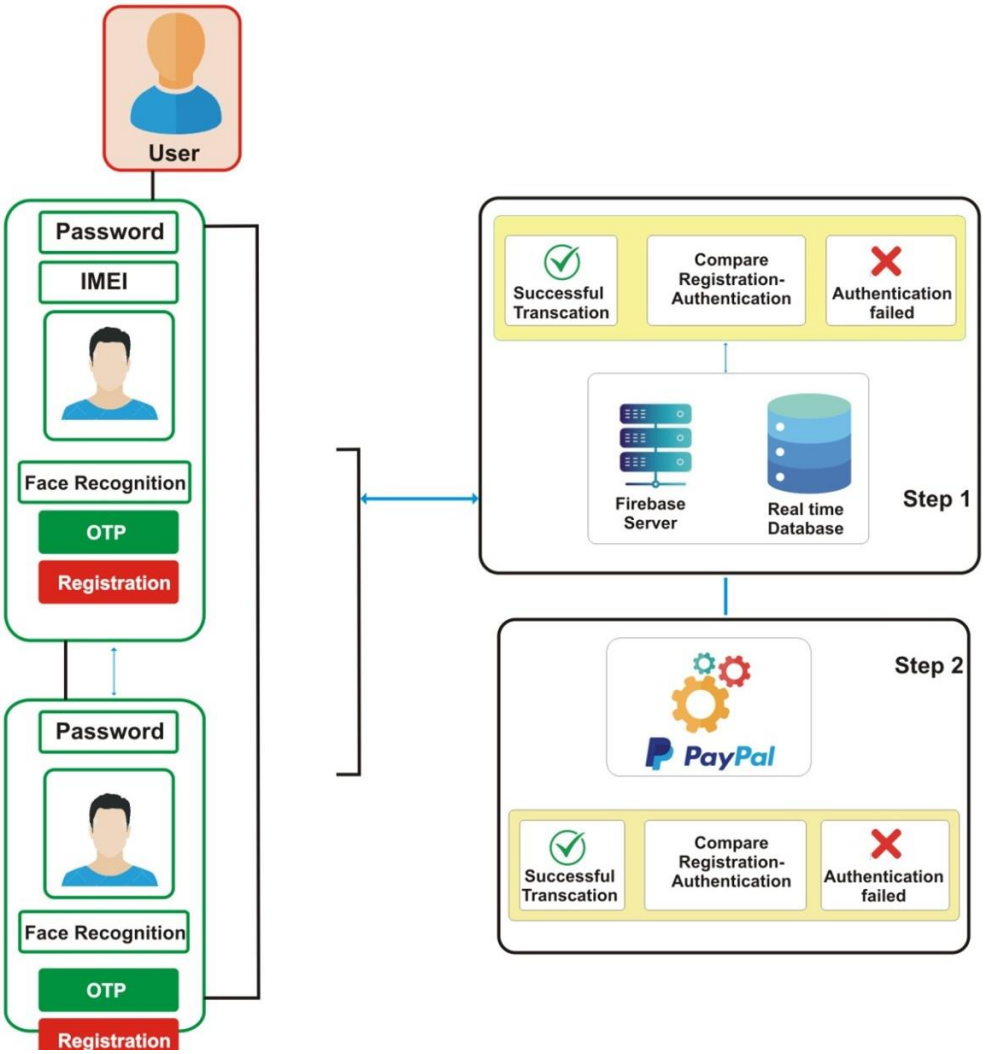


Figure 1. Framework of the Proposed Technique.

 # Adaptive Authentication: AI adjusts authentication requirements based on contextual factors such as user location and device.

 # Behavioral Biometrics: AI analyzes patterns in user behavior, such as typing speed and navigation habits, to enhance security.

Advantages: AI-enhanced multi-phase authentication provides a dynamic and context-aware authentication approach, improving security and user experience.

AI-enhanced multi-factor authentication represents a significant advancement in securing mobile banking systems. By integrating AI with traditional multi-phase authentication methods, these systems provide improved security through adaptive and context-aware authentication processes. Ongoing research and practical implementations continue to refine these systems, addressing user experience, privacy, and algorithmic robustness challenges. This literature review offers a thorough overview of current practices, benefits, challenges, and future directions in AI-enhanced multi-phase authentication for mobile banking (Wang et al., 2020; Nwabueze et al., 2017; Dasgupta et al., 2016).

Primary Contribution of the Suggested Approach: The main contribution of the proposed work in "AI-Enhanced Secure Mobile Banking System Utilizing Multi-Factor Authentication" likely revolves around the integration of artificial intelligence (AI) with multi-phase verification to enhance the security of mobile banking platforms. Here's a breakdown of the potential key contributions and results:

DOI: https://doi.org/10.52756/ijerr.2024.v45spl.012

1. **AI-Driven Risk Assessment:** The integration of AI into the Multi-Phase authentication process enables real-time risk assessment, allowing the system to adapt authentication requirements based on the detected threat level. This dynamic approach enhances the security of mobile banking systems.

2. **Enhanced Multi-Factor Authentication:** The proposed system incorporates multiple layers of authentication, including traditional (password, PIN), biometric (fingerprint, facial recognition), and contextual factors (location, device recognition). The AI component intelligently determines when additional factors are required, balancing security with user convenience.

3. **User-Centric Security:** By focusing on minimizing disruptions to the user experience, the system optimizes when and how additional authentication factors are requested, ensuring that legitimate users face minimal friction while maintaining a high level of security.

4. **Scalability and Flexibility:** The proposed system is engineered for scalability and adaptability to various banking environments, designed as a flexible solution for various types of financial institutions.

Table 1. List of Symbols Used.

Notations	Description
Ui	User
IDi	Unique Identifier of User
PWDi	Password of the User
FRi	face recognition feature of User
IMEI Identity of user	International Mobile Station Equipment
d *e* *n* numbers *(p* and *q)*	Private key in *RSA* Public key in *RSA* Computed as product of chosen prime
RC *Time Password)*	Random Challenge *(in this context—One*
CS	Concatenated String *(Ui)*
ES*Q*	Encrypted String
PK*P*	Plain Private Key
PK*E*	Encrypted Private Key
E (Data, Encryption Key)	Encryption Function
R*F* (*PWDi, IMEI, FFi)D* (Data, Encryption Key)	Result of Comparison Function Decryption Function

Conceptual Design

To ensure user satisfaction, each software must incorporate an effective structure. Structure is a procedure for arranging components most efficiently to achieve a specific objective. The verification design involves two key phases: registration and authentication. Before using INDFunds, users must go through the registration process, where they enter their information. The authentication phase then verifies this information (AliBabaee and Broumandnia, 2018; Houngbo et al., 2019; Vengatesan et al., 2020).

In the verification stage, the client provides verification details, such as a passcode, face recognition, and an OTP generated for mobile. This information is then transferred to a real-time Firebase database, where it is securely stored. In the

DOI: https://doi.org/10.52756/ijerr.2024.v45spl.012

verification stage, the client must re-enter their verification details on mobile. The server compares these credentials with the stored registration information to verify the client's identity.

The proposed method utilizes Android Studio, which links the server and application through various Application Programming Interfaces (APIs). This approach simplifies the development of an e-wallet by leveraging cutting-edge features and libraries offered by the Android platform. A real-time Firebase database is used for the back-end and Payment Service Provider are employed.

The application is developed and the next phase is the evaluation process, which tests the performance of INDFunds through requirement testing. The main objective of this testing is to validate the functionality and ensure it meets the desired criteria. The suggested conceptual framework of the verification process is illustrated in Figure 1.

Authentication Workflow

The authentication procedure is a critical component of the suggested Zam_wallet. The emphasis here is on user verification, which is divided into two stages: sign-up and verification. This section details the verification procedure, including the notations used in the proposed method.

Registration Phase

To access the service, users must complete a one-time registration process, where their information is collected. The server uses this information along with the login method to verify the user's legitimacy. Figure 2 outlines the proposed registration procedure. In this phase, users are required to create their accounts by providing the necessary information. The verification phases include:

1. **Begin** the registration procedure.
2. The user provides their information, including Password, International Mobile Equipment Identity (IMEI), and Face Recognition (FRi) on the mobile device.
3. The system generates two large prime numbers, rrr and sss, and computes C=r×sC = r \times sC=r×s.
4. The system selects integers aaa and bbb that satisfy e×d mod ((r−1)×(s−1))=1e \times d \, \text{mod} \, ((r-1) \times (s-1)) = 1e×dmod((r−1)×(s−1))=1.
5. An OTP is generated and sent to the user's inputted numbers.
6. The user inputs the OTP.
7. If the OTP matches, proceed to Phase 8; otherwise, return to Phase 6.
8. A Random Challenge (RC) is generated for the user, where the IDi is created using PWDi, IMEI, and FRi.
9. All user information is stored.
10. **End** the registration process.

