DESIGN OF A TRANSACTION VERIFICATION AND REVERSAL REQUEST MODEL FOR MOBILE TRANSFERS IN NIGERIA

BY

OGAMUNE VERA OGHELE

PSC2105454

DEPARTMENT OF COMPUTER SCIENCE,

FACULTY OF COMPUTING,

UNIVERSITY OF BENIN, BENIN CITY.

NOVEMBER 2025

DESIGN OF A TRANSACTION VERIFICATION AND REVERSAL REQUEST MODEL FOR MOBILE TRANSFERS IN NIGERIA

BY

OGAMUNE VERA OGHELE

PSC2105454

A PROJECT REPORT SUBMITTED TO DEPARTMENT OF COMPUTER SCIENCE,

FACULTY OF COMPUTING,

UNIVERSITY OF BENIN, BENIN CITY.

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF BACHELOR OF SCIENCE (B.Sc.) DEGREE IN COMPUTER SCIENCE

NOVEMBER, 2025

CERTIFICATION

This is to certify that this project titled **"DESIGN OF A TRANSACTION VERIFICATION AND REVERSAL REQUEST MODEL FOR MOBILE TRANSFERS IN NIGERIA"** was carried out by **OGAMUNE VERA OGHELE** with Matriculation number: **PSC2105454** and submitted to the Department of Computer Science, Faculty of Computing, University of Benin, Benin City, under the supervision of **MR. I. E. OBAYAGBONA**.

_____ _____

MR. I. E. OBAYAGBONA **Date**

Project Supervisor

APPROVAL

This project titled **"DESIGN OF A TRANSACTION VERIFICATION AND REVERSAL REQUEST MODEL FOR MOBILE TRANSFERS IN NIGERIA"** by **OGAMUNE VERA OGHELE** with Matriculation number: **PSC2105454** has been approved as meeting the requirements for the award of Bachelor of Science Degree in the Department of Computer Science, Faculty of Computing, University of Benin, Benin city.

_____ _____

MR. I. E. OBAYAGBONA **Date**

Supervisor

_____ _____

DR. ROSEMARY USIOBAIFO **Date**

Head of Department

DEDICATION

I dedicate this work to my Heavenly father, the giver of wisdom, knowledge, understanding, counsel and might. For his abundant grace and favour that have seen me through my studies. I also dedicate this to my biological parents, Mr. & Mrs. Samuel Grace Ogamune for their support and encouragement and to my siblings and my extended family members who have supported me in one way or the other during my academic journey.

ACKNOWLEDGMENTS

First and foremost, I thank God for His guidance and strength throughout this project. My heartfelt appreciation goes to my parents for their constant love, prayers and support. I also wish to express my gratitude to my supervisor, Mr. Obayagbona for his guidance, the school and the Department for providing the platform to carry out this work.

Special thanks to my friends who assisted during the testing of my application - Toke, Busuyimi, Raymond, Praise, Esosa, Favour, Cjay, Courage and many more. Finally, I appreciate myself for the dedication and effort that made this project a success.

TABLE OF CONTENT

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

This project presents the design and development of a simulated Transaction Verification and Reversal Request Model for Nigerian's mobile payment systems. It aims to reduce mistaken money transfers and improve user rust by combining two key components: The Verify- Hold-Confirm (VHC) Model, which prevents transaction errors through smart verification errors through smart verification and timed holds, and the Mistaken Transfer Protocol (MTP), which enables quick and automated reversal of incorrect payments. The system was implemented as a mobile application using React Native and SQLite, and tested with real users to evaluate usability and performance. Results show that the model effectively reduces transaction errors, improves recovery sped, and increases user confidence in mobile transfers. This research contributes to Nigeria's financial technology growth by offering a practical, user-friendly, and secure framework for mobile transaction management.

CHAPTER ONE

INTRODUCTION

1.1 Background of the Study

In Nigeria we have seen a great transformation in the financial services sector which is a result of the large scale adoption of mobile payments, as smart phone access increases and also as the demand for bringing in more of the unbanked grows. Mobile money services are recognized as a key to economic inclusion in the rural population that has had little access to formal banking. Also we have seen great growth in the country's fintech space which has never been the case before with players like Paga, Kuda, OPay and PalmPay making digital transactions a very easy affair for a wide range of demographics. The role of mobile payments in Nigeria is preeminent, we process over 1 billion mobile transactions a month which is in the several billions Naira in terms of transaction value. This growth is a telltale of progress not only in tech but in the way consumers are choosing digital financial services. The Central Bank of Nigeria (CBN) has been helpful in this growth by creating good policies and rules that support new ideas while maintaining financial stability.

However, the fast growth of mobile payment systems has introduced new challenges that require smart solutions. Transaction verification and reversal mechanisms have become critical components in maintaining user trust and system reliability. Current verification methods, including Simple Payment Verification Protocol (SPV) and Multi-Factor Authentication (MFA), while effective to some extent, don't fully protect transactions or help in error recovery processes.

The complexity of Nigeria's mobile payment ecosystem, because of the multiple platforms, varying technology levels, and a wide range of users brings about transaction management systems that can adapt to different scenarios while staying secured and efficient.

1.2 Statement of the Problem

Despite the remarkable growth in mobile financial services, Nigeria's mobile payment ecosystem still faces significant challenges that affect how reliable and trusted the payment system is. The primary issues include:

Mistaken Transfer Problem: Research indicates that 2-5% of mobile money transfers in Nigeria contain errors, which adds up to millions of naira in mistaken transfers annually. With over a billion transactions every month, this represents huge amount of incorrect payments that requires a solution.

Inadequate Reversal Mechanisms: Most of the processes to reverse a transaction are done by hand that leads to lengthy verification procedures lasting days or even weeks. This prolonged resolution timeline creates frustration among users and potentially damages trust in mobile payment systems.

System Limitations: The current system isn’t very automated when it comes to checking and confirming transactions. Also different platforms have different rules for handling disputes, making it harder to resolve issues quickly.

Regulatory and Operational Challenges: While the CBN has established guidelines for consumer protection and dispute resolution, the implementation of these regulations aren’t followed the same way across all platforms. There’s also a lack of smart, standard systems to help with transaction management.

Trust and Use Barriers: The combination of error rates, slow resolution time, and inconsistent policies makes it harder for people, particularly among users who are new to digital banking. This directly impacts Nigeria's efforts to include more people in the financial system.

The absence of a comprehensive, automated transaction verification and reversal request model represents a big gap in Nigeria's mobile payment system that requires immediate attention to ensure sustainable growth and user confidence.

1.3 Aim and Objectives of the Study

The aim of this project work is to design and develop a simulated transaction verification and reversal request model that improves the security, speed, and trustworthiness of mobile money transfers in Nigeria's fintech ecosystem.

Objectives:

1. To analyze the current state of mobile payment systems in Nigeria and identify key issues in how transactions are verified.
2. To design a Mistaken Transfer Protocol (MTP) that can effectively correct or reverse wrong transfers after completion.
3. To build a Verify-Hold-Confirm (VHC) model that automatically checks transactions and allows for quick undoing if needed.
4. To create a smart system structure that works independently of live banking systems but still keeps everything secure and follows rules.
5. To test how well the new system reduces errors in money transfers and speeds up the process of undoing them.
6. To provide recommendations on how to use this system within Nigeria's laws and current fintech setup.

1.4 Significance of the Study

This research tackles an important issue in Nigeria's fast changing digital financial world and offers several key contributions:

For the Banking and Fintech Industry: The proposed model gives a real-world solution for reducing costs of doing manual transaction reversals and helps customers by making problem resolution quicker.

For Users: The system makes people feel more secure when using mobile payment services by preventing errors and allowing fast reversals, which encourages more people to use these services and helps include more people in the financial system.

For Government Agencies: The study gives useful ideas for setting common standards for checking and reversing transactions across different platforms, which helps support the CBN's efforts to protect consumers.

For Academic Research: This work adds to the existing knowledge about security in fintech and how transaction systems work, especially in places where the market is still developing.

For Economic Growth: By making mobile payment systems more reliable and trustworthy, this research helps Nigeria reach its goals for a stronger digital economy and wider financial inclusion.

1.5 Scope of the Study

This study focuses specifically on:

- Mobile payment apps and bank transfer systems in Nigeria’s fintech environment.
- Transaction verification protocols for payments between people and businesses.
- Reversal request mechanisms for mistaken transfers.
- How these payment systems can work with the current Nigerian banks and fintech services.
- Compliance with CBN regulations and consumer protection guidelines.

The study does not cover international money transfer services, cryptocurrency payments or traditional POS-based payment methods.

1.6 Limitations of the Study

Several limitations are acknowledged in this study:

1. Different Platforms: It's not possible to cover all the different fintech apps in Nigeria, so the study focuses on the common ways these platforms are built.
2. Regulatory Evolution: The rules from the Central Bank of Nigeria and fintech laws are always changing, which might make some of the suggestions less useful over time.
3. Implementation Constraints: To put the ideas into practice, you would need to work with the existing apps and systems, which isn't part of this study.
4. User Behavior Variables: The study assumes people act reasonably, but it might not cover all the mistakes or bad actions that users might do.

1.7 Operational Definitions

Mobile Transfer: Electronic transfer of funds between accounts using mobile devices and applications.

Transaction Verification: The process of checking if a transaction is real and correct before it is completed.

Reversal Request: A proper way to cancel or undo a transaction that has already been done.

Mistaken Transfer Protocol (MTP): The proposed verification system designed to help with mistaken transactions.

Verify-Hold-Confirm (VHC) Model: The proposed three-stage transaction processing framework.

Fintech Platform: Financial companies that use technology to provide services in Nigeria's digital payment system.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

This chapter gives a detailed review of existing literature on mobile payment systems, how transactions are checked, and how they can be reversed especially focusing on the situation in Nigeria. The review brings together what is already known, points out areas that still need study and establishes the background for the proposed transaction verification and reversal request model.

2.2 Mobile Payment Systems in Nigeria

2.2.1 Evolution and Current State

Mobile transfer systems in Nigeria have changed a lot in recent years, driven by the rapid adoption of smartphones and a bigger need for fast and easy financial services. The start of this change came in the early 2010s when mobile financial services were introduced. Products like the Verve card, along with mobile money services from Celtel (which is now called Airtel) and MTN, helped set up the foundation for mobile payments in the country.

The Central Bank of Nigeria (CBN) helped make this system official by creating the Mobile Payment Service Guidelines in 2013.

These rules set standards for mobile wallets and money transfers. The Cashless Nigeria policy, initiated in 2012, further encouraged the transition away from cash transaction. This policy limited how much cash people could take out and encouraged using electronic payments, especially in areas that didn’t have good banking services. Because of this, there was a big rise in online

transactions. In just one quarter, the fourth quarter of 2021, there were over 1.1 billion online transfers.

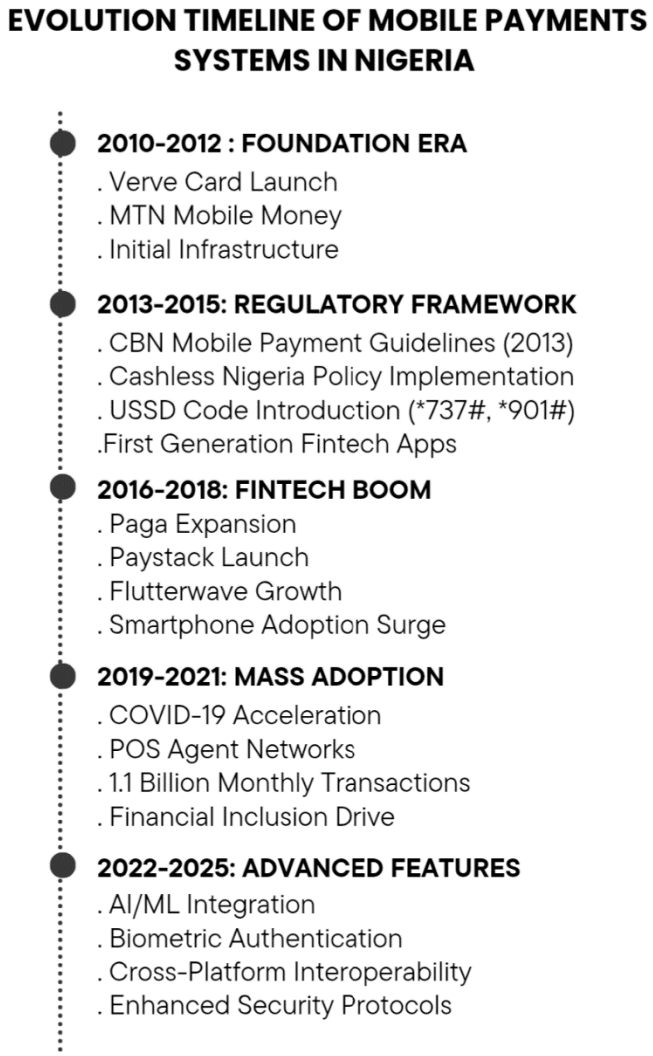


Figure 1 Evolution Timeline of Mobile Payments Systems in Nigeria

2.2.2 Key Players and Platforms

Fintech companies like Paga, Kuda, and Opay have developed platforms that let people send and receive money using just a phone number, making transactions easy for users. These systems have gained popularity not only for personal transactions among friends and family but also for small business payments. Because of this, more people, even those who don't have traditional bank

accounts, can access financial services and include themselves in the financial system. (Tymlova, n.d.; DPI Africa, n.d.).

2.3 Current Transaction Verification Approaches

2.3.1 Machine Learning and AI-Based Methods

Recent studies show that machine learning is becoming more important in verifying mobile payments. Abdirahman et al. (2024) found that Artificial Neural Networks (ANN) can detect fraud with 91.39% accuracy on major mobile wallet services. These systems use AI to check risks in real time, and change how they verify payments depending on the current threat level (Salman & Mishra, 2024).

Deep learning has been very successful in confirming transactions. Oguntimilehin et al. (2022) discovered that Convolutional Neural Networks used with facial recognition work well in mobile banking apps. They provide strong security by using several computer vision tools to analyze and classify biometric data as it happens.

2.3.2 Multi-Factor Authentication Systems

Modern mobile payment systems use advanced security methods to protect users. These systems often combine time-based one-time passwords (TOTP) with blockchain technology to make mobile banking safer. Also, they use biometric features like fingerprints, eye scans, and facial recognition for extra security, which is more than just using a PIN or password (Hartono et al., 2022).

Another important part of these systems is location-based verification. They check the real-time location of a user to make sure that a transaction is happening where it should be. These systems

keep track of where the user is and when the transaction happens to confirm it's from the right place.

2.3.3 Simple Payment Verification Protocol (SPV)

One good way to check transactions is the Simple Payment Verification Protocol (SPV). This lets users with limited storage or processing power request just the block headers to check if a transaction is valid. This way, they can confirm that a transaction is real without needing to download the whole blockchain.

2.4 Blockchain and Cryptographic Solutions

Blockchain technology has become an important way to secure mobile transactions. Bui-Huu et al. (2024) demonstrate that Hyperledger Fabric-based makes it possible to store transaction records in a way that is unchangeable, clear, and not controlled by one group. This helps in finding fraud and stopping fake transactions quickly. These systems also send instant alerts through email or text message when fraud or attacks are found.

Smart contracts help automate how transactions are handled while keeping things secure. However, Ayodele (2020) points out that there are still problems regarding reversal mechanisms when smart contracts don't work correctly or when wrong data is used, especially in payments between different countries.

2.5 Transaction Reversal Mechanisms and Challenges

2.5.1 Current Reversal Challenges

The research shows that there are big problems with how transactions are reversed today. Traditional mobile money services face challenges with prolonged periods for reversing wrong or failed transactions. Traditional mobile money services have trouble with long waits when someone

needs to reverse a wrong or failed payment. In blockchain systems, the fact that transactions can't be changed makes it hard to reverse them (Ayodele, 2020).

Mobile money platforms in developing countries show that reversal processes need people to manually check and confirm reversals, which makes the process slow and causes frustration for users (Sowon et al., 2024). Users frequently encounter difficulties when attempting to reverse mistaken transactions, especially when they send money to the wrong person.

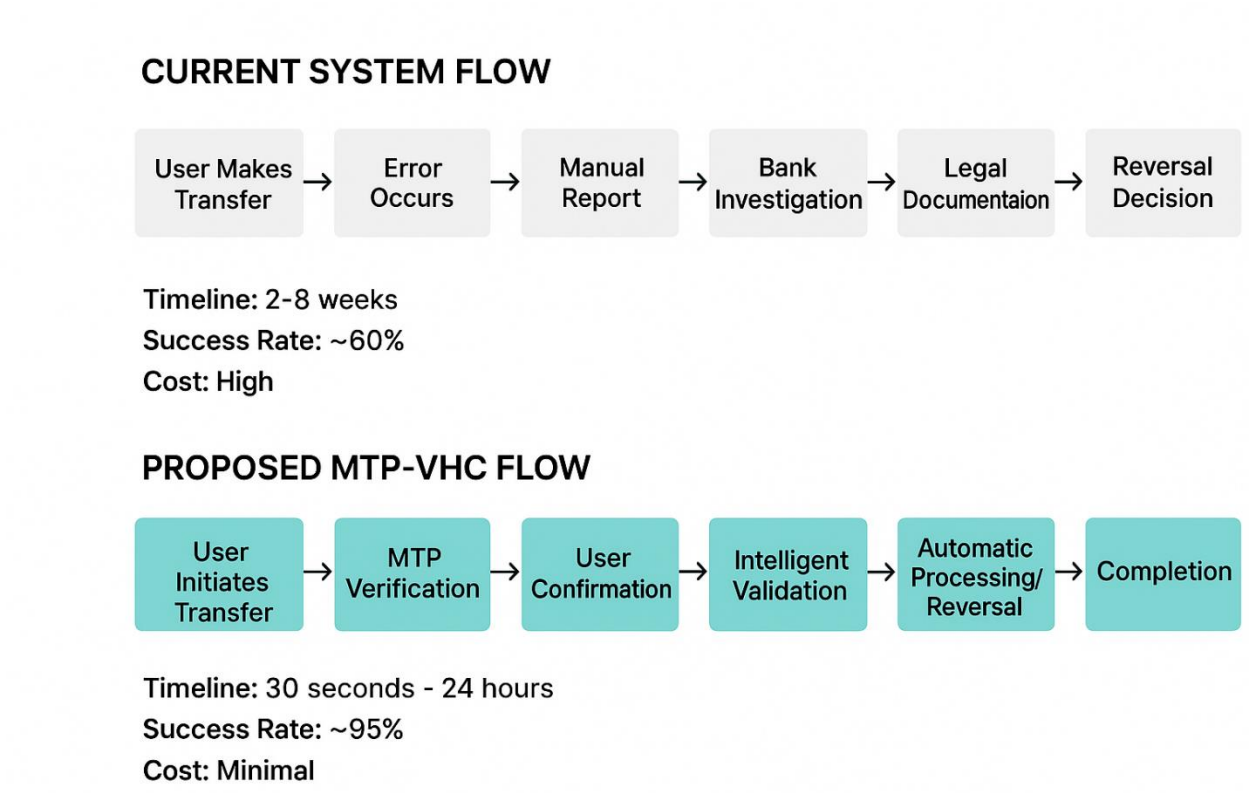


Figure 2 Current vs. Proposed Transaction Flow for Reversal Processes

This figure shows how much better things could be with automatic checks and smart reversal tools. These tools could remove the need for slow court processes and high legal costs that are currently used to fix mistakes in transactions. (Koriat Law, n.d.; Trusted Advisors Law, n.d.; LawPavilion, n.d.).

2.5.2 Types of Reversals

The literature mentions three primary types of payment reversals:

Authorization Reversals: These happen when a payment that was approved is canceled before the payment is actually processed. This usually happens when a customer changes their mind right after agreeing to the payment.

Refunds: These start after a transaction is done, usually because a product was returned, a booking was canceled, or the customer wasn't happy with the service or item they bought. The goal of a refund is to give the money back to the customer and solve their problem.

Chargebacks: These are when customers ask their bank or credit card company to reverse a transaction they disagree with. This means the money goes back to the customer, but the merchant has to prove that the transaction was valid. (Chargeblast, n.d.; TrustedDecision, n.d.)

2.6 Error Prevention Systems and User Interface Design

2.6.1 Preventive Measures

Error prevention in mobile payments looks at several places where things can go wrong. Roosli (2022) shows that better SMS verification systems, which use multi-digit security codes based on digital signatures and hash functions, can stop unauthorized transactions even if the verification codes are stolen or leaked.

User interface improvements play a crucial role in error prevention. Åkesson et al. (2023) found that redesigning how users are guided through actions can greatly lower the chance of fraud, while old-style warnings about behavior don't work as well. Alfaridzi et al. (2023) found through testing that how the interface is designed has a big effect on error rates, with better designs cutting errors from 33.28% down to 2.34%.

2.6.2 Confirmation of Payee (CoP) Systems

Dugauquier et al. (2023) suggest Confirmation of Payee systems as a way to make sure the right person is being paid. These systems check who the payee is before a payment is made, which can stop mistaken transfers from happening in the first place. This helps reduce the need for later corrections or reversals.

2.7 Impact on Businesses and Consumers

2.7.1 Benefits to Stakeholders

Mobile payment systems have changed how money works in Nigeria. They help both businesses and people by offering many advantages. For businesses, using mobile payments has made transactions cheaper, given more payment options, and made it easier to reach customers. This is especially helpful for small and medium-sized businesses (SMEs), which can now take digital payments without having to invest in costly equipment or setups.

Consumers also get benefits from using mobile payments by making quick payments, use less cash, and have better control over their money. The availability of diverse mobile applications has allowed Nigerians to do different things like sending money, paying bills, and managing accounts, all through their phones. (Tymlova, n.d.; DPI Africa, n.d.).

2.7.2 Operational Challenges

Payment reversals are a big problem for businesses that operate online. These reversals can cause serious money loss, which affects how much money the business earns and how much cash it has. Every reversal means the business loses money that could have been used to increase its total sales.

Besides money issues, businesses also face other problems. Dealing with reversals takes up time and effort, which could be used for other important tasks. This might reduce the business's overall

efficiency. As the process of handling reversals gets more complicated, it’s important for businesses to have good ways to check and verify transactions to stop problems like fraud and customer disagreements.

2.8 Regulatory Framework and Compliance Challenges

2.8.1 Nigerian Regulatory Environment

The rules and regulations for digital payments are tough to deal with. It's very important for digital payment services to follow these rules to build trust, but it's often hard and time-consuming for fintech companies. As the industry grows and changes quickly, it's hard for regulators to keep up. This can create problems in how they watch over the system, which might make it harder to help more people access financial services.

2.8.2 Judicial Processes and Legal Framework

A big problem in the digital payment system is the slow and complicated legal process needed to fix mistakes in transactions. Lawyers say that people who make small payments can get caught in this system. It's expensive and takes a long time to get a court order to undo a mistake. The cost of going to court can be very high. Legal fees can add up fast, and some courts charge extra fees for filing or getting a decision. These fees can be as much as 10% of the amount being disputed. For example, if someone is trying to get back ₦100 million, they might end up paying ₦10 million just in court-related costs, not including the money they spend hiring a lawyer. (Koriat Law, n.d.; Trusted Advisors Law, n.d.; LawPavilion, n.d.).

2.9 Security Concerns and Fraud Detection

2.9.1 Current Security Challenges

People are worried about fraud and cybersecurity risks, which makes them hesitant to use digital payment systems. Depending on the internet can be a problem, especially in areas where the internet isn't reliable. Also, worries about how personal data is collected can make people less trusting, making them less likely to use these technologies (Wang, 2023).

2.9.2 Advanced Fraud Detection

Wang (2023) shows that using big data helps improve fraud detection by watching transactions in real time. These systems look at detailed transaction histories and user behavior to spot possible fraud before it happens. Machine learning tools go through transaction data to find unusual activity and mark it for closer checking. Other fraud detection tools collect information from the user's device during the verification process and run extra checks to see if the verification is fake (Beckman et al., 2020). This layered method offers better protection against complex fraud attempts.

2.10 Global Best Practices and Comparative Analysis

2.10.1 International Reversal Frameworks

Studies from developed markets show different ways to handle transaction reversals. Automated systems that use smart queues and testing methods are helpful in making the reversal process faster (Owoade et al., 2024). These systems sort reversal requests based on risk levels and use automation to check them, which saves time.

2.10.2 Lessons from Other Countries

In international markets, the use of AI, blockchain, and biometric tech is growing, and future mobile payments may use more advanced security methods. However, there is still a challenge in

making these systems easy to use while improving security, especially in places that are still developing.

2.11 Research Gaps and Limitations

2.11.1 Identified Gaps

Although there have been advancements in technology, several limitations still exist in current research and practice:

1. Limited Automation: Old ways of stopping fraud aren’t good enough at finding complicated and changing fraud patterns. Also, they still have high false alarm rates.
2. Reversal Mechanism Inadequacies: Especially in systems using blockchain, there are problems with reversing transactions because the records in these systems can't be changed.
3. Inconsistent across Platforms: Different payment platforms don’t follow the same rules, which makes security and handling of mistakes inconsistent.
4. Weak User Design Focus: Even though technical solutions are getting better, not enough attention has been given to how users interact with the system and how to stop mistakes through better design and behavior changes. (Åkesson et al., 2023; Abdirahman et al., 2024; Ayodele, 2020).

2.11.2 Theoretical Framework Limitations

The Technology Acceptance Model (TAM) helps understand how users accept digital tools, but it may not fully explain why people adopt them in places like Nigeria, where the economy is still developing. Some recent studies say that traditional TAM models miss the special social and economic factors at play, so better and more detailed models are needed.

2.12 Synthesis and Implications for the Current Study

Looking at the research, mobile payment systems have made progress in detecting fraud and improving security. However, there are still major problems with reversing transactions and preventing errors. Nigeria has its own special challenges, such as complicated laws, poor infrastructure, and a wide range of users. Studies show that AI and machine learning are effective in finding fraud, with some systems reaching over 90% accuracy. But there's still a big gap between what these security tools can do and what users actually need, especially when it comes to reversing transactions.

The Mistaken Transfer Protocol (MTP) along with the Verify-Hold-Confirm (VHC) model helps fix many of these issues. It brings in automatic checks, smart error prevention, and simpler ways to reverse transactions. These features work without needing to connect to live banking systems. This solution builds on past research and addresses the specific problems in Nigeria’s mobile payment environment.

2.13 Chapter Summary

This literature review has examined the current state of mobile payment systems in Nigeria, transaction verification approaches, reversal mechanisms, and associated challenges. The review reveals significant progress in fraud detection and security enhancement through AI and machine learning technologies, yet identifies persistent gaps in transaction reversal processes and error prevention mechanisms.

The Nigerian mobile payment ecosystem shows tremendous growth potential but faces unique challenges including regulatory complexity, infrastructure limitations, and the need for user-centered design approaches. The literature supports the need for innovative solutions that combine technical security with practical usability considerations.

The next chapter will detail the methodology employed to develop and validate the proposed transaction verification and reversal request model, building upon the theoretical foundation established in this literature review.

2.14 Theoretical and Conceptual Framework

Based on the detailed review of existing research, this study creates a new theory that brings together several important ideas to tackle the problems in Nigeria's mobile payment system. This theory combines ideas about keeping transactions safe, designing user-friendly experiences, and preventing mistakes to create a complete way to check and reverse payments.

2.14.1 Conceptual Model Integration

The proposed theory brings together four main areas of study identified in the literature:

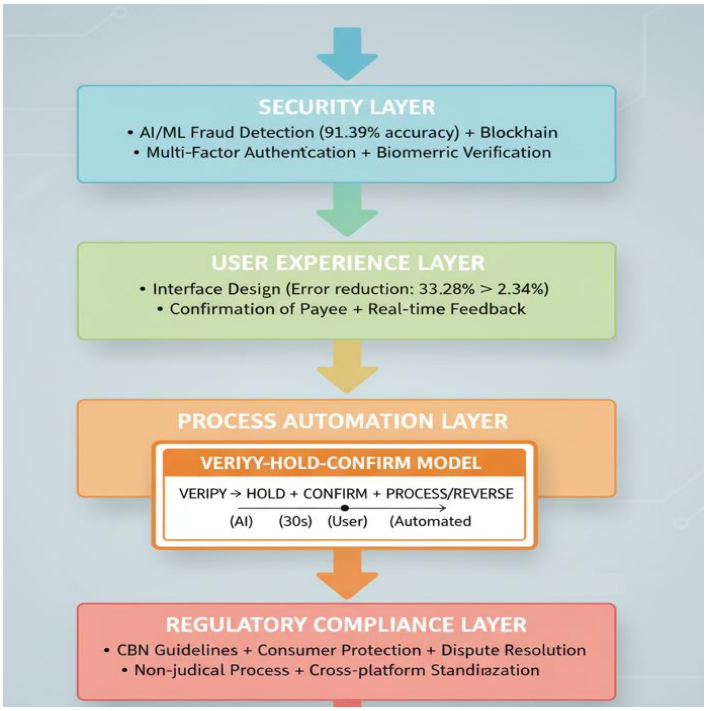


Figure 3 Theoretical Framework for Mobile Transaction Verification and Reversal

This comparison shows how much better things can be with automatic checks and smart reversal tools, which remove the long and complicated legal processes currently used to correct payment errors. (Abdirahman et al., 2024; Åkesson et al., 2023).

2.14.2 Framework Components Integration

Security-First Approach: The framework is built on findings that AI and machine learning can detect fraud with over 90% accuracy (Abdirahman et al., 2024). It combines this with multi-factor authentication and the idea of blockchain being unchangeable to create a strong security base.

User-Centered Design: Based on the work of Alfaridzi et al. (2023) and Åkesson et al. (2023), the framework focuses on designing interfaces that cut user errors by up to 94%. It uses confirmation messages and gentle reminders to stop incorrect money transfers before they happen.