In the verification stage, the user provides the necessary personal information to log into the system. Only after completing this phase can the user access the system.

Authentication Stage

The authentication server must verify their identity when the user attempts to log in. If the provided credentials match the stored values, authentication is successful. The process includes two stages: login and verification. Initially, the user logs in using their approved password, fingerprint, and OTP. Once logged in, they can access their account information. To finalize a transaction, the user must verify their identity again using their thumb impression. The operation will only be processed after successful fingerprint authentication. Figure 3 illustrates the authentication process, which involves the following Phases:

1. **Start** the authentication process.
2. The user enters their information via the mobile device, including PWDi, IMEI, and FRi.
3. The system checks for verification by applying encryption to the combined string, resulting in ESQ1 = E(CS1).
4. The encryption function is applied to the unencrypted private key and combined string, generating PKE = E(PKP, CS1).
5. The system compares RF1 with the decryption method: if RF1 = D(PKE, CS), continue
6. An OTP is generated and sent to the user's inputted numbers.
7. The user inputs the OTP.
8. If the OTP matches, proceed to Phase 9; otherwise, return to Phase 7.
9. Access is granted.

DOI: https://doi.org/10.52756/ijerr.2024.v45spl.012

10. **Stop** the authentication process.

IV. Comprehensive Design

The proposed authentication framework goes beyond just the registration and verification stages. It also features component diagrams and sequence diagrams that illustrate the overall structure and process.

Modules in INDFunds

The INDFunds application comprises two primary modules: the administrator module and the user module.

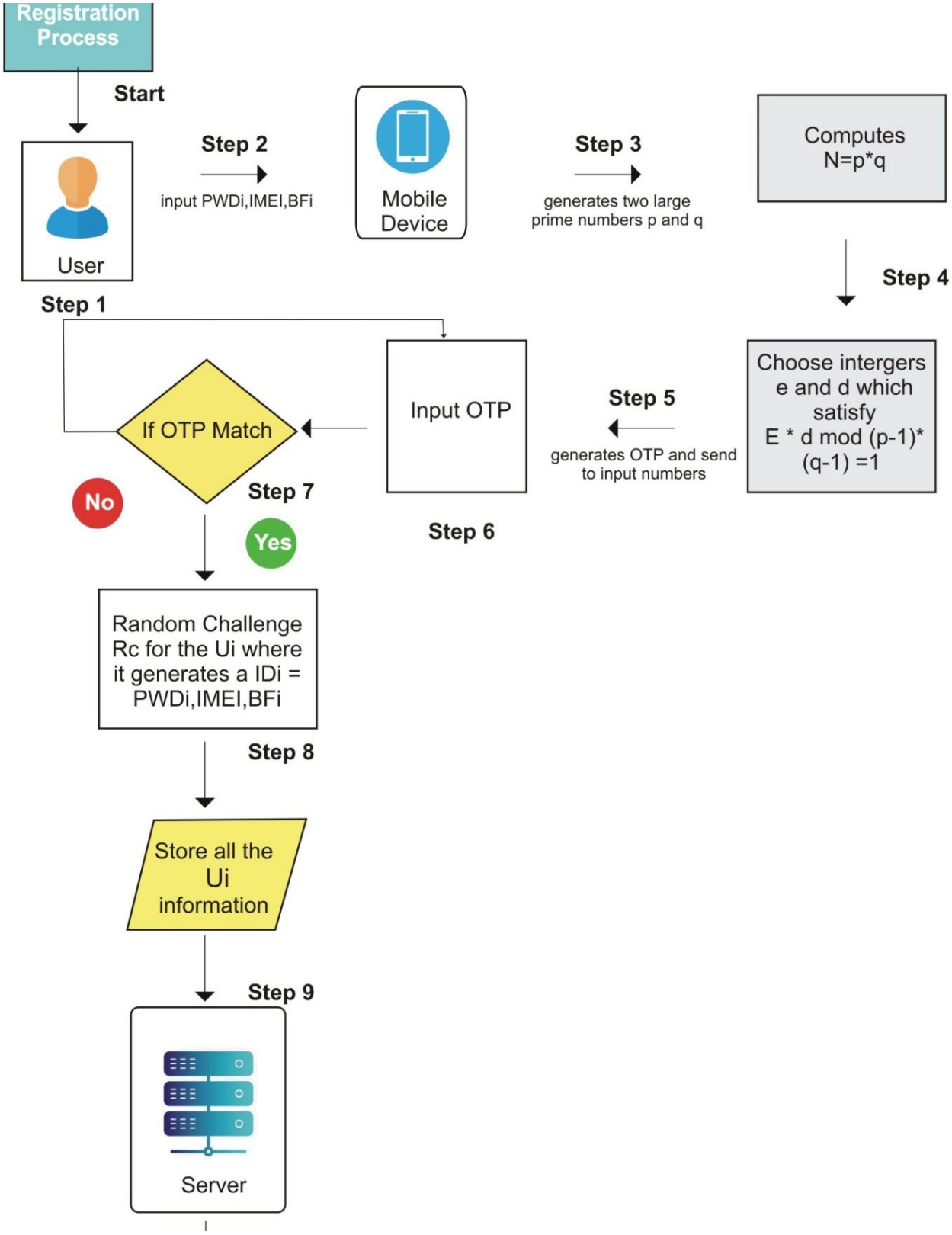


Figure 2. Registration Workflow.

DOI: https://doi.org/10.52756/ijerr.2024.v45spl.012

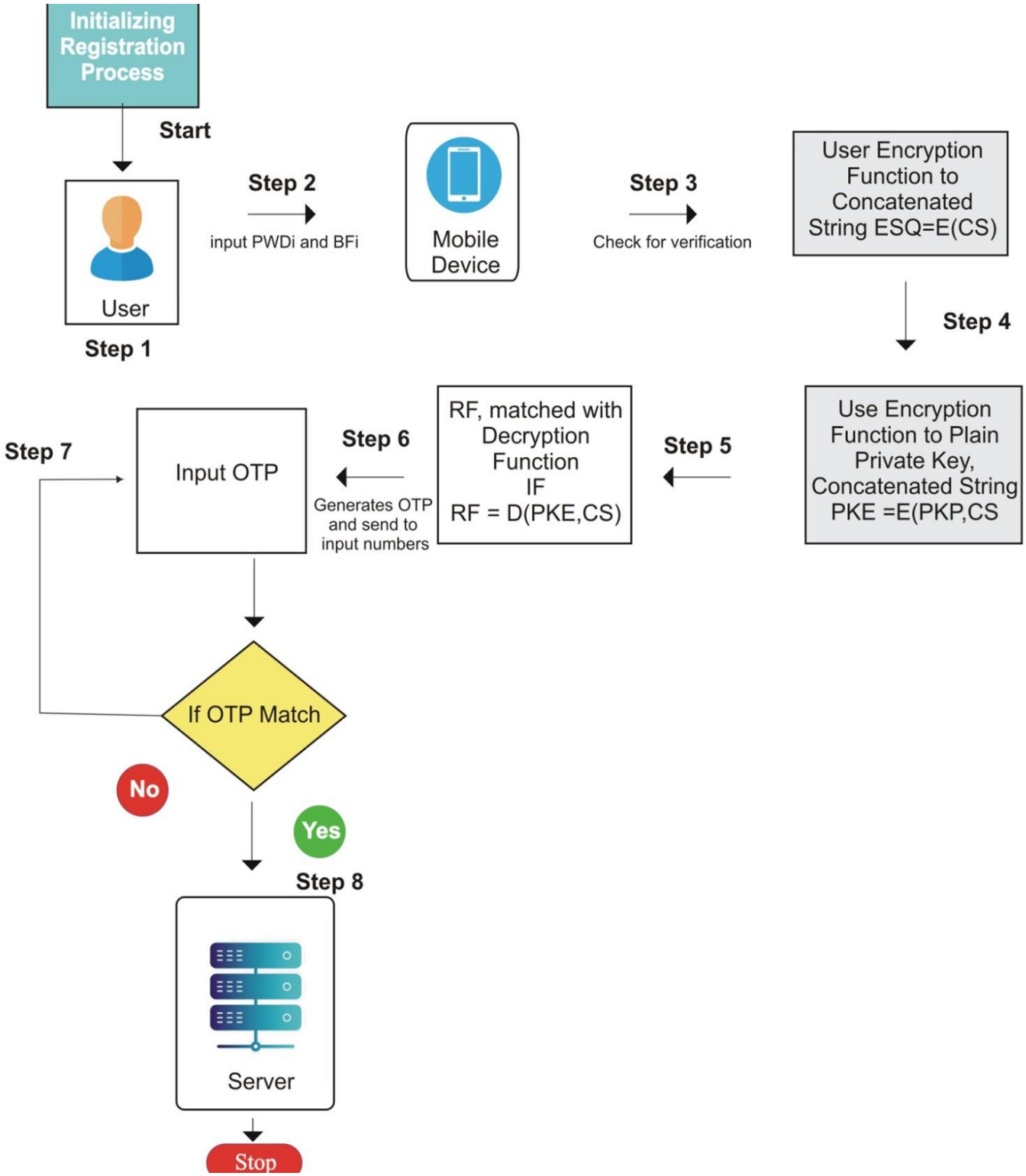


Figure 3. Authentication Workflow.

DOI: https://doi.org/10.52756/ijerr.2024.v45spl.012

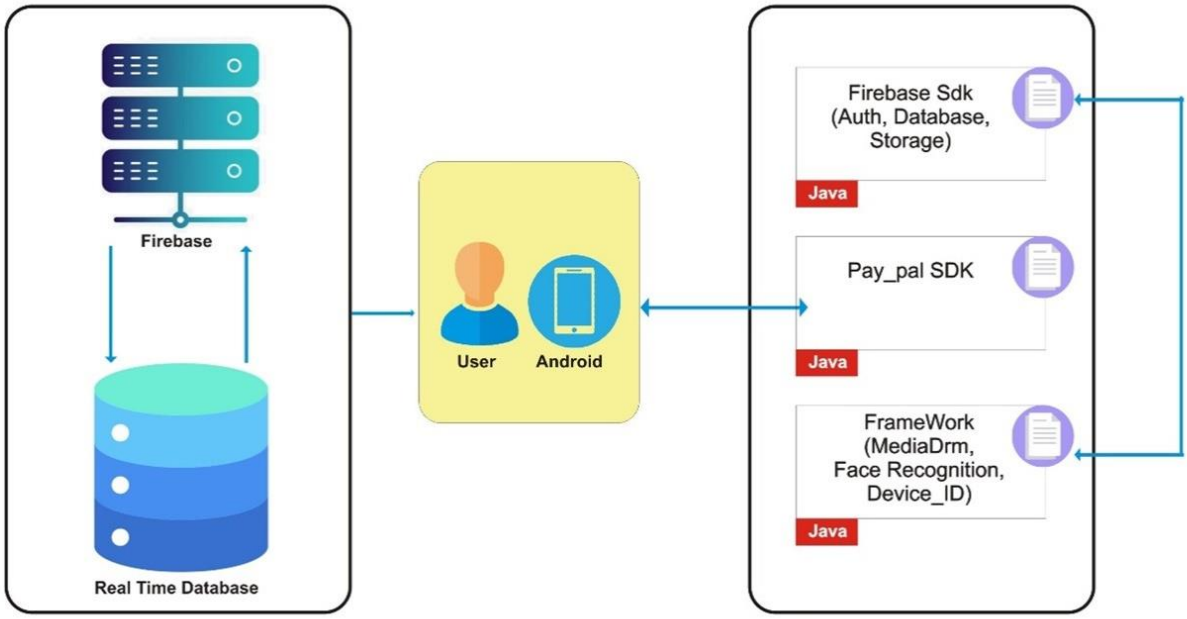


Figure 4. Component Diagram for the Proposed Method.

Administrator Module: This module is used for managing the system, including entering login details, adding/updating/deleting items, managing user information, and updating order statuses. It serves as the system management interface, allowing the admin to interact with the database, update or remove user data, review orders, collect transaction summaries, and handle instructions from registered users. The admin, who often manages the system, operates this back-end module. The proposed method utilizes a real-time Firebase database with automated user updates. These models enhance data security within the application and help mitigate various types of attacks, including malware detection and cloud security threats (Khan et al., 2023; Mostafa et al., 2023; Ahmad et al., 2023; Polas et al., 2022).

User Module: This is the front-end graphical user interface of the application. Through this interface, users can register, sign in, update profiles, view transactions, add funds to their accounts, transfer money, and sign out.

Component Diagram Overview

The component diagram illustrates how the framework is broken into various subsystems. INDFunds include multiple layers.

Firebase Realtime Database: This is linked to Firebase management and is responsible for real-time data handling.

Application Controller: Positioned as the middle layer, it connects with both the Firebase SDK and Payment Service Provider SDK, as well as the framework components. The controller (typically a mobile device) facilitates communication between the database and the GUI. The component diagram visually represents how these various elements interact within the application (Alzu'bi et al., 2021; Wang et al., 2020; Massaro and Galiano, 2020; Ali et al., 2020).

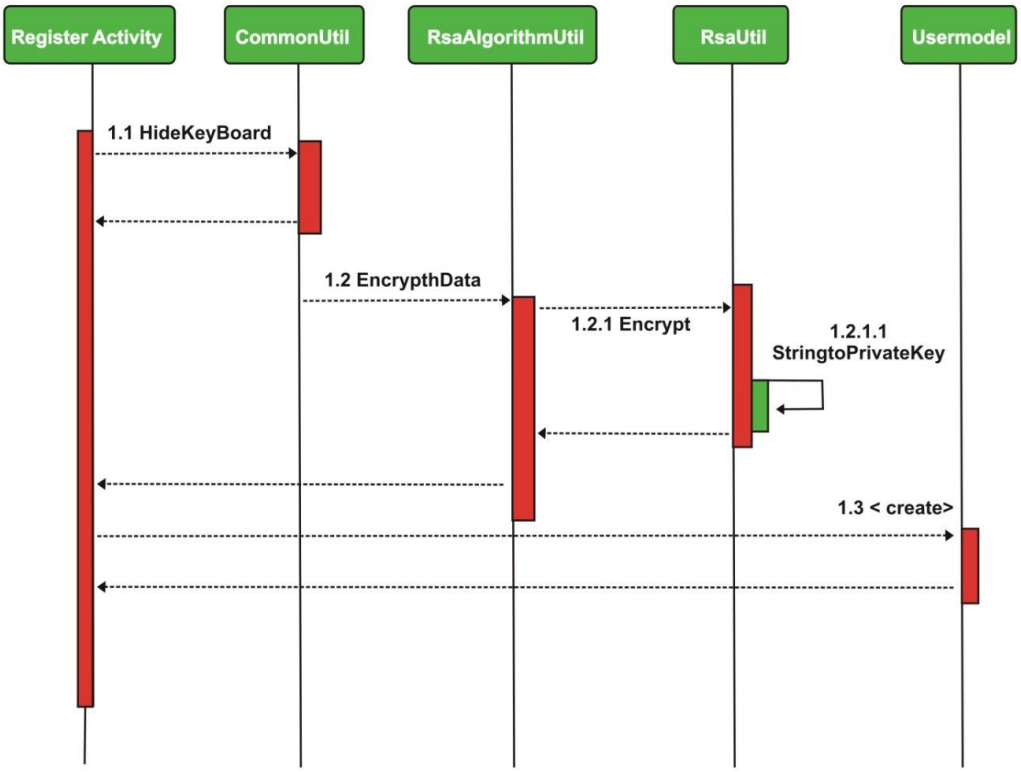
Sequence Diagram Overview

How do objects interact in a particular sequence? It shows the order of messages exchanged between objects or components over time, depicting the dynamic behavior of a system. It is an effective tool for visualizing and validating different runtime scenarios, helping to predict system behavior and identify the roles that classes may play in redrawing a fresh framework. This part presents the sequence diagram for the suggested INDFunds app's user interface, where users can register, authenticate, add funds, and transfer funds.