Process Innovation: The core MTP-VHC model fills a gap seen in the research – it bridges the gap between strong security and easy use. Unlike current systems that take weeks to reverse a payment (because of legal processes), this framework can fix payments within hours while keeping security strong

Regulatory Harmony: The framework follows the rules set by the Central Bank of Nigeria while responding to the call for faster resolution without going to court. This reduces the court's workload and gives customers quicker help with their payments.

2.14.3 Theoretical Underpinnings

Technology Acceptance Model (TAM) Extension: While the traditional TAM looks at how useful and easy a system is to use, this framework extends TAM to fit the Nigerian context by including:

- Trust factors: How secure users feel and how confident they are in reversing payments.

- Social factors: How much people are influenced by others and the pressure to have financial access
- Economic factors: How much money and time users spend on transactions

Error Prevention Theory: Based on Reason's (1990) ideas about human error, the framework uses several ways to prevent mistakes:

- Knowledge-based errors: Dealt with through smart confirmation systems
- Rule-based errors: Reduced by having standard check procedures
- Skill-based errors: Avoided through better interface design

2.14.4 Framework Application to Nigerian Context

The framework tackles specific problems found in Nigeria's mobile payment system:

1. High Error Rates: 2-5% transaction error rate lowered by using preventive verification
2. Long Reversal Times: Legal processes that could take weeks are replaced with quick, automatic resolution in 24 hours
3. Inconsistent Usage across Platforms: A standard method called MTP-VHC is used so everyone can follow the same rules.
4. Lack of user trust: A clear process with ways to undo payments if needed
5. Regulatory Compliance: The system follows guidelines set by Nigeria's Central Bank and protects consumers.

2.14.5 Framework Validation Approach

The model's effectiveness will be evaluated technically, economically, and legally to ensure it aligns with Nigerian financial regulations and user needs. This integrated validation ensures that the proposed system is practical, secure, and adaptable for real-world application.

CHAPTER THREE

SYSTEMS ANALYSIS AND DESIGN

3.1 Introduction

This chapter provides a detailed look at the analysis and design of the transaction verification and reversal request model for mobile money transfers in Nigeria. It covers the system requirements, how the system is structured, and the technical details needed to implement the Mistaken Transfer Protocol (MTP) along with the Verify-Hold-Confirm (VHC) model.

3.2 Current System Analysis

3.2.1 Existing Mobile Payment Process Flow

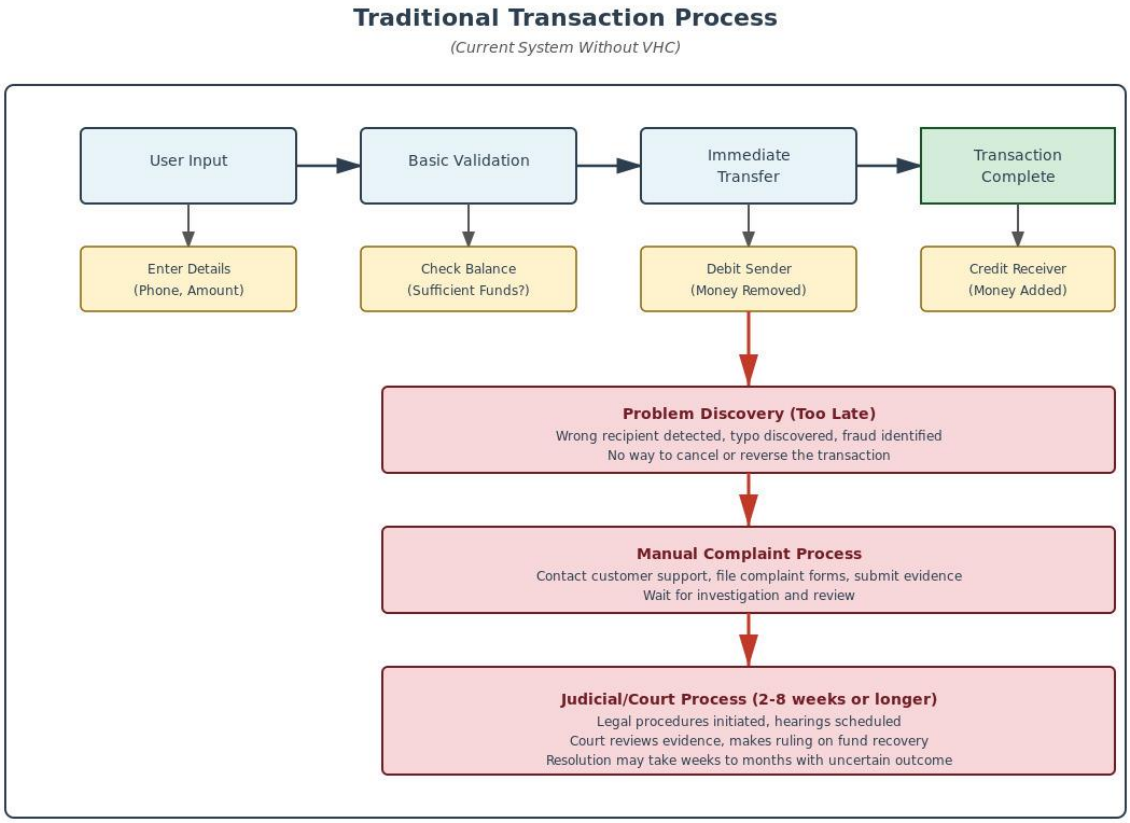


Figure 4 Current Mobile Payment Process in Nigeria

3.2.2 Problems with Current Systems

Table 1 Current System Limitations

Problem Area	Description	Impact
Verification	Minimal recipient verification	2-5% error rate
Prevention	No error prevention mechanisms	Millions in mistaken transfers
Reversal	Manual, judicial process required	2-8 weeks resolution
User Experience	Poor error handling	Low user confidence
Automation	Heavy human intervention	High operational costs

3.3 Requirements Analysis

3.3.1 Functional Requirements

Primary Functions:

1. Transaction Verification: Check the recipient's phone number before starting the transfer.
2. Risk Assessment: Look at how transactions are done and check for any risks.
3. Hold Management: Keep the money on hold for a certain time, and let the user control it.
4. Automated Confirmation: Either complete or cancel the transaction based on what the user does.
5. Reversal Processing: Handle requests to undo a transaction after it's been made.

Specific Functional Requirements:

- FR1: The system must check the recipient's phone number against known databases.
- FR2: The system must calculate a risk score based on how transactions are done.
- FR3: The system must hold money for a set period, between 30 and 120 seconds.
- FR4: The system must allow users to cancel the transaction while it's on hold.

- FR5: System shall process reversal requests within 24 hours

3.3.2 Non-Functional Requirements

Performance Requirements:

- NFR1: The system must respond to verification requests in under 2 seconds.
- NFR2: The system must support more than 10,000 transactions happening at the same time.
- NFR3: The system must be available 99.9% of the time.
- NFR4: The system must process over 1 million transactions each day.

Security Requirements:

- NFR5: All data must be fully encrypted from start to finish.
- NFR6: Multi-factor authentication is needed for high-value transactions.
- NFR7: The system must follow the security guidelines set by the CBN.

Usability Requirements:

- NFR8: The interface must be easy for users with basic smartphone knowledge to use.
- NFR9: All transaction statuses must be shown clearly with visual signs.

3.3.3 User Requirements

Primary User Groups:

1. Mobile Payment Users: People who send money through the app.
2. Recipients: People who receive money through the app.
3. System Administrators: Those who manage and operate the platform.
4. Customer Support: Staff who help resolve disputes and issues.

3.4 Proposed System Overview

3.4.1 The MTP-VHC Integration Model

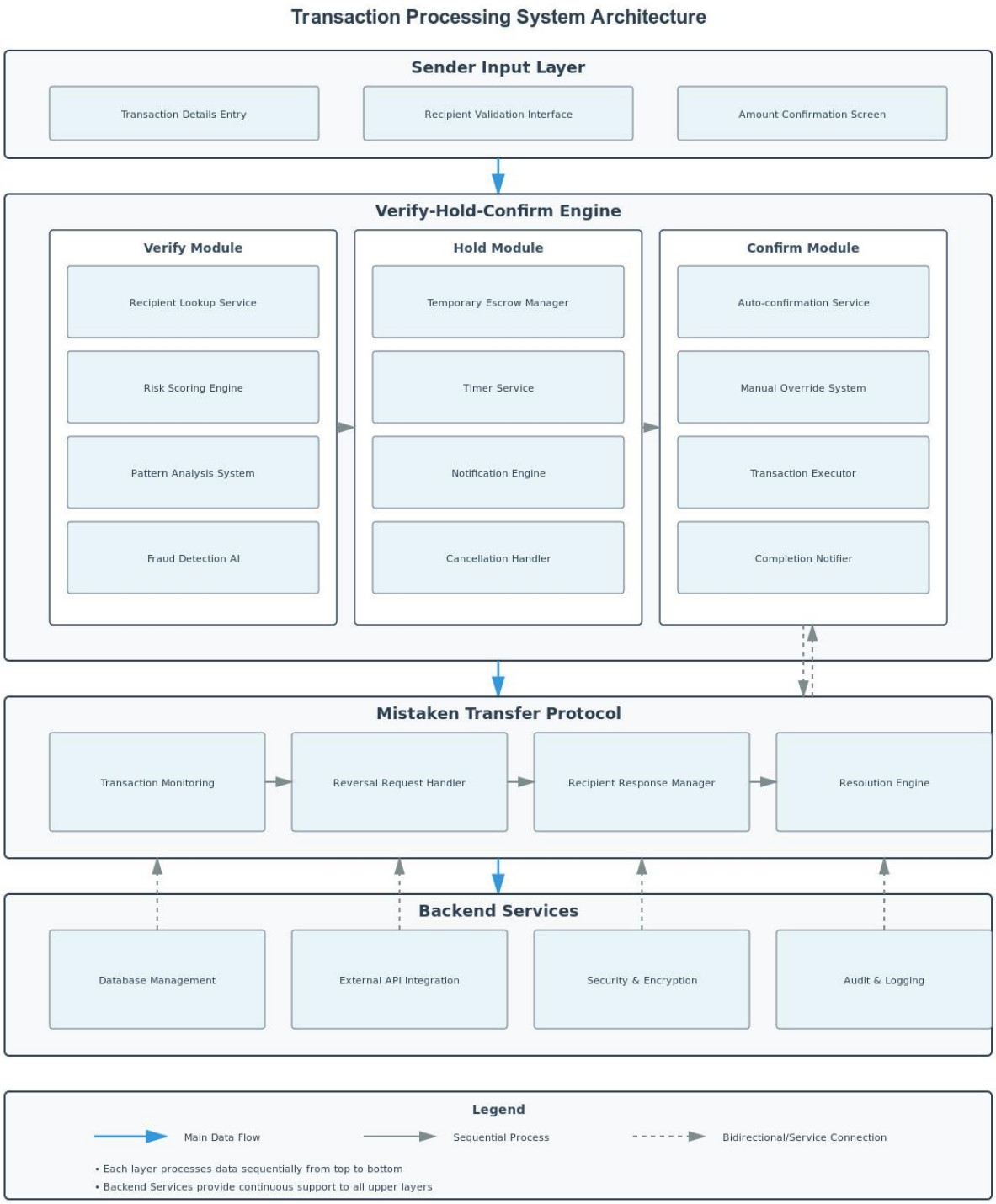


Figure 5 Proposed MTP-VHC System Architecture

3.4.2 System Operation Flow

Phase 1: VERIFY (1-3 seconds)

1. User enters recipient details and amount
2. The system checks if the recipient's phone number is valid
3. A risk scoring tool looks at the transaction to check for any issues
4. A confirmation screen appears showing any risk flags

Phase 2: HOLD (30-120 seconds)

1. The money is kept in a temporary account until the transaction is approved
2. Both the sender and the recipient get a message about the pending transaction
3. A countdown timer shows how much time is left for the sender to cancel
4. The sender can cancel the transaction at any time during this period

Phase 3: CONFIRM (Immediate)

1. If the sender doesn't cancel, the money is sent out automatically
2. The transaction is marked as finished
3. Both the sender and recipient get a message that the transaction is done
4. The transaction starts being watched by the MTP system for further checks