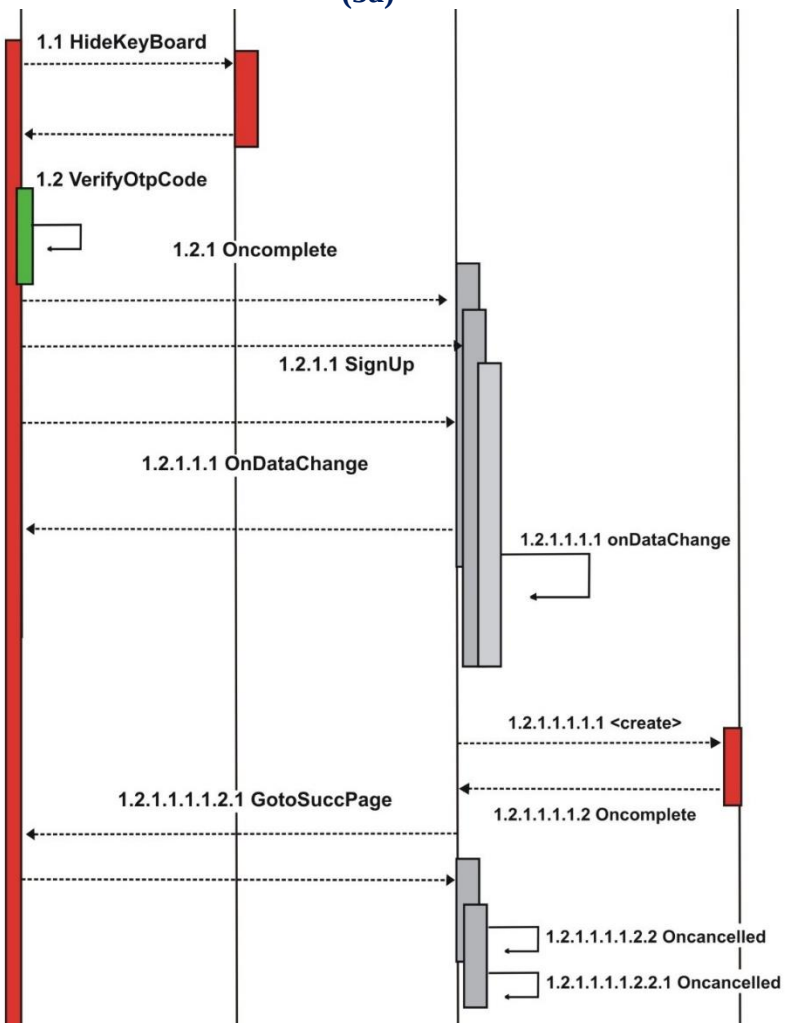
User Sign-Up Feature

The suggested system includes two registration processes: the registration and OTP verification tasks. Establishing a profile is the initial phase of using the software application, and the verification process is straightforward. Firstly, the user must fill out all required information in the verification form, which includes four fundamental verification types: passcode, device ID, face recognition, and OTP. After completing the form, the user submits it (James and Garnett, 2024).

DOI: https://doi.org/10.52756/ijerr.2024.v45spl.012



(5a)



(5b)

Figure 5(a) Registration Task of the Suggested Approach and (b) OTP Verification Task of the Suggested Approach.

DOI: https://doi.org/10.52756/ijerr.2024.v45spl.012

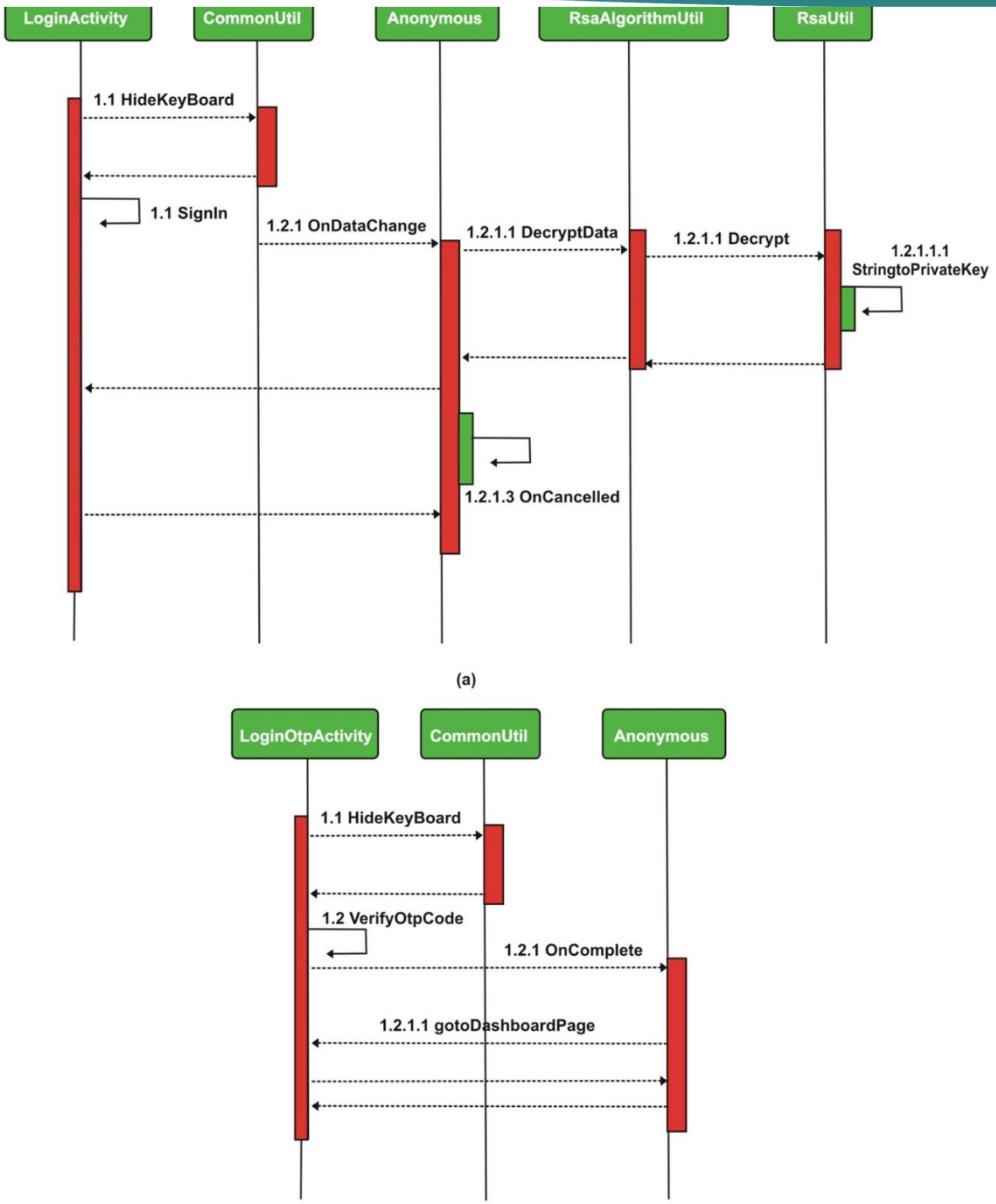


Figure 6(a). Login Task of the Suggested Approach and (b) Login OTP Verification Task of the Suggested Approach.

The server-side procedure of the signup form checks that all required information is correctly filled out. If there are any errors, the system displays a message detailing the issue and redirects the user back to the signup form. In the suggested INDFunds, user passcodes are secured with encryption through the RSA (Rivest-Shamir-Adleman) algorithm for security purposes. Additionally, this "hide keyboard" is employed to prevent the on-screen keyboard from causing automatic correction issues, enhancing security.

Once the data is encrypted, employ an RSA algo. The public encryption key creates a client profile for the registration task. After the input is completed, a one-time password verification code is sent to the phone number associated with the user's account. If all details are OK, the user is instantly redirected to the login page as a client. Figure 5a illustrates the registration process, while Figure 5b depicts the OTP verification task.

n panel (a):

Phase 1.1: The Register_Task hides the keyboard using the Common_Util utility functions and sends a message back to the Register_Task.

Phase 1.2: The Encrypt_Data function in RSA_AlgorithmUtil processes the user's input data.

DOI: https://doi.org/10.52756/ijerr.2024.v45spl.012

Phase 1.2.1: The data provided by the user is encrypted using RSA_Util.

Phase 1.2.1.1: The data is encrypted with RSA algorithm using the public key in RSA_Util, and a response message is sent from RSA_AlgorithmUtil to Register_Task.

Phase 1.3: The Register_Task transfer a request to User_Model to set up an account and awaits.

In panel (b):

Phase 1.1: The Register_Otp_Task hides the keyboard using Common_Util and sends a message back to the Register_Otp_Task.

Phase 1.2: The verify_Otp function initiates and concludes with Register_Otp_Task, awaiting a response.

Phase 1.2.1: Once the OTP has been validated, the on_Complete function waits for a response from unidentified (containing utility operations) to handle the request.

Phase 1.2.1.1: Unidentified replies to the Register_Otp_Task request, confirming successful signup.

Phase 1.2.1.1.1: Register_Otp_Task sends an on_Data_Change request to unidentified.

Phase 1.2.1.1.1.1: The on_Data_Change operation is executed and completed within the time.

Phase 1.2.1.1.1.1.1: Unidentified submits a request to User_Model to generate a user profile in the live database.

Phase 1.2.1.1.1.1.2: User_Model responds to anonymous with an on_Complete response.

Phase 1.2.1.1.1.1.2.1: Unidentified then sends a final response to Register_Otp_Task, confirming successful registration.

User Login Feature

The user must provide a passcode, undergo face recognition, and complete one-time code verification to access the system. The user's passcode is encoded using the RSA algorithm with the secret key within the RSA module. When the user submits their login credentials, the system verifies the information against the details stored in the database. If the information is correct, a one-time code verification is sent to the registered mobile. An error alert is shown for the corresponding field if there is a mismatch.

Upon successful verification, the user is redirected to the main page with their login profile. The login process consists of two activities: login Task and OTP verification Task. Figure 6a shows the sequence diagram for the user login Task. Figure 6b illustrates the OTP verification Task.

In panel (a):

Phase 1.1: The Login_Task hides the keyboard using Common_Util and returns to the LoginTask.

Phase 1.2: The signing process begins and concludes with Login_Task, which then awaits a response.

Phase 1.2.1: Login_Task sends an on_Data_Change request to unidentified.

Phase 1.2.1.1: Unidentified forwards the request to decrypt_Data within RSA_Algorithm_Util.

Phase 1.2.1.1.1: Input data from the user is decrypted and transfer to RSA_Util.

Phase 1.2.1.1.1.1: The information is decrypted using the RSA module with the secret key in RSA_Util, and a response message is sent from RSA_Algorithm_Util to unidentified.

Phase 1.2.1.2: An unidentified party sends a response back to Login_Task, which then proceeds to the sign-in page.

Phase 1.2.1.2: At last, Login_Task handles the on_Cancelled event.

In panel (b):

Phase 1.1: The Login_Otp_Task hides the keyboard using Common_Util and returns to Login_Otp_Task.

Phase 1.2: The verify_Otp process starts and concludes with Login_Otp_Task, awaiting a response.

Phase 1.2.1: After the OTP has been confirmed, the on_Complete task holds for a response from an unidentified party to process the request.

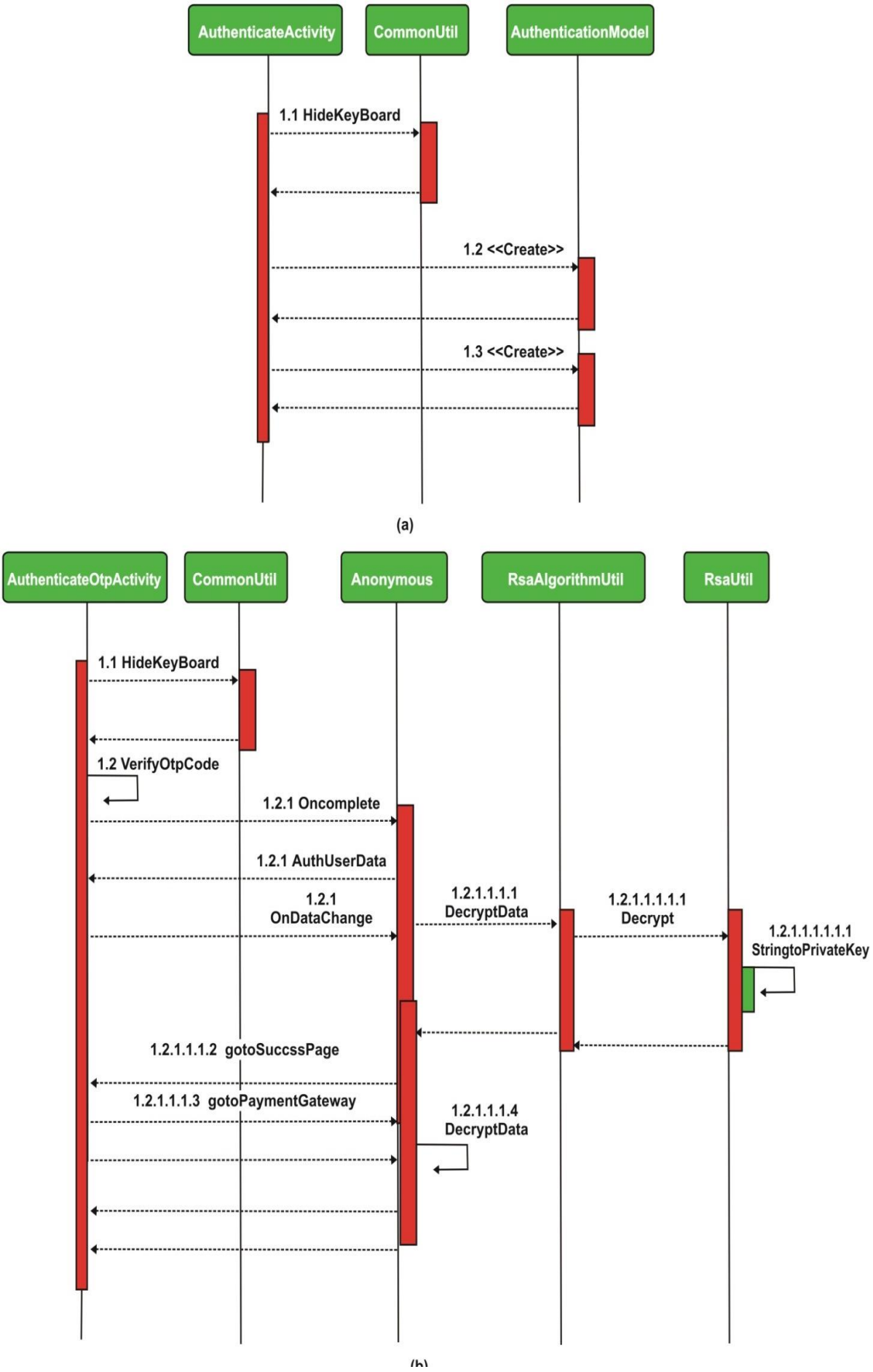
Phase 1.2.1.1: Unidentified party responds to Login_Otp_Task, which then navigates to the Dashboard_Page.

User Authentication Feature

Verification is a critical feature of INDFunds, encompassing essential top-ups and money transfer functions. Users must provide a password, face recognition, and one-time code authentication to access the system. The user's passcode is secured with the RSA algorithm using a security key within the RSA module (Balani et al., 2024).

The OTP identification code must be validated against the user's credentials during authentication. When the user checks their login information and clicks the OTP option, the system sends the authentication code sent to the registered mobile. An

DOI: https://doi.org/10.52756/ijerr.2024.v45spl.012


AuthenticateActivity
CommonUtil
AuthenticationModel
1.1 HideKeyBoard
1.2 <<Create>>
1.3 <<Create>>
(a)
AuthenticateOtpActivity
CommonUtil
Anonymous
RsaAlgorithmUtil
RsaUtil
1.1 HideKeyBoard
1.2 VerifyOtpCode
1.2.1 Oncomplete
1.2.1 AuthUserData
1.2.1 OnDataChange
1.2.1.1.1.1 DecryptData
1.2.1.1.1.1.1 Decrypt
1.2.1.1.1.1.1.1 StringtoPrivateKey
1.2.1.1.1.2 gotoSuccssPage
1.2.1.1.1.3 gotoPaymentGateway
1.2.1.1.1.4 DecryptData
(b)