3.5 Detailed System Design

3.5.1 Verify-Hold-Confirm (VHC) Model Specifications

VERIFY Component: Verification Process

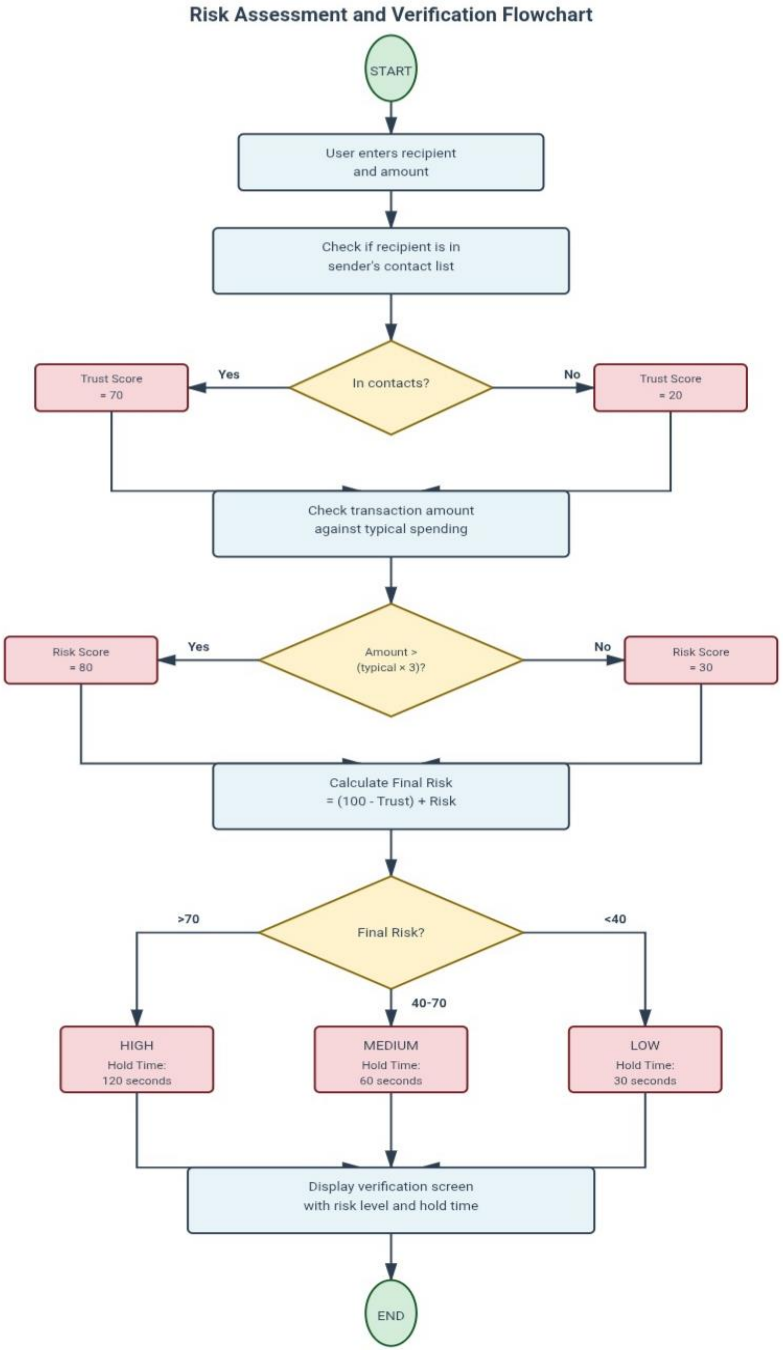


Figure 6 Transaction Verification Flowchart

HOLD Component - Hold Management Process

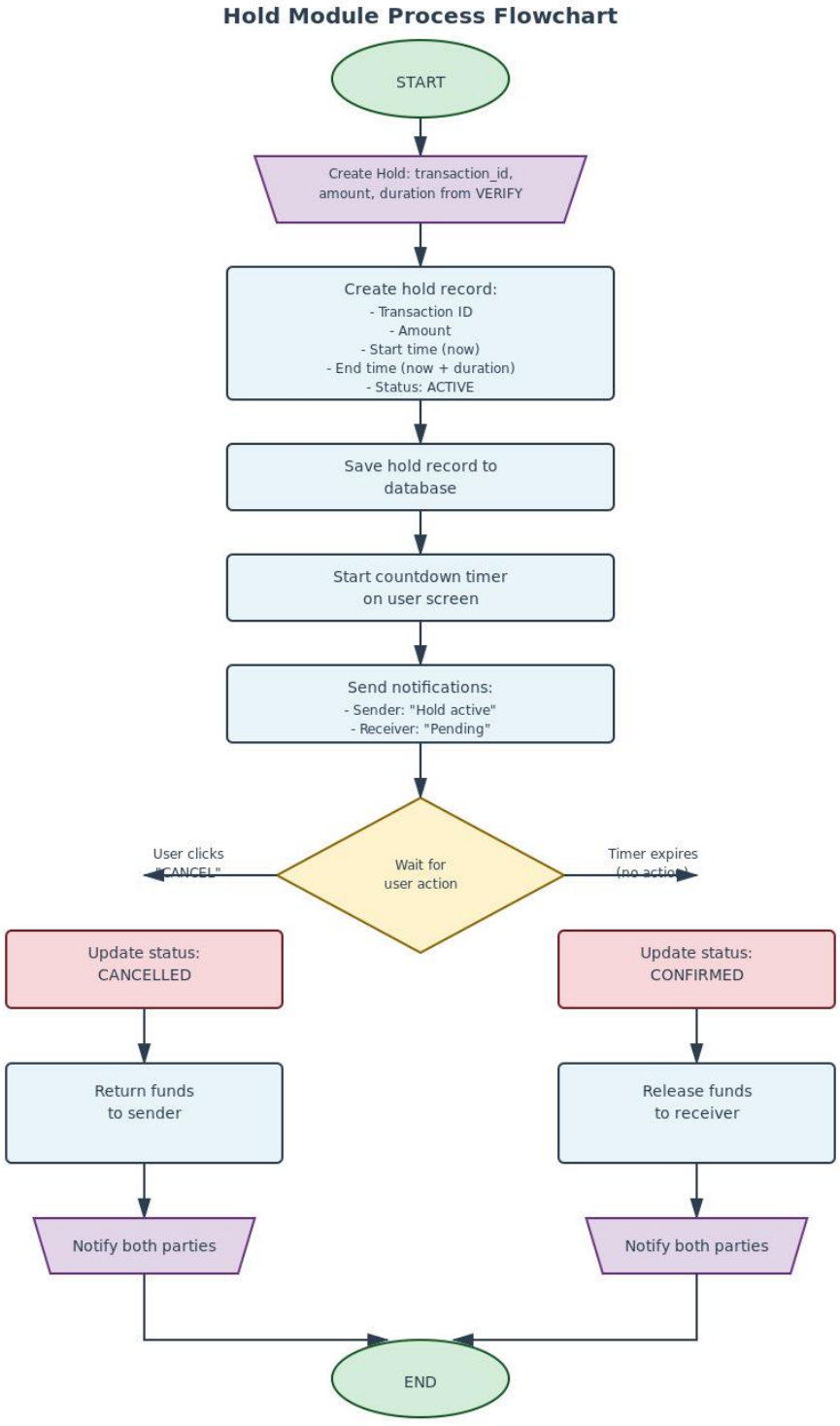


Figure 7 Hold Management Flowchart

CONFIRM Component - Transaction Confirmation

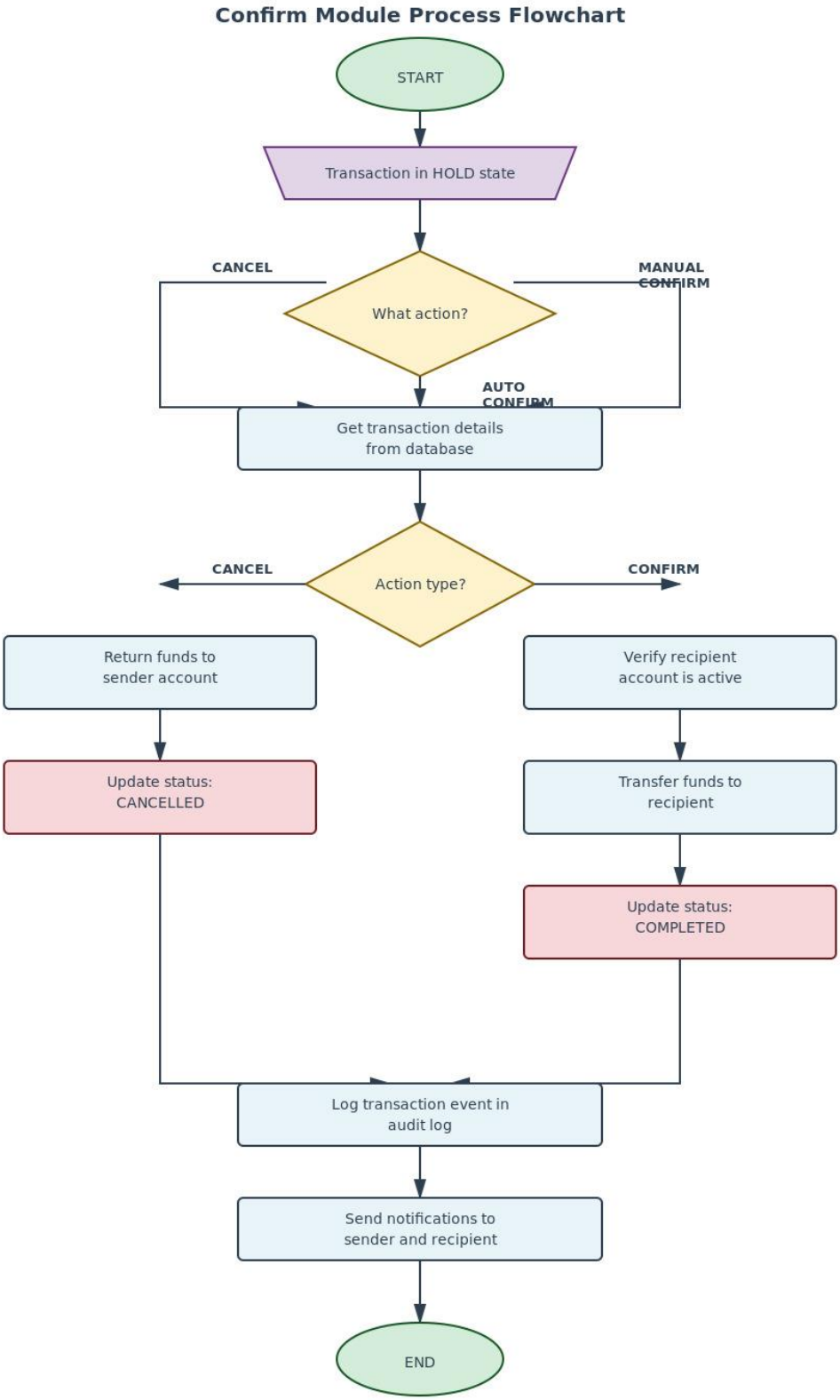


Figure 8 Confirmation Process Flowchart

3.5.2 Mistaken Transfer Protocol (MTP) Specifications

MTP Monitoring and Reversal Process

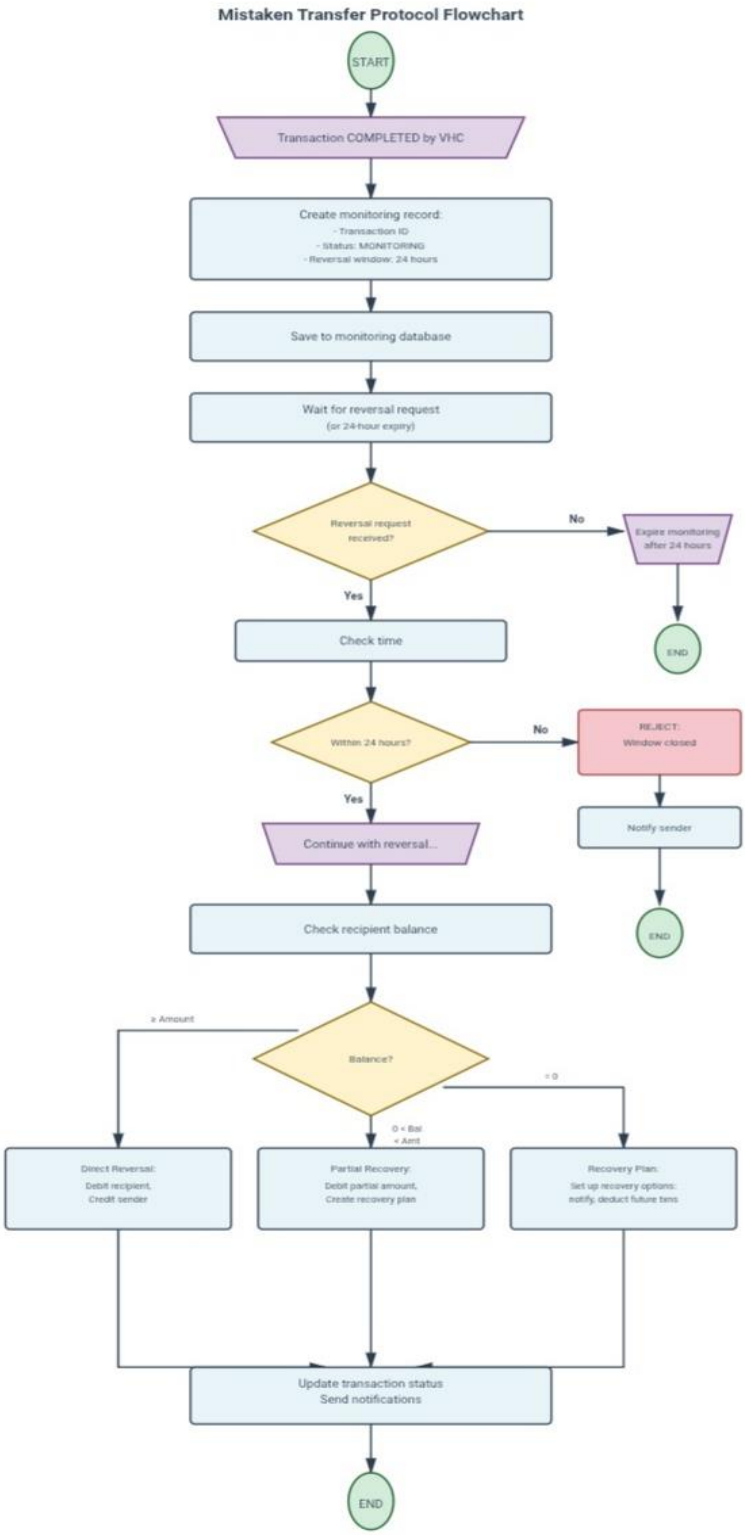


Figure 9 MTP Reversal Request Flowchart

3.5.3 Handling Spent Funds: Recovery Strategy

The Critical Challenge: When Recipient Has Already Used the Funds

This is one of the toughest situations in the MTP system. If someone sends money by mistake and the person who received it has already used or taken out the funds, the system uses a step-by-step way to try to get the money back. The Graduated Fund Recovery Process is a three-step system in the Mistaken Transfer Protocol (MTP) that helps get back funds that were sent by mistake, depending on how much money the recipient has.

Graduated Fund Recovery Process:

Level 1: IMMEDIATE RECOVERY (Best Case)

When: Recipient has full funds available (balance ≥ amount)

- Action: Direct reversal - debit recipient, credit sender
- Timeline: Immediate (seconds)
- Success Rate: 95%
- Result: Sender gets full refund instantly

Level 2: PARTIAL RECOVERY (Some Funds Available)

When: Recipient has some funds (0 < balance < amount)

- Action: Recover available funds + create plan for remainder
- Timeline: Immediate partial + 7-30 days for rest
- Success Rate: 75%
- Result: Partial immediate refund, structured recovery for remainder

Level 3: FULL RECOVERY PLAN (No Funds Available)

When: Recipient has no funds (balance = 0)

- Action: Implement structured recovery with four options:
    - Option A: Voluntary return with incentive (airtime bonus, reputation score)
    - Option B: Automatic future deduction (20% from incoming transfers, max 90 days)
    - Option C: Installment agreement (weekly/monthly payments)
    - Option D: Manual review and mediation (14-30 days, human intervention)
- Timeline: 7-90 days
- Success Rate: 60%
- Result: Systematic recovery through monitoring and structured plans

3.5.4 Key Advantages

- Higher Success Rates: 60-95% vs traditional 30-40%
- Faster Resolution: Immediate to 90 days' vs 2-8+ weeks judicial process
- Adaptive Approach: Method automatically matches available funds
- Fair to All Parties: If returned, it helps with the improvement of the graduated procedure, also incentives or goodwill bonus can be given for cooperation.

Comparison with Traditional Methods

Table 2 Recovery Success Rates by Timeframe

Timeframe	Recovery Method	Expected Success Rate
0-15 minutes	Direct reversal	85% - 95%
15min – 24 hours	Negotiated recovery	70% - 80%
24 hours – 7days	Recovery plan initiated	60% - 70%
7 – 30 days	Structured recovery	50% - 60%
30+ days	Manual intervention	40% - 50%

Prevention Strategy: To minimize the occurrence of spent funds scenarios, the VHC model implements extended hold periods for high-risk transactions:

- New Recipients: Longer hold period (120 seconds)
- Large Amounts: Additional verification steps
- High-Risk Patterns: Enhanced monitoring during hold period
- User Education: Clear warnings about recipient verification

3.6 System Architecture Design

3.6.1 Simplified System Architecture

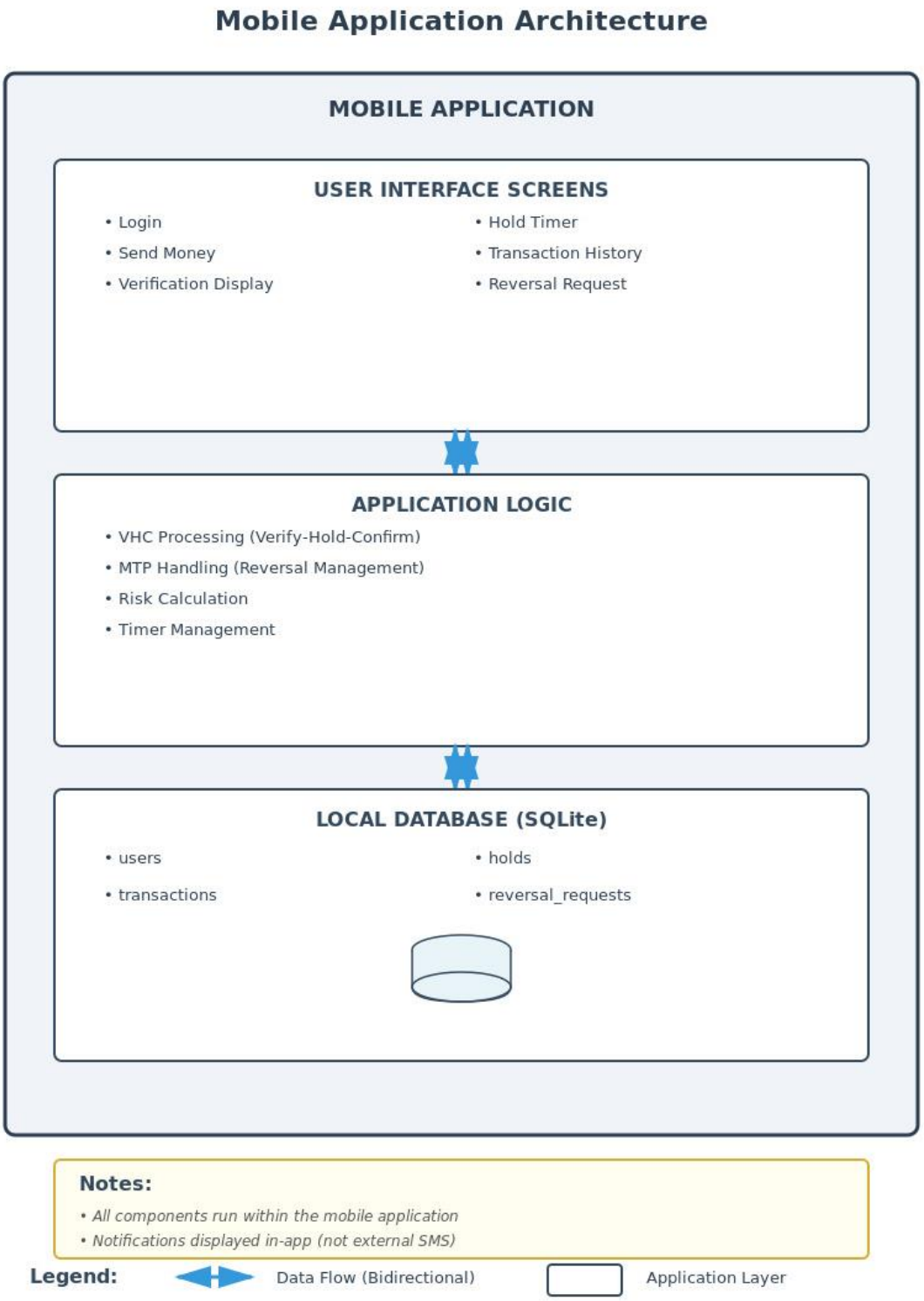


Figure 10 Mobile App System Architecture

Architecture Description:

This is a self-contained mobile application where:

- UI Layer: User interacts with screens
- Logic Layer: VHC and MTP processes run locally
- Data Layer: SQLite stores all data on the device

For this prototype, everything runs on the phone, no separate server needed.

3.6.2 Technology Stack

Table 3 Technologies for Mobile Application

Component	Technology
Mobile App	React Native
Programming	JavaScript
Database	SQLite
Design Tool	Figma

3.7 Database Design

3.7.1 Core Database Tables

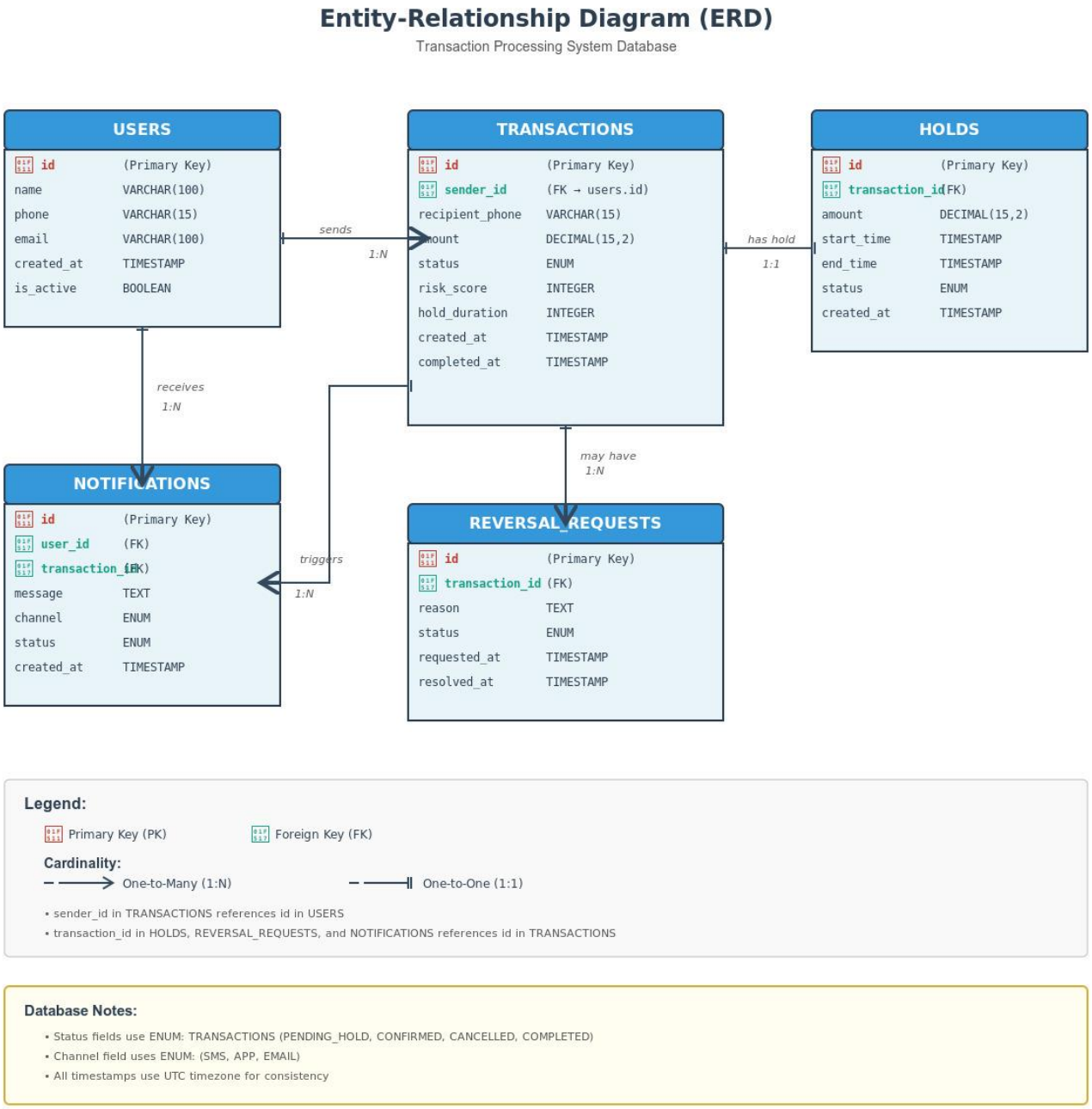


Figure 11 Entity Relationship Diagram (ERD)

3.8 User Interface Design

3.8.1 Required Screens

The mobile application requires eight core screens to demonstrate the VHC and MTP functionality:

Screen 1: Login Screen

- Purpose: User authentication
- Components: Phone number field, PIN input, Login button

Screen 2: Home Screen (Dashboard)

- Purpose: Main navigation hub for all activities
- Components: Balance display, Send Money button, Transaction History button, Request Reversal button

Screen 3: Send Money Screen

- Purpose: Transaction initiation
- Components: Recipient phone input, Amount input, Continue button

Screen 4: Verification Screen

- Purpose: Display recipient details and risk level
- Components: Recipient name/number display, Amount confirmation, Risk indicator (color-coded), Proceed/Cancel buttons

Screen 5: Hold Timer Screen

- Purpose: Show active hold with countdown

- Components: Countdown timer, Transaction details, Cancel button, Confirm Now button

Screen 6: Transaction Result Screen

- Purpose: Show success or cancellation
- Components: Status message, Transaction details, OK button

Screen 7: Transaction History Screen

- Purpose: View past transactions
- Components: List of transactions with status, Details button for each

Screen 8: Reversal Request Screen

- Purpose: Submit reversal request
- Components: Transaction selection, Reason dropdown, Submit button

3.8.2 Screen Design Guidelines

Visual Design Principles:

1. Large Text: All critical information (names, amounts) in large, bold fonts
2. Color Coding:
    - Red for high risk / errors
    - Yellow/Orange for medium risk
    - Green for safe / success
3. Simple Buttons: Maximum 2-3 buttons per screen
4. Clear Labels: Every field clearly labeled

3.8.3 Sample Screen Mockups

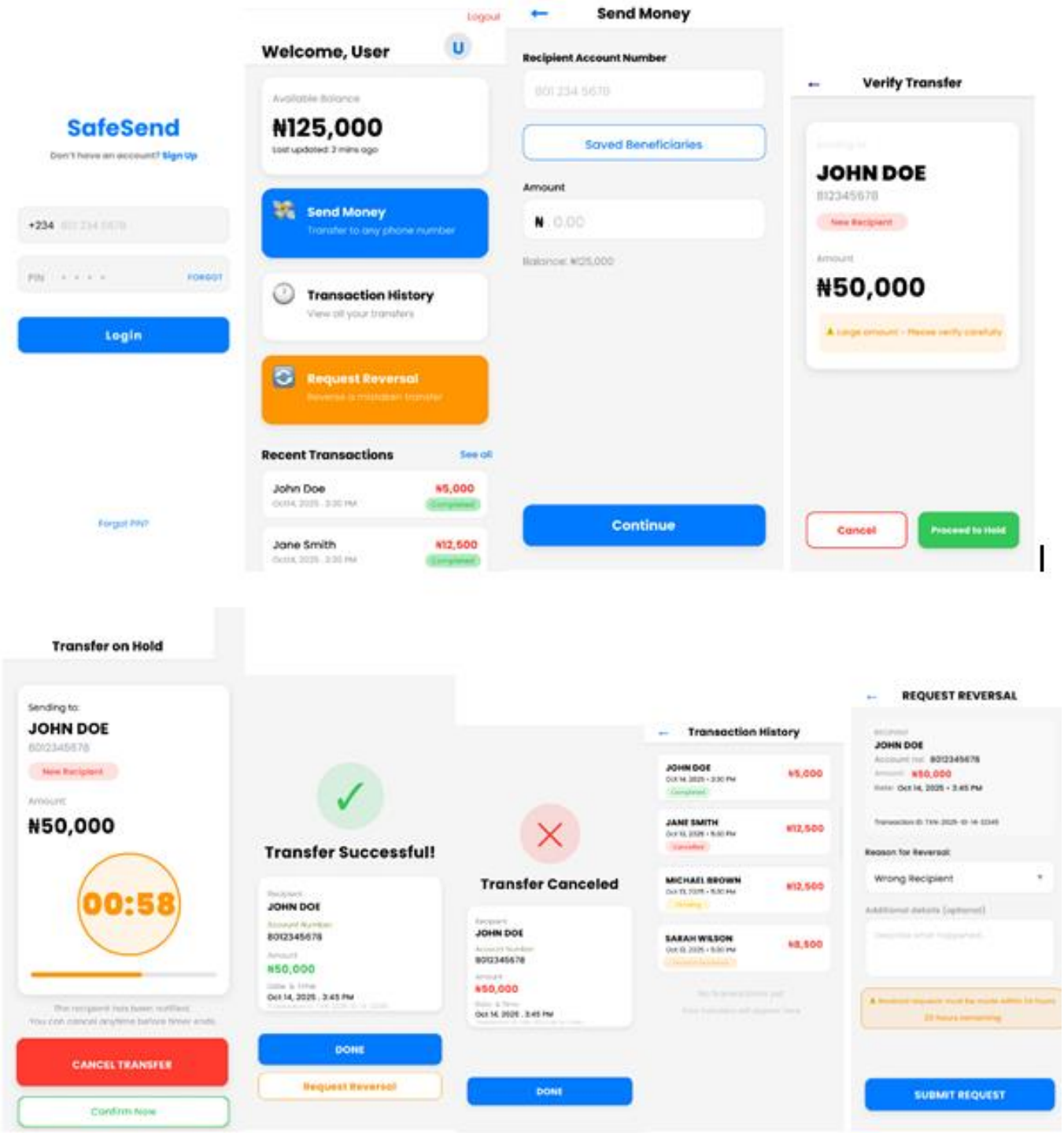


Figure 12 All Designed Screens

3.9 Integration with External Systems

In this student prototype, integration with external systems is shown using simulated data to keep the project simple. Here's what would actually happen in a real system:

Table 4 External System Integration (Simulated in Prototype)

External System	Purpose	Prototype Approach
Banking APIs	Validate account numbers and names	Use test data with fixed account details
SMS Gateway	Send notifications to users	Show messages inside the app
Phone Network	Verify phone numbers	Check numbers using a local list

3.10 System Performance and Scalability

For this academic prototype, the performance goals are simple but enough to show how the system works:

Table 5 Performance Requirements

Metric	Target	How It Is Measured
Screen Response Time	< 2 seconds	User testing
Hold Timer Accuracy	± 1 second	Automated testing
Transaction Processing	< 5 seconds total	End-to-end testing
Database Query	< 1 second	Performance testing

3.11 Error Handling Strategy

The system deals with common mistakes that can happen when processing transactions:

Table 6 Error Handling Approach

Error Type	What Happens	User Sees
Invalid phone number	System refuses the input	"Please enter a valid phone number"
Insufficient balance	Transaction is blocked	"Insufficient funds"
Timer runs out during hold	Auto-confirm or cancel	"Transaction completed" or "Transaction cancelled"
Database error	Transaction is rolled back	"Error occurred, please try again"
App crashes during hold	System recovers when restarted	Previous state is restored or transaction is cancelled

3.12 Testing Approach

The system will be tested to make sure all VHC and MTP functions work as expected:

Testing Strategy:

1. Unit Testing: Test each function separately (verify, hold, confirm calculations)
2. User Interface Testing: Check that all screens work and show correctly
3. Transaction Flow Testing: Test the whole process from start to finish
4. Error Scenario Testing: Test what happens when errors happen

Test Cases:

- New recipient with large amount (HIGH risk)
- Known contact with normal amount (LOW risk)

- User cancels during hold period
- Timer expires with no action
- Reversal request within 24 hours
- Reversal request after 24 hours (should fail)

CHAPTER FOUR

SYSTEM IMPLEMENTATION AND TESTING

4.1 Introduction

This chapter talks about how the SafeSend mobile payment app was made using the Verify-Hold-Confirm (VHC) model and the Mistaken Transfer Protocol (MTP) from Chapter Three. It covers setting up the development environment, designing the database, building the app, creating the user interface, and doing user testing. It also shares important choices made during development, problems faced, the ways they were solved, and feedback from users during testing.