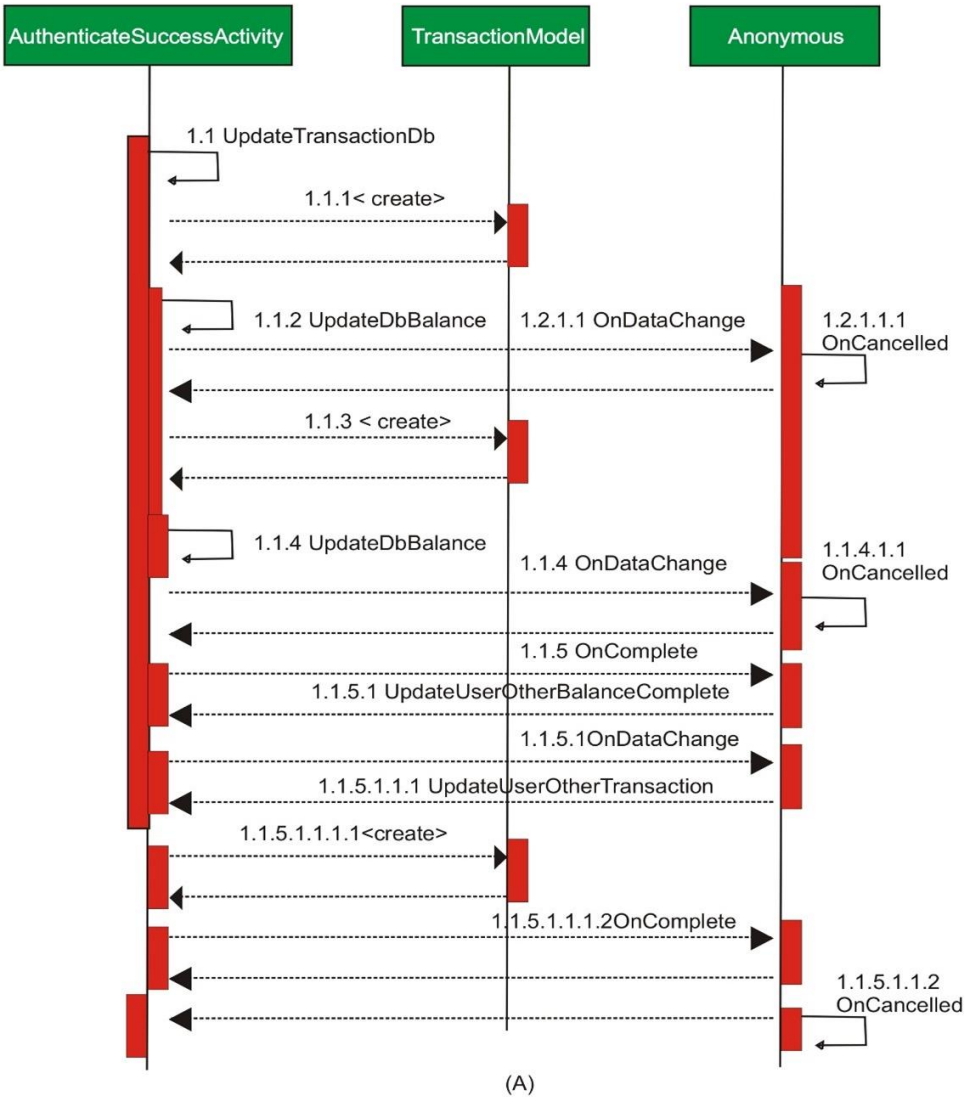


Figure 7(a). Authentication Task Using Face Recognition, (b) Verify OTP Task, and (c) Verification Successful Task of the Suggested Approach.

DOI: https://doi.org/10.52756/ijerr.2024.v45spl.012

error message is shown if there is a mismatch, and the user must revalidate with OTP authentication. If the data matches, the user is directed to their profile, where they can view updated transactions. The system updates the user's balance and displays the changes in real-time.

The authentication feature is broken into three phases: Authenticate Task, Authenticate OTP Task, and Authenticate Success Task. Figure 7a illustrates the sequence diagram for the Authenticate Task, Figure 7b shows the Authenticate OTP Task, and Figure 7c depicts the Authenticate Success Task.

In panel (a):

Phase 1.1: The Authenticate_Task hides the keyboard using Common_Util and returns to Authenticate_Task.

Phase 1.2: Authenticate_Task sends a request to Authentication_Model to start account creation and waits for a response.

Phase 1.3: Authentication_Model processes the request and sends a reply to Authenticate_Task.

In panel (b):

Phase 1.1: The Authenticate_Otp_Task hides the keyboard using Common_Util and returns a message to Authenticate_Otp_Task.

Phase 1.2: The verifyOtp process begins and concludes with Authenticate_Otp_Task, waiting for a response.

Phase **1.2.1:** After the OTP has been validated, the on_Complete operations await an anonymous response before proceeding.

Phase 1.2.1.1: Anonymous responds to Auth_User_Data, which is then sent back to Authenticate_Otp_Task.

Phase 1.2.1.1.1: Authenticate_Otp_Task sends an on_Data_Change request to anonymous.

Phase 1.2.1.1.1.1: Anonymous requests decrypt_Data from RSA_algorithm_Util.

Phase 1.2.1.1.1.1.1: The data entered by the user is decoded and forwarded to RSA_Util.

Phase 1.2.1.1.1.1.1.1: Data is decoded through the RSA module with the security key in RSA_Util, and a response message is sent from RSA_Algorithm_Util to an unidentified module.

Phase 1.2.1.1.1.2: Unidentified provides a reply to Authenticate_Otp_Task, which then navigates to the Success_Page.

Phase 1.2.1.1.1.3: Upon success, Authenticate_Otp_Task sends a request to the Payment_Gateway.

Phase 1.2.1.1.1.4: Finally, Authenticate_OTP_Task handles the on_Cancelled event.

In panel (c):

Phase 1.1**:** Update_Transaction_Db starts and sends a message to Authenticate_Success_Task.

Phase 1.1.1: Authenticate_Success_Task generates an amount in Transaction_Model and returns it to Authenticate_Success_Task.

Phase 1.1.2: The update_Db process begins and concludes with Authenticate_Success_Task, which awaits a reply.

Phase 1.1.2.1: Authenticate_Success_Task sends an on_Data_Change request to unidentified.

Phase 1.1.2.1.1: on_Cancelled begins and concludes with an unidentified module, which sends a request to Authenticate_Success_Task.

Phase 1.1.3: Authenticate_Success_Task creates an amount in Transaction_Model and returns it to Authenticate_Success_Task.

Phase 1.1.4: The update_Db process starts and ends with Authenticate_Success_Task, awaiting a response.

Phase 1.1.4.1: Authenticate_Success_Task sends an on_Data_Change request to an unidentified module.

Phase 1.1.4.1.1**:** on_Cancelled starts and ends with anonymous, which then sends a request to Authenticate_Success_Task.

Phase 1.1.5: The on_Complete request is sent to an unidentified module.

Phase 1.1.5.1: Unidentified sends Update_Other_User to Authenticate_Success_Task. The process then continues with the same flow.

Add Funds Feature

After authenticating the system, users can use various banks to top up their money. They must select the amount they wish to add and choose their bank account. The payment processing system is then directed to the user, with Payment Service Provider being the standard payment gateway for transactions. Registered users can deposit money to their Payment Service Provider account, and the Payment Service Provider facilitates this process smoothly. Once the transaction is completed, the

DOI: https://doi.org/10.52756/ijerr.2024.v45spl.012

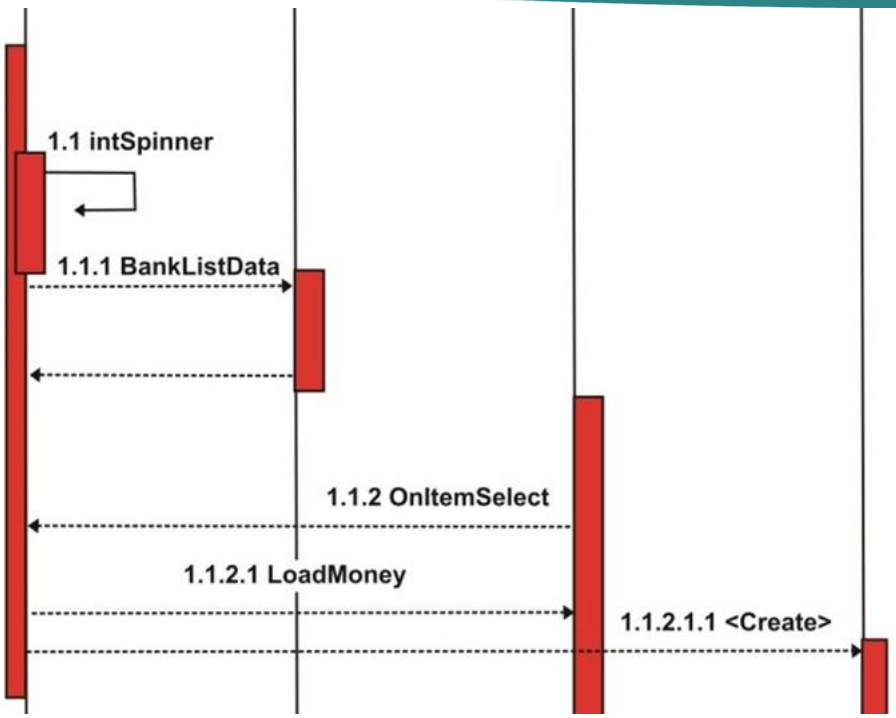


Figure 8. Add Funds Feature Task of the Suggested Approach.

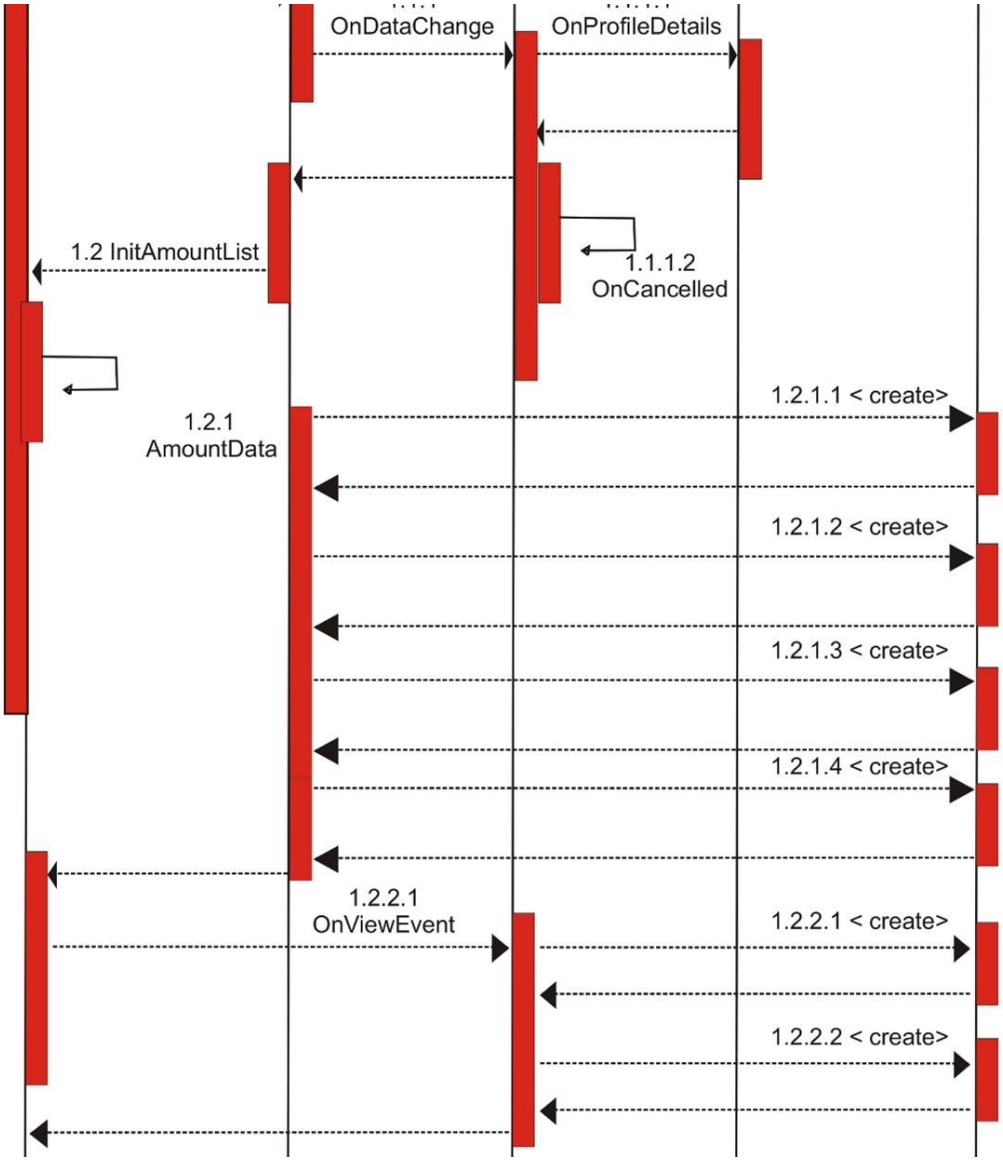


Figure 9. Money Transfer Feature of the Suggested Approach.

DOI: https://doi.org/10.52756/ijerr.2024.v45spl.012

user's balance is instantly refreshed in both the application and the database. Figure 8 illustrates the sequence diagram for the add fund function in the suggested INDFunds app.

Phase 1.1: The Init_Spinner process begins, and a message is sent to Add_Funds_Feature.

Phase 1.1.1: Add_Funds_Feature requests Bank_List_Data from Data_Util and waits for a response.

Phase 1.1.2: Add_Funds_Feature requests On_Item_Select from unidentified and awaits a reply.

Phase 1.1.2.1: Unidentified responds with load_Money to Add_Funds_Feature and waits for further requests.

Phase 1.1.2.1.1: Add_Funds_Feature sets an amount in Authenticate_Model and returns it to Add_Funds_Feature.

Money Transfer Feature

After adding funds to their account, users can send money to another registered INDFunds account. To start a transfer, users select the amount they wish to send and can retrieve the recipient's profile information from their contact list. Once the recipient is selected, the user must authenticate using their password, face recognition, and a one-time password (OTP). Upon successful authentication, the funds are transferred to the recipient's account. Figure 9 shows the fund transfer feature sequence diagram in the suggested INDFunds app.

Phase 1.1: Money_Transfer_Feature sends a request for get_Profile_Details to Data_Util and returns to reply.

Phase 1.1.1: Data_Util sends an on_Data_Change request to unidentified and waits for a reply.

Phase 1.1.1.1: Unidentified requests on_Data_Profile_Details from User_Details_Callback and sends the response back to unidentified.

Phase 1.2: The init_Amount_List process starts, and a message is sent to Money_Transfer_Feature.

Phase 1.2.1: Money_Transfer_Feature requests Amount_Data from Data_Util and awaits a reply.

Phase 1.2.1.1: Data_Util adds money in Money_Transfer_Feature and returns it to Transfer_Money_Task.

Phase 1.2.2: Money_Transfer_Feature sends a request for on_View_Event to an unidentified module and awaits a reply.

Phase 1.2.2.1: Unidentified processes the request initiated in Money_Transfer_Feature and sends a response back to Money_Transfer_Feature.

Discussion

Authentication is becoming increasingly vital in the modern digital landscape. Users today often rely on multifactor authentication multi-phase authentication to enhance security beyond traditional passwords. Despite ongoing concerns about user privacy, protection, ease of access, and reliability, multi-phase authentication remains a robust system that balances security and efficiency for those accessing sensitive information. Biometrics, in particular, is vital in multi-phase authentication, complementing conventional protective measures, including passwords, security tokens and PINs rather than replacing them.

Combining two or more verification techniques for digital payment apps can significantly improve security. This discussion emphasizes the main execution and protective features of the suggested identity based on the device INDFunds system. The system incorporates four primary authentication classifications: information (e.g., Authentication Code), hardware verification, biometric verification, and proprietorship. While each technique offers robust security individually, combining them enhances protection against various attacks.

This segment presents research findings from experiments conducted on the prototype, comparing the proposed authentication scheme with existing methods. Previous approaches have identified various levels of authentication, such as ownership and multifactor methods, but often fail in practical applications. Many frameworks, like multifactor authentication, remain theoretical and ineffective in real-world scenarios. Our approach utilizes a combination of device identifier and user authentication information, encompassing information, device verification, biometric verification, face recognition as a strategy for effective authentication. The evaluation of our method, shown in Table 9, compares the existing and proposed authentication schemes, with the latter offering an additional security feature.

Knowledge: (U = username), (Pas = password), (Pi = PIN), (Pa = pattern lock)

Ownership: (O = SMS OTP), (S = smart cards), (H = hardware token)

Biometrics: (Fi = face recognition), (Fa = facial recognition), (I = iris recognition)

Device ID: IMEI

DOI: https://doi.org/10.52756/ijerr.2024.v45spl.012

Table 2. Assessment of Verification Schemes Between the Suggested and Existing Frameworks.

Authentication Categories												
Authors	**Knowledge**		**Ownership**			**Biometrics**			**Device ID**			**Total Point**
	U	**Pas**	**Pi**	**Pat**	**O**	**S**	**H**	**Fi**	**Fa**	**I**	**IMEI**	
Harish et al., 2019						1			1			2
Vengatesan et al., 2020			1			1						2
Benli et al., 2017			1						1			2
Okpara and Bekaroo, 2017								1	1			2
L. Sharma and Mathuria, 2018	1		1						1			3
K. Tiwari, 2016									1			1
Gupta et al., 2020										1		1
Houngbo et al., 2019									1			1
Proposed INDwallet			1			1		1			1	4

Table 2 illustrates that INDFunds leads with four active points across authentication categories. The second scheme by Sharma and Mathuria (2018) has three points (Harish et al., 2020; Vengatesan et al., 2020; Benli, 2020). Each feature has two points in its authentication categories. The remaining authors have one point each for their authentication categories. Notably, some proposed frameworks are theoretical and lack practical applicability.

Outcomes: The outcomes of the "AI-Enhanced Secure Mobile Banking System Utilizing Multi-Factor Authentication" likely include several key advancements and practical results that demonstrate the effectiveness of the proposed system. These outcomes focus on improving security, user experience, and overall system performance in mobile banking.

Results

1. **Improved Security Metrics:** The results might show significant improvements in security metrics, such as a reduction in successful fraudulent transactions, lower false positive rates in fraud detection, and faster response times to potential security threats.
2. **Enhanced User Experience:** The study could demonstrate that the AI-enhanced multi-phase authentication system improves user satisfaction by reducing the frequency of unnecessary authentication Phases while maintaining high security.
3. **Performance Benchmarks:** The work might provide benchmarks comparing the proposed system to existing security solutions, highlighting improvements in both security and efficiency.
4. **Real-world Case Studies:** Results from real-world case studies or simulations could show the effectiveness of the system in various scenarios, such as different types of fraud attempts or user behaviors.
5. **Cost-Effectiveness:** The results may also indicate that the AI-enhanced system is cost-effective, offering a robust security solution without significantly increasing operational costs for banks.