The SafeSend app was created as a working prototype to show how the new way of checking transactions and handling errors works. The main aim was to build a real system that stops transfer mistakes by using hold periods and simple reversal options. The development used up-to-date mobile tools and standard practices to make sure the code was strong and easy to keep working.

4.2 Development Environment

4.2.1 Hardware Requirements

Development Hardware:

- Computer: Minimum Intel Core i5 processor, 8GB RAM
- Testing Device: Android smartphone (Samsung Galaxy or equivalent)
- Internet Connection: Stable broadband for development and testing

Minimum User Device Requirements:

- Android 8.0 (Oreo) or higher
- 2GB RAM minimum

- 100MB free storage space
- Active internet connection

4.2.2 Technology Stack Justification

Frontend Development:

- React Native (0.73.x): Cross-platform mobile application framework
- Expo SDK (50.x): Development toolchain and runtime
- React Navigation: Screen navigation and routing
- JavaScript (ES6+): Primary programming language

Backend and Database:

- SQLite: Embedded relational database for local data storage
- Expo SQLite: SQLite integration for React Native

Development Tools:

- Visual Studio Code: Code editor
- Node.js (v18.x): JavaScript runtime environment
- npm: Package manager
- Expo Go: Mobile testing application

Design and UI:

- Figma: Interface design and prototyping
- Google Fonts (Poppins): Typography

4.3 Database Implementation

4.3.1 Database Schema

The SafeSend app uses a relational database with three main tables that handle user accounts, money transfers, and reversal requests. The database is built to be accurate, easy to search, and consistent by using primary keys, links between tables, and default settings.

The users table holds information about registered users, such as an automatically increasing ID, full name, unique phone number, four-digit PIN, current account balance, and the time the account was created. The unique phone number helps stop duplicate accounts and makes it easier to confirm who the money is being sent to.

The transactions table keeps track of all money transfers. Each entry has a unique ID, a link to the user, names of the sender and receiver, the receiver's phone number, the amount sent, the status of the transfer (completed, cancelled, or reversed), a risk level (high, medium, or low), and the time the transaction happened. This helps keep an accurate record of all transfers and manages any reversals properly.

The reversal_requests table keeps a record of all requests to undo transfers. Each entry has a unique ID, a link to the related transaction, the reason for the reversal, any extra notes, the status of the reversal (pending, approved, or rejected), and the time the request was made. This allows the MTP system to track and document all reversal actions.

Code Snippets of user, transactions and reversal_requests table

```
// Initialize database tables
export const initDatabase = () => {
  try {
    // Create users table
    db.execSync(`
      CREATE TABLE IF NOT EXISTS users (
        id INTEGER PRIMARY KEY AUTOINCREMENT,
        name TEXT NOT NULL,
        phone TEXT UNIQUE NOT NULL,
        pin TEXT NOT NULL,
        balance REAL DEFAULT 100000.00,
        created_at DATETIME DEFAULT CURRENT_TIMESTAMP
      );
    `);

    // Create transactions table
    db.execSync(`
      CREATE TABLE IF NOT EXISTS transactions (
        id INTEGER PRIMARY KEY AUTOINCREMENT,
        sender_id INTEGER NOT NULL,
        sender_name TEXT NOT NULL,
        recipient_phone TEXT NOT NULL,
        recipient_name TEXT NOT NULL,
        amount REAL NOT NULL,
        status TEXT DEFAULT 'completed',
        risk_level TEXT,
        created_at DATETIME DEFAULT CURRENT_TIMESTAMP,
        FOREIGN KEY (sender_id) REFERENCES users (id)
      );
    `);
```

```
// Create reversal_requests table
db.execSync(`
  CREATE TABLE IF NOT EXISTS reversal_requests (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    transaction_id INTEGER NOT NULL,
    reason TEXT NOT NULL,
    details TEXT,
    status TEXT DEFAULT 'pending',
    created_at DATETIME DEFAULT CURRENT_TIMESTAMP,
    FOREIGN KEY (transaction_id) REFERENCES transactions (id)
  );
`);
```

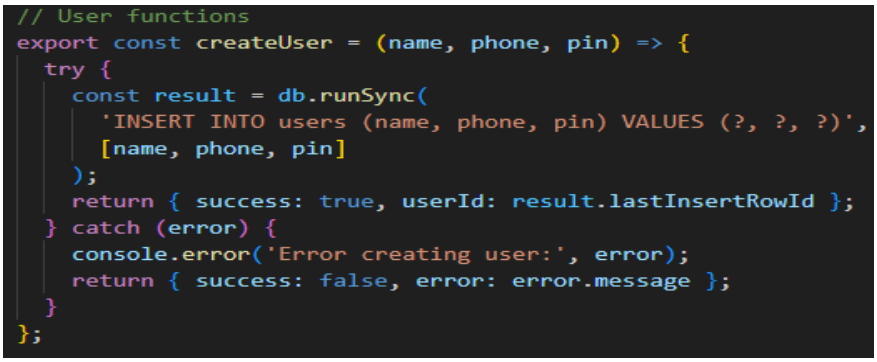
4.3.2 Database Operations

The app does important tasks that help it work properly. When someone signs up, a new entry is made in the users table with a confirmed phone number, a PIN, and a starting balance of ₦100,000 for testing. To check if someone can log in, the app looks for a matching phone number and PIN in the system.

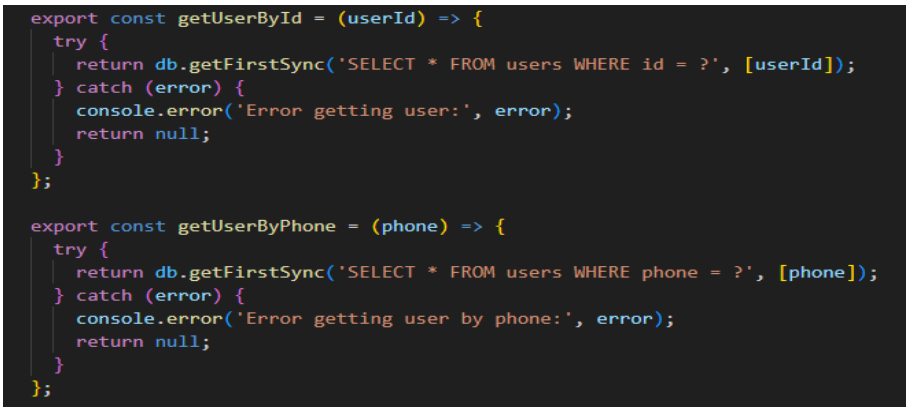
When someone makes a transaction, the app follows several steps. First, it checks if the sender has enough money. Then, it creates a new record for the transaction with all the details, including a risk level. If the person receiving the money is already registered, both accounts are updated. For

transaction history, the app finds all records where the user is either sending or receiving money, puts them together, sorts them, and shows them in order. If someone wants to reverse a transaction, the app checks if it's possible (within 24 hours). If it's allowed, the money is taken back from the recipient, given back to the sender, the transaction status is changed to “reversed,” and the action is recorded in the reversal_requests table. All these steps happen at the same time to keep the data correct and consistent.

Code Snippets for Core Database Functions



```
// User functions
export const createUser = (name, phone, pin) => {
  try {
    const result = db.runSync(
      'INSERT INTO users (name, phone, pin) VALUES (?, ?, ?)',
      [name, phone, pin]
    );
    return { success: true, userId: result.lastInsertRowId };
  } catch (error) {
    console.error('Error creating user:', error);
    return { success: false, error: error.message };
  }
};
```



```
export const getUserById = (userId) => {
  try {
    return db.getFirstSync('SELECT * FROM users WHERE id = ?', [userId]);
  } catch (error) {
    console.error('Error getting user:', error);
    return null;
  }
};

export const getUserByPhone = (phone) => {
  try {
    return db.getFirstSync('SELECT * FROM users WHERE phone = ?', [phone]);
  } catch (error) {
    console.error('Error getting user by phone:', error);
    return null;
  }
};
```

```
// Transaction functions
export const createTransaction = (senderId, senderName, recipientPhone, recipientName, amount, status, riskLevel) => {
  try {
    const result = db.runSync(
      'INSERT INTO transactions (sender_id, sender_name, recipient_phone, recipient_name, amount, status, risk_level) VALUES (?, ?, ?,
      [senderId, senderName, recipientPhone, recipientName, amount, status, riskLevel]
    );
    return { success: true, transactionId: result.lastInsertRowId };
  } catch (error) {
    console.error('Error creating transaction:', error);
    return { success: false, error: error.message };
  }
};
```

```
export const createReversalRequest = (transactionId, reason, details) => {
  try {
    const result = db.runSync(
      'INSERT INTO reversal_requests (transaction_id, reason, details) VALUES (?, ?, ?)',
      [transactionId, reason, details]
    );
    return { success: true, requestId: result.lastInsertRowId };
  } catch (error) {
    console.error('Error creating reversal request:', error);
    return { success: false, error: error.message };
  }
};
```

4.3.3 Data Integrity and Validation

Making sure the data was correct and dependable was very important when setting up the database. The system has several ways to stop wrong information from getting in. It checks phone numbers to make sure they follow the right Nigerian format, requires PINs to be exactly four digits, and stops negative balances by checking if there are enough funds before any transaction happens.

The system also uses foreign key rules to make sure every transaction is linked to a real user ID, which stops records from being left without a user. The users table stops people from having the same phone number more than once. Fields like balance, status, and timestamps have default values so each record starts with proper information. If something goes wrong, the system catches the error, logs it, and shows clear messages to the user instead of making the app crash.

4.4 Core System Implementation

4.4.1 User Authentication System

The authentication system allows users to securely and easily access their accounts. When signing up, users provide their name, phone number, and a four-digit PIN. The system checks if the phone number is already taken and confirms the PIN has exactly four digits before creating a new account with a starting balance. Once registered, users are automatically logged in and taken to the main screen.

To log in, users need to enter the same phone number and PIN used during registration. The system checks these details against the stored data. If the login is successful, the system gets the user's ID, name, and balance and sends this information through React Navigation so the user remains logged in as they use the app. If the login fails, the system shows an error message so the user can try again.

The database helps keep everything running smoothly. Foreign keys make sure all user references are correct, the unique phone number rule stops duplicate accounts, and default values set up the account properly from the start. If any database problems happen, the system handles the error and tells the user what went wrong without making the app crash.

4.4.2 VHC Module Implementation

Risk Assessment Algorithm

The Verify stage in the VHC model uses a risk assessment tool that checks transactions based on how well the recipient is known and the amount of money involved. It begins with a trust score of 20 for people the user doesn't know. Then it looks at the user's past transactions. If the recipient has received money before, the trust score goes up to 70, meaning the user trusts them more.

The tool then compares the current transaction amount with the user's usual spending. If the amount is more than three times the average, it gives a risk score of 80; otherwise, 30. These scores are added together to determine the final risk level. If the total is above 70, it's considered HIGH risk and the transaction is held for 90 seconds. If it's between 40 and 70, it's MEDIUM risk with a 60-second hold. If it's below 40, it's LOW risk and the transaction is held for 30 seconds.

Code Snippet: Risk Calculation Algorithm

```
// Calculate risk level
let riskLevel = 'LOW';
if (isNewRecipient) {
  riskLevel = 'HIGH';
}
if (amountNum > 20000) {
  riskLevel = 'HIGH';
}
if (amountNum > 10000 && isNewRecipient) {
  riskLevel = 'HIGH';
}
```

Hold Timer Mechanism

The Hold phase uses a countdown timer to show how much time is left before a transaction is confirmed automatically. The timer starts at 30, 60, or 90 seconds, depending on the risk level, and is controlled using React's useEffect hook, which updates the countdown every second. A progress bar shows how much time is left.

During this time, users can choose to Cancel Transaction to stop the transfer and go back to the home screen without any changes to their balance, or they can Confirm Now to finish the transaction right away. If the user doesn't take any action, the transaction will automatically confirm once the timer runs out. The timer also has cleanup code to stop any unnecessary processes when the component is removed.

Code Snippet: Hold Timer Implementation

```
const handleProceedToHold = () => {
  // Calculate hold duration based on risk
  let holdDuration = 30; // Default 30 seconds
  if (riskLevel === 'HIGH') {
    holdDuration = 90; // 90 seconds for high risk
  } else if (riskLevel === 'MEDIUM') {
    holdDuration = 60; // 60 seconds for medium risk
  }
```

Transaction Confirmation

The Confirm phase finalizes the transaction or cancels it depending on what the user does or if the timer runs out. If the user chooses to cancel during the hold time, the system labels the transaction as “cancelled” and keeps a record for review, but no changes are made to the account balances. If the transaction is confirmed or the timer ends, the full transfer process starts.

The process happens step by step: first, the sender’s balance is lowered, and if the recipient is a registered user (checked using the users table), their balance goes up by the same amount. Then, a full record of the transaction is saved in the transactions table with the status “completed” and the linked risk level. Lastly, the user is taken to a results page that shows either the success message or the cancellation details.

4.4.3 MTP Reversal System

The Mistaken Transfer Protocol (MTP) lets users ask to undo a completed transaction within 24 hours. Users can start a reversal either from the screen that shows the transaction was successful or by picking a completed transaction from the history on the home screen.

When a reversal request is made, the system checks if the transaction happened within the 24-hour time limit by looking at the time stamps. If the time has passed, the request is refused and the user

gets a message explaining why. If the transaction is within the time frame, the system checks the recipient's balance to see if the reversal can happen right away. If the recipient has enough money, the system takes the amount from their account, gives the money back to the sender, changes the transaction status to "reversed," and keeps a record of the action. If the recipient doesn't have enough, the user is told that someone will have to check the request manually, although future versions might use the recovery methods from Chapter Three.

Users can pick one of seven reasons for the reversal: Wrong Recipient, Wrong Amount, Accidental Transfer, Duplicate Transaction, Fraudulent Transaction, Service Not Received, or other and they can also add more details in a text box. These reasons help to better understand what went wrong with the transaction and make the verification process more accurate.