Conclusions and Future Work

Security is a crucial aspect of e-wallet systems. The proposed system enhances user access through multifactor authentication, utilizing a combination of passwords, face recognition, and OTP to verify users. This approach significantly improves existing authentication methods by addressing four key authentication categories. The implementation and evaluation of the proposed INDFunds confirmed its stability and consistent performance, making it user-friendly even for those with limited mobile experience. It effectively prevents unauthorized data access during exchanges between user devices and database servers. Additionally, the proposed solution demonstrates a notable improvement in security

DOI: https://doi.org/10.52756/ijerr.2024.v45spl.012

performance compared to existing systems while maintaining a low effective cost. This cost efficiency is achieved because the INDFunds does not require additional face recognition or hardware devices. The study outlined a conceptual design for an authentication method centered on device identity in e-wallets. Future work could extend this method to iOS devices and explore additional features such as mobile recharges and utility payment functionalities. Furthermore, static and dynamic instruments will be employed to assess the protection of the suggested applications further. However, this study is limited by the fact that the proposed method cannot be used on multiple devices simultaneously and is not compatible with basic mobile phones lacking face recognition capabilities.

Conflict of interest

None

References

Ahmad, M. O., Tripathi, G., Siddiqui, F., Alam, M. A., Ahad, M. A., Akhtar, M. M., & Casalino, G. (2023). BAuth-ZKP—A blockchain-based multi-factor authentication mechanism for securing smart cities. *Sensors*, *23*(5), 2757. https://doi.org/10.3390/s23052757

Ali, G., Ally Dida, M., & Elikana Sam, A. (2020). Two-factor authentication scheme for mobile money: A review of threat models and countermeasures. *Future Internet*, *12*(10), 160. https://doi.org/10.3390/fi12100160

AliBabaee, A., & Broumandnia, A. (2018). Biometric authentication of fingerprint for banking users, using stream cipher algorithm. *Journal of Advances in Computer Research*, *9*(4), 1–17.

Alzu'bi, A., Albalas, F., Al-Hadhrami, T., Younis, L. B., & Bashayreh, A. (2021). Masked face recognition using deep learning: A review. *Electronics*, *10*(21), 2666. https://doi.org/10.3390/electronics10212666

Balani, V., Kharya, C., Shivhare, S. N., & Singh, T. P. (2024). An Enhanced RSA Algorithm to Counter Repetitive Ciphertext Threats Empowering User-centric Security. *Scalable Computing: Practice and Experience*, *25*(6), 4669–4682. https://doi.org/10.12694/scpe.v25i6.3386

Benli, M. (2020). External debt burden–economic growth nexus in Turkey. *Sosyal Bilimler Araştırma Dergisi*, *9*(1), 107–116.

Chaudhry, S. A., Farash, M. S., Naqvi, H., & Sher, M. (2016). A secure and efficient authenticated encryption for electronic payment systems using elliptic curve cryptography. *Electronic Commerce Research*, *16*, 113–139. https://doi.org/10.1007/s10660-015-9192-5

Dasgupta, D., Roy, A., & Nag, A. (2016). Toward the design of adaptive selection strategies for multi-factor authentication. *Computers & Security*, *63*, 85–116. https://doi.org/10.1016/j.cose.2016.09.004

Elliot, M., & Talent, K. (2018). A robust and scalable four factor authentication architecture to enhance security for mobile online transaction. *Int. J. Sci. Technol. Res.*, *7*(3), 139–143.

Harish, Vijayan, S., Mangold, N., & Bhardwaj, A. (2020). Water-Ice Exposing Scarps Within the Northern Midlatitude Craters on Mars. *Geophysical Research Letters*, *47*(14), e2020GL089057. https://doi.org/10.1029/2020GL089057

Houngbo, P. J., Hounsou, J. T., Damiani, E., Asal, R., Cimato, S., Frati, F., & Yeun, C. Y. (2019). Embedding a digital wallet to pay-with-a-selfie, from functional requirements to prototype. *Emerging Technologies for Developing Countries: Second EAI International Conference, AFRICATEK 2018, Cotonou, Benin, May 29–30, 2018, Proceedings 2*, 47–56. https://doi.org/10.1007/978-3-030-05198-3_4

Huseynov, E., & Seigneur, J.-M. (2017). Context-aware multifactor authentication survey. *Computer and Information Security Handbook*, 715–726. https://doi.org/10.1016/B978-0-12-803843-7.00050-8

Ibrahim, R. M. (2018). A review on online-banking security models, successes, and failures. *Proceedings of the 2018 International Conference on Electrical, Electronics, Computers, Communication, Mechanical and Computing (EECCMC), Tamil Nadu, India*, 28–29.

James, T. S., & Garnett, H. A. (2024). The determinants of electoral registration quality: A cross-national analysis. *Representation*, *60*(2), 279–302. https://doi.org/10.1080/00344893.2023.2207194

Kanimozhi, G., & Kamatchi, K. S. (2017). Security aspects of mobile based E wallet. *International Journal on Recent and Innovation Trends in Computing and Communication*, *5*(6), 1223–1228.

Khan, H. U., Sohail, M., Nazir, S., Hussain, T., Shah, B., & Ali, F. (2023). Role of

authentication factors in Fin-tech mobile transaction security. *Journal of Big Data*, *10*(1), 138. https://doi.org/10.1186/s40537-023-00807-3

Massaro, A., & Galiano, A. (2020). Image Processing and Post-Data Mining Processing for Security in Industrial Applications: Security in Industry. IGI Global, In *Handbook of Research on Intelligent Data Processing and Information Security Systems*, pp. 117–146. https://doi.org/10.4018/978-1-7998-1290-6.ch006

Mostafa, A. M., Ezz, M., Elbashir, M. K., Alruily, M., Hamouda, E., Alsarhani, M., & Said, W. (2023). Strengthening cloud security: an innovative multi-factor multi-layer authentication framework for cloud user authentication. *Applied Sciences*, *13*(19), 10871. https://doi.org/10.3390/app131910871

Nwabueze, E. E., Obioha, I., & Onuoha, O. (2017). Enhancing multi-factor authentication in modern computing. *Communications and Network*, *6*(03), 172. https://doi.org/10.4236/cn.2017.63012

Polas, M. R. H., Jahanshahi, A. A., Kabir, A. I., Sohel-Uz-Zaman, A. S. M., Osman, A. R., & Karim, R. (2022). Artificial intelligence, blockchain technology, and risk-taking behavior in the 4.0 IR Metaverse Era: evidence from Bangladesh-based SMEs. *Journal of Open Innovation: Technology, Market, and Complexity*, *8*(3), 168. https://doi.org/10.3390/joitmc8030168

Sharma, L., & Mathuria, M. (2018). Mobile banking transaction using fingerprint authentication. *2018 2nd International Conference on Inventive Systems and Control (ICISC)*, 1300–1305. https://doi.org/10.1109/ICISC.2018.8399016

Skračić, K., Pale, P., & Kostanjčar, Z. (2017). Authentication approach using one-time challenge generation based on user behavior patterns captured in transactional data sets. *Computers & Security*, *67*, 107–121. https://doi.org/10.1016/j.cose.2017.03.002

Tan, S. F., & Samsudin, A. (2018). Enhanced security of internet banking authentication with extended honey encryption (XHE) scheme. *Innovative Computing, Optimization and Its Applications: Modelling and Simulations*, 201–216. https://doi.org/10.1007/978-3-319-66984-7_12

Tellini, N., & Vargas, F. (2017). *Two-Factor Authentication: Selecting and implementing a two-factor authentication method for a digital assessment platform*.

Vengatesan, K., Kumar, A., Eknath, K. H., Samee, S., Vincent, R., & Ambeth Kumar, V. D. (2020). Intrusion detection framework using efficient spectral clustering technique. IOS Press, In *Intelligent Systems and Computer Technology*, pp. 98–103. https://doi.org/10.3233/APC200125

Vengatesan, K., Kumar, A., & Parthibhan, M. (2020). *Advanced Access Control Mechanism for Cloud Based E-Wallet*. Springer International Publishing: Berlin/Heidelberg, Germany.

Wang, C., Wang, Y., Chen, Y., Liu, H., & Liu, J. (2020). User authentication on mobile devices: Approaches, threats and trends. *Computer Networks*, *170*, 107118. https://doi.org/10.1016/j.comnet.2020.107118

Wang, Z., Zhang, X., Yu, P., Duan, W., Zhu, D., & Cao, N. (2020). A new face recognition method for intelligent security. *Applied Sciences*, *10*(3), 852. https://doi.org/10.3390/app10030852

How to cite this Article:

Mohd. Salman and Rahul Kumar Mishra (2024). AI-Enhanced Secure Mobile Banking System Utilizing Multi-Factor Authentication. *International Journal of Experimental Research and Review*, *44*, 153-172.

DOI : https://doi.org/10.52756/ijerr.2024.v45spl.012