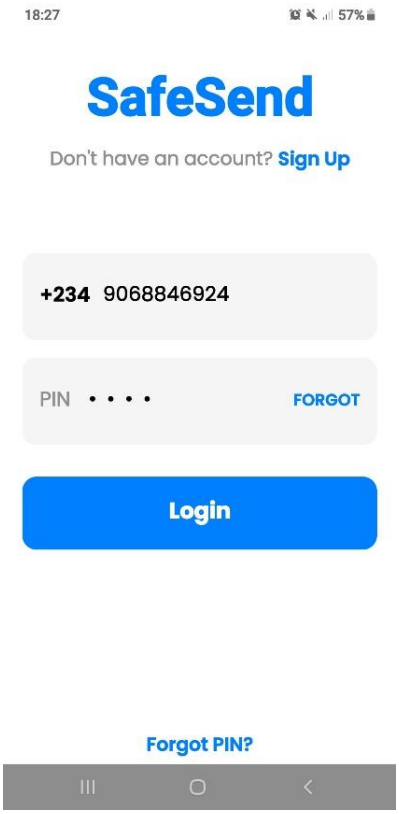
Code Snippet: Checking Reversal Eligibility

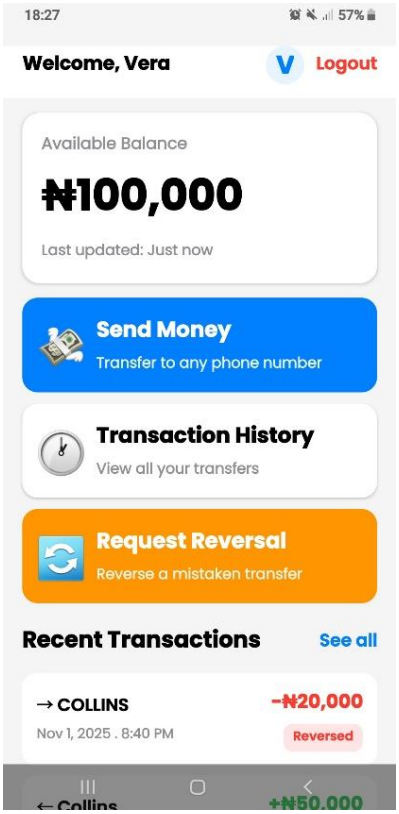
```
// Calculate time remaining
let hoursRemaining = 24;
if (selectedTransaction) {
  const transactionTime = new Date(selectedTransaction.created_at);
  const now = new Date();
  const hoursDiff = (now - transactionTime) / (1000 * 60 * 60);
  hoursRemaining = Math.max(0, 24 - Math.floor(hoursDiff));
}
```

4.5 User Interface Implementation

4.5.1 Application Screens

The SafeSend application consists of eight primary screens that guide users through the complete transaction lifecycle from authentication to reversal requests. Each screen was designed with clarity and usability as primary objectives, ensuring that users can easily understand and interact with the system even during their first use.





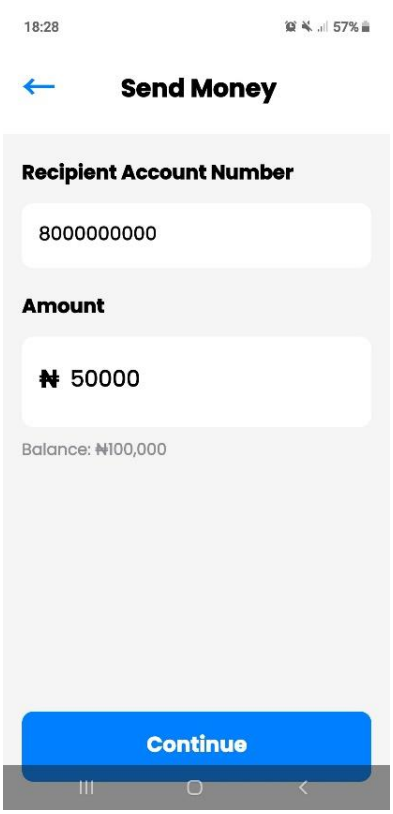
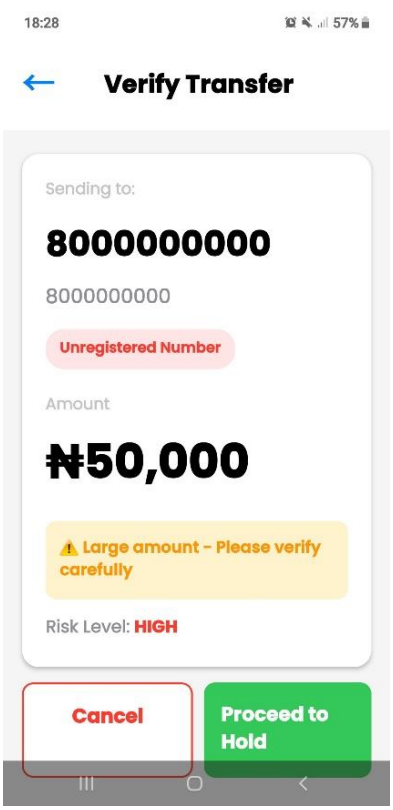
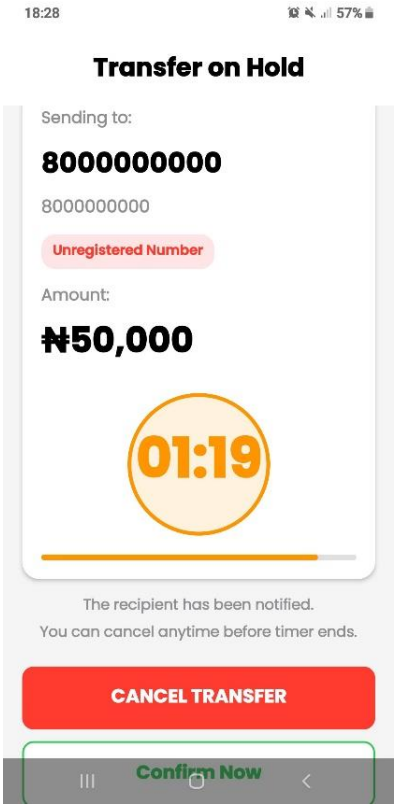


Fig 13: Login Screen **Fig 14: Home Screen** **Fig 15: Send Money Screen**





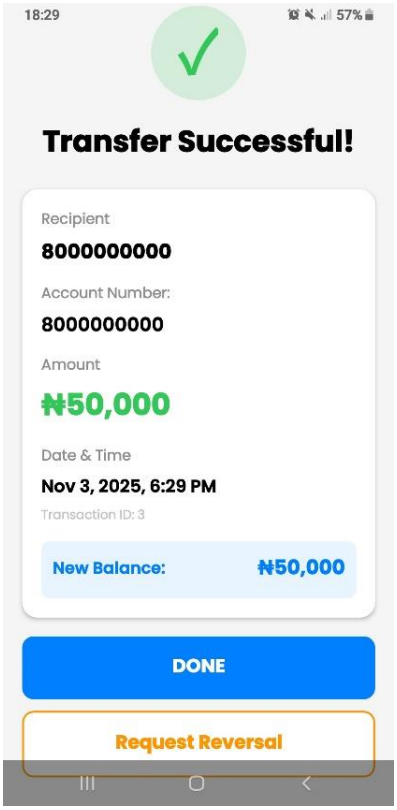


Fig 16: Verify Transfer **Fig 17: Hold Timer** **Fig 18: Transfer Result**

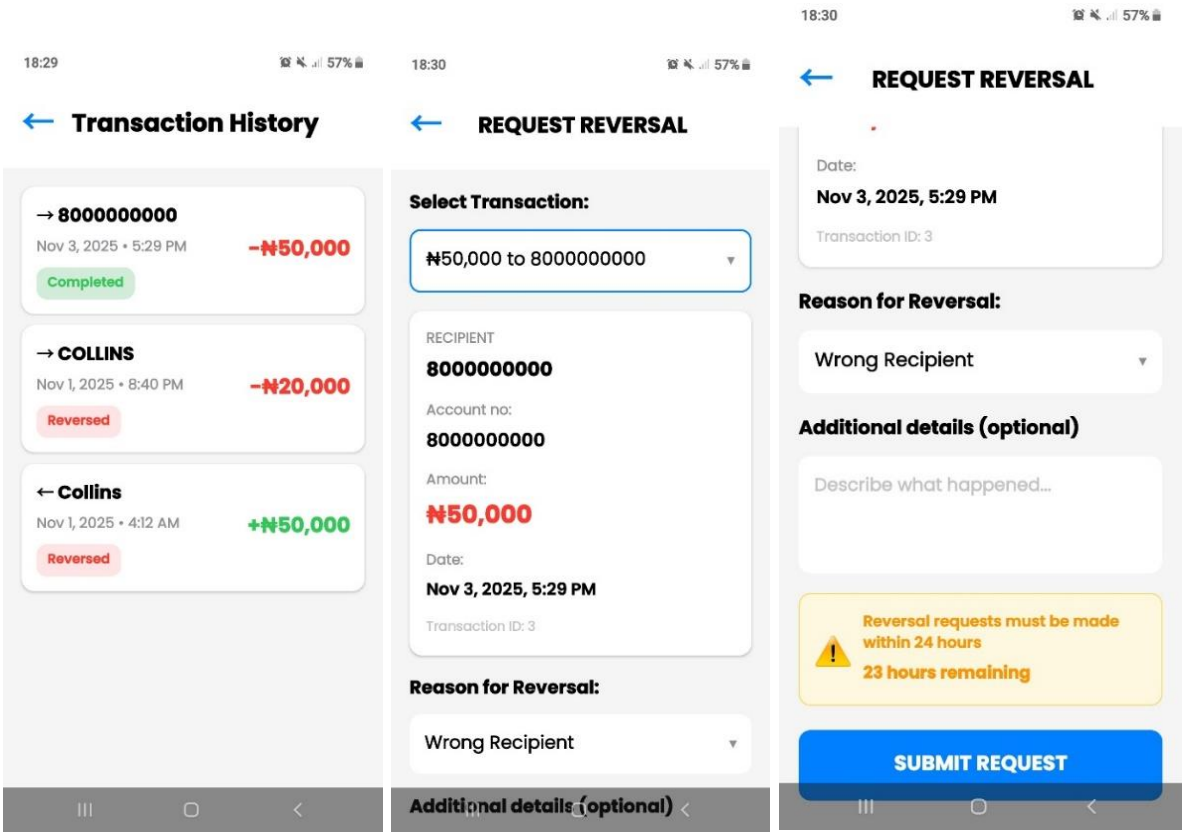


Fig 19: Transaction History **Fig 20: Reversal Request**

4.5.2 Design Implementation

Safe Send's design focuses on being easy to use and accessible for everyone. The Poppins font was picked because it looks clean, modern, and is easy to read on phones. Important details like who the money is sent to and how much is involved are shown in big, bold text to grab attention. This helps users follow along easily as they move through the app.

Colors are used to show how risky a transaction is and its current status. Red means high risk or that money has already been sent. Orange or yellow shows medium risk, and green means low risk or that money has been received. Arrows pointing in and out also help tell the difference between sending and receiving money, making it easier to understand at a glance. Buttons are made big, clearly written, and spaced out so they don't get accidentally pressed. The difference between text

and background is clear so everything is easy to see. ScrollView features make sure all the information can be viewed on phones of different sizes.

4.6 System Testing

4.6.1 Testing Approach

The SafeSend app went through different testing steps to make sure it works well, is reliable, and is easy to use. During unit testing, they checked how the app handles database tasks like making, getting, changing, and deleting information. They also made sure the risk algorithm and the countdown timer worked correctly on various devices.

In integration testing, they looked at how all the parts of the app worked together during complete processes, such as starting a transaction, verifying it, putting it on hold, confirming it, and reversing it. Both cases where reversals worked and where they didn’t were tested to ensure the app handles both instant and manual review situations properly.

User acceptance testing had thirteen people use SafeSend on their own phones to do transaction tasks and give their feedback. This helped find any usability problems, understand what users thought, and make sure the VHC and MTP models match how people actually use mobile payments in real life.

4.6.2 Test Scenarios and Results

Seven test situations were made to check if the SafeSend app works properly. In high-risk tests, a new user tried to send ₦50,000 to a number that wasn't registered, and the system correctly marked it as HIGH risk with a 90-second hold. For low-risk tests, small amounts were sent again and again to known people, and the system showed LOW risk with a 30-second hold, proving it calculates trust scores right. Tests on canceling transactions at different times during the hold showed that all

cancellations worked properly without changing any money balances. Tests for reversing transactions showed that if the request was made within 24 hours and the person receiving the money had enough funds, the transfer was processed.

But if the request was made too late, it was rejected with clear messages. When there wasn't enough money in the account, the system stopped the transfer and didn’t change any data in the database. Tests with multiple users confirmed that both the sender and the receiver had correct balance updates and records of all transactions. All the tests were successful, showing that the SafeSend app correctly uses the VHC and MTP systems. The app also worked the same way on different Android phones and versions.

4.7 User Testing and Feedback Analysis

4.7.1 Testing Methodology and Participant Demographics

The SafeSend app was tested with thirteen people who use mobile payment services in Nigeria. The testing took place over three days, and each person used the app on their Android phone to do important tasks like signing up for an account, sending money to both known and unknown users, seeing how the hold timer worked at different risk levels, and asking for money to be reversed. After using the app, they filled out a detailed survey about their experience and what they thought could be improved.

The users tested were similar to the people who would actually use the app. Most of them, 92%, were between 18 and 24 years old, and one was between 25 and 34. In terms of gender, 62% were male and 38% were female. When it came to jobs, 77% were students, 15% had jobs, and 8% had other types of work.

Everyone who took part used mobile payment apps regularly. 85% of them used these services every day, while the remaining 15% used them two to three times a week. Their experience with

other payment apps helped them give useful feedback on how easy and effective SafeSend is to use.

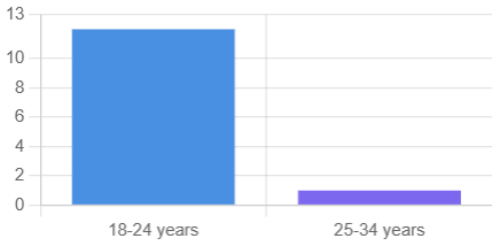


Figure 21: Participant Age Distribution

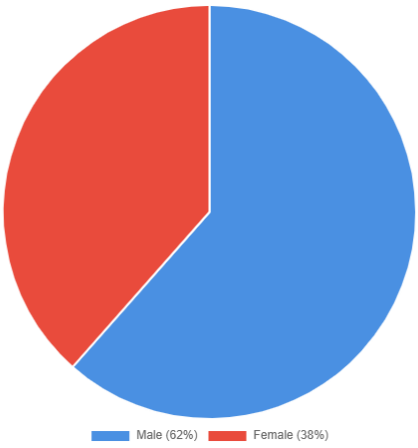


Figure 22: Participant Gender Distribution

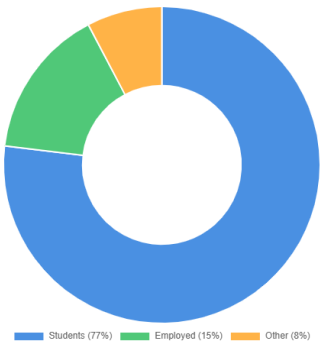


Figure 23: Participant Occupation Distribution

4.7.2 Past Mistaken Transfer Experience Analysis

User testing showed that many people made mistakes when sending money through mobile wallets. About 77% of the participants, or 10 out of 13, had sent money by accident at least once, which shows how big of a problem this is. Of those, 46% made several mistakes, 31% made just one, and only 23% had never made a mistake.

The biggest reason for these errors was entering the wrong phone number, which happened in 70% of cases. This highlights how easy it is to make a mistake when typing in numbers without checking. The next biggest cause was sending the wrong amount, which happened in 60% of cases, often because people were distracted or in a hurry 30% of users said this was their reason for making mistakes. Other reasons included sending money to the wrong person (30%), app problems (20%), and repeating the same transaction (20%). These different causes show that sending money by accident happens for many reasons, so a complete solution is needed instead of just fixing one part of the problem.

Figure 24: Past Mistaken Transfer Experience

The recovery experience for mistaken transfers was largely negative. About 60% of users who made mistakes never recovered their money, resulting in permanent losses. Only 10% recovered funds within 24 hours, another 10% within 1–3 days, and another 10% within 4–7 days. No participant reported recoveries beyond a week, likely because those cases remained unresolved or were abandoned.

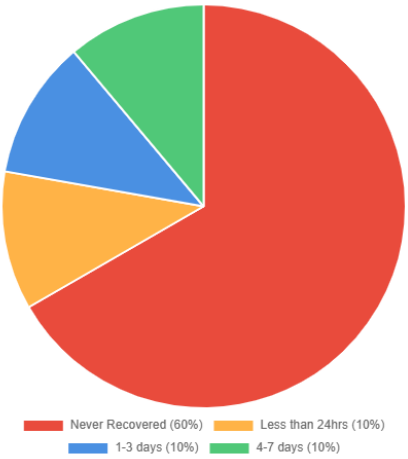


Figure 25: Time Taken to Recover Mistaken Transfers

The financial impact of these mistakes was significant. Half of the affected users (50%) lost between ₦1,000 and ₦5,000, 30% lost between ₦5,000 and ₦20,000, and 20% lost less than ₦1,000. For many, these amounts represented major portions of their funds, making the loss especially painful. The recovery process was also highly stressful, averaging 4.2 out of 5, with most users rating their stress between 4 and 5, showing it was both emotionally and financially distressing.

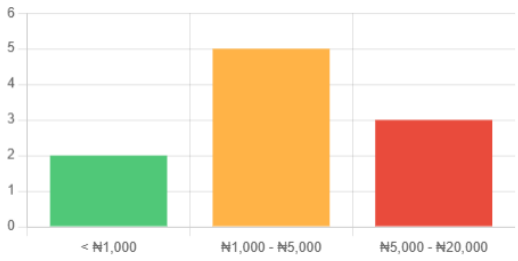


Figure 26: Financial Impact of Mistaken Transfers

4.7.3 VHC Feature Evaluation Results

The test of the Verify-Hold-Confirm (VHC) feature showed that users really liked it and found it useful. Every person involved (100%) saw the hold timer when making transactions, which means it was easy to notice. Most people understood what the hold period meant 62% fully got it, 31% had a basic understanding, and just 7% weren't sure what it was for. When it came to how long the hold should last, 54% thought the time was just right, while 31% believed it should depend on how much money was being moved, showing they understand the idea of adjusting timing based on

risk. Only 15% thought the timer was too long, which means the balance between keeping things safe and making things easy worked well.

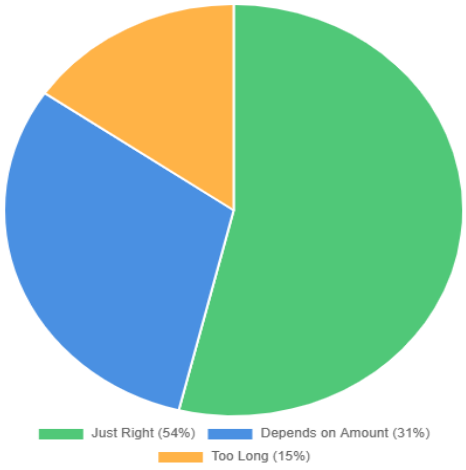


Figure 27: User Perception of Hold Timer Duration

The risk level indicator proved highly effective, with 54% of users rating it “very helpful” and 31% “somewhat helpful.” Only 15% found it unhelpful or failed to notice it, showing that the visual risk cues worked for most users. When asked if the hold timer could have prevented their past mistaken transfers, 38% said “yes, definitely,” 23% “probably yes,” and 31% “maybe.” This 92% positive response strongly validates the hold timer as a solution to real user needs.

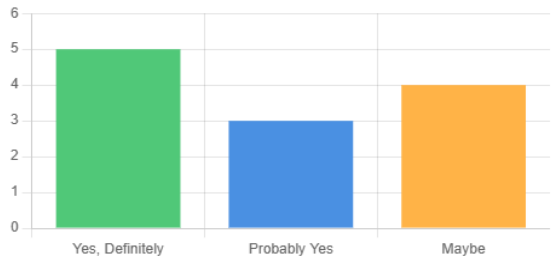


Figure 28: Would Hold Timer Have Prevented Past Mistakes?

The cancel button received a high average rating of 4.3 out of 5, showing users valued the option to abort transactions during the hold period. The verification screen was noticed by 92% of participants and made a strong impact. Displaying recipient names earned the highest feature rating at 4.8 out of 5 for helpfulness. When asked if the verification screen could have prevented their past mistakes, 31% said “yes, definitely,” 46% “probably yes,” and 23% “maybe,” giving a

100% positive response rate. The “new recipient” indicator was rated very useful by 69%, somewhat useful by 23%, and unnoticed or unhelpful by only 8%, confirming its effectiveness in prompting users to double-check before sending money.

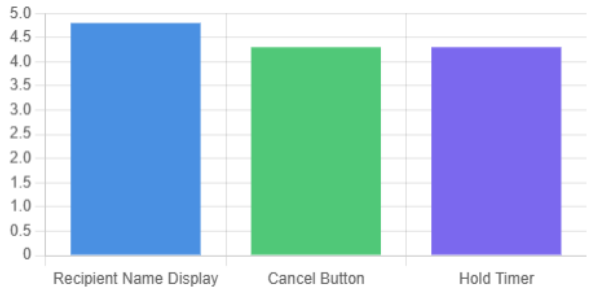


Figure 29: VHC Feature Usefulness Ratings

4.7.4 MTP Feature Evaluation Results

The Mistaken Transfer Protocol features, particularly the reversal functionality, received overwhelmingly positive feedback. A strong majority of 85% of participants tried the reversal feature during testing, demonstrating high interest in and engagement with this capability. Of those who used it, 64% rated it as "very easy" to use, while 27% found it "easy," and 9% did not provide a rating. This high ease-of-use rating indicates that the reversal interface successfully simplified what is typically a complex and frustrating process.

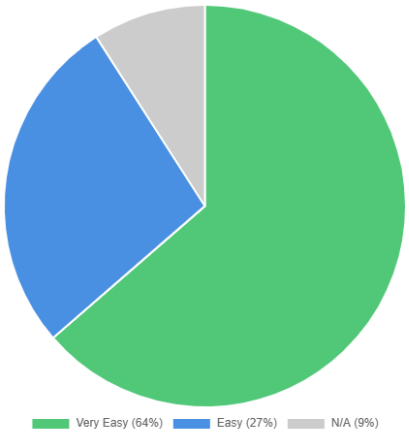


Figure 30: Ease of Using Reversal Feature

The 24-hour reversal window was checked to see if it was the right amount of time. Most people, 73%, thought it was just right. 18% weren't sure, and only 9% felt it was too long. No one said it was too short, which means 24 hours gives users enough time to spot and fix errors without being so long that it causes problems. When users were asked how they felt about the reversal process compared to their past experiences, they gave it an average score of 4.4 out of 5. This shows that the new process is much better than old ways, which often involved calling customer service, waiting a long time, and not knowing what would happen.

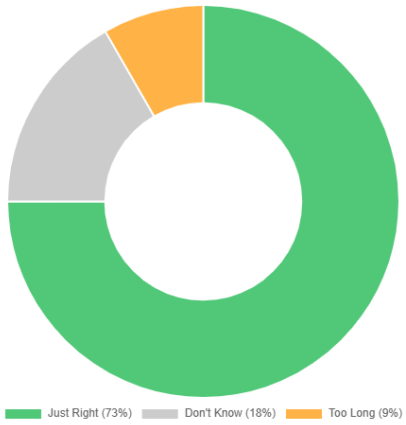


Figure 31: Perception of 24-Hour Reversal Window

4.7.5 Overall Application Assessment

The overall user experience ratings showed both the good parts and things that could be better about SafeSend. The sign-up process got a high score of 4.7 out of 5, meaning users found it easy to create an account without getting confused. Navigation also scored 4.7, showing that people found the app simple and easy to use. The look of the app received a 4.3, which means most people were happy with it, but there might be room to make it look better.

The app's professional appearance had the lowest score at 3.9, meaning it needs more polish to feel as good as other commercial apps. Users felt pretty safe using the app, with a 4.0 rating for security, but adding more visible signs of security could help them feel even more confident. Overall, users were very satisfied with the app, giving it a 4.2 out of 5, which shows they like the main idea and how it works, even though there are a few areas that could still be improved.

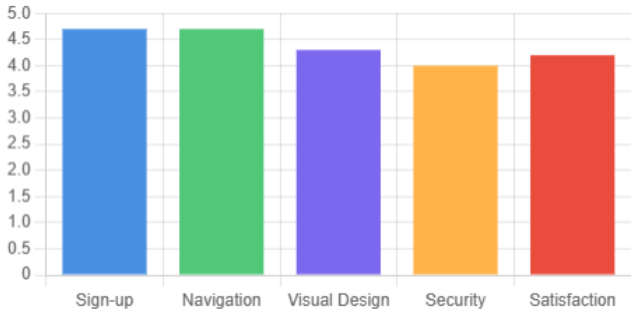


Figure 32: Overall User Experience Ratings

Trust and adoption numbers looked promising. When asked if they would trust SafeSend with big transfers of ₦50,000 or more, 62% said “yes” or “probably yes,” 23% were unsure, and 15% said “no.” This high level of trust for a new product shows that the VHC and MTP features really made users feel more confident. When compared to popular mobile payment apps like Opay, Kuda, and PalmPay, SafeSend got an average rating of 3.4 out of 5. This means users liked its special features, but the well-known apps still had an edge in terms of design and being part of a bigger network.

People also showed strong interest in using SafeSend. About 69% of those surveyed preferred SafeSend over their current apps because of the hold timer and the ability to reverse payments. When asked if they would download SafeSend from the Play Store, 46% said “yes, definitely,” 39% said “probably yes,” and 15% said “maybe.” Altogether, 85% of people were positive about using SafeSend, showing strong interest and possible future use.

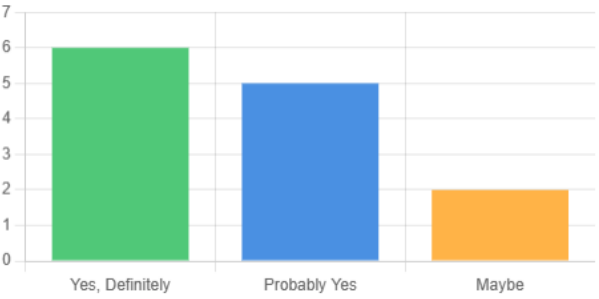


Figure 33: User Adoption Intentions

An interesting insight emerged from the question about willingness to pay for the hold timer feature. The largest group (38%) felt the feature should be free, viewing it as a basic protection mechanism rather than a premium service. However, 31% would pay ₦100-500 per month, 8% would pay ₦500-1,000 monthly, and 23% would pay ₦1,000-2,000 monthly. This suggests that while many users expect error prevention as a standard feature, a significant portion recognizes sufficient value to pay for enhanced protection.

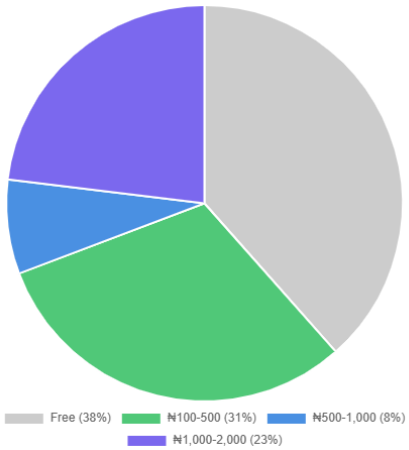


Figure 34: Willingness to Pay for Hold Timer Feature

4.7.6 Key Findings and User Feedback

The user testing phase proved that the main ideas behind the SafeSend design were correct. The 77% rate of wrong transfers showed how common the problem is, while the fact that 60% of users never got their money back showed how weak the current way to undo transfers is. The 92% agreement that a hold timer could have stopped past mistakes strongly supported the idea that the VHC model works well.

Participants liked being able to control their transactions instead of being stuck with transfers that can't be undone. The verification screen, especially showing the recipient's name and risk warnings, made users feel more confident and encouraged them to check things carefully. The reversal feature was the most liked, even more than the hold timer, showing that users really value being able to get their money back as a safety measure. Three main reasons for success came up:

stopping mistakes before they happen, being clear about all the details and risk levels, and having a simple design that's easy to use. These things helped build trust and made it easier for users to make bigger transfers.

However, there were areas that needed improvement. About 38% of users worried about security because the app didn't include BVN or NIN during registration, suggesting that stronger ways to confirm identity were needed. Also, 23% wanted the user interface to look more professional, like commercial apps. Users also asked for extra features like buying airtime or data, paying bills, using OTP verification, linking email, and doing proper identity checks. Though these weren't part of the prototype, they showed what people expect from a full payment platform. As one user said: "It's a good app… I love it… but try to add more features." Overall, the feedback showed that SafeSend's idea is solid.

4.8 Implementation Challenges and Solutions

- Database Management: A major challenge was displaying transactions from both sender and recipient perspectives.
- State Management Across Screens: SafeSend required persistent user data (ID, name, balance) across multiple screens.
- Updated balances were re-fetched after every transaction to maintain real-time accuracy.
- Balancing Security and Convenience: Finding the right hold period duration was challenging.

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATIONS

5.1 Summary of the Study

The study created a full Transaction Verification and Reversal Request Model that uses two different systems working together: The Verify-Hold-Confirm (VHC) model to stop mistakes before they happen, and the Mistaken Transfer Protocol (MTP) to fix things after a transaction is done. Right now, in Nigeria, mobile payments aren't checking who the receiver is properly, which causes errors in 2 to 5% of all transactions. The MTP system helps fix these mistakes by keeping an eye on completed transactions for 24 hours. If someone wants to undo a transfer, the system uses automatic steps to process the request. It also uses a step-by-step way to get the money back. The whole system was built as a mobile app using React Native and SQLite. It has a full database that keeps track of users, their transactions, holds, and reversals.

5.2 Limitations of the Study

The system is still in the prototype phase and hasn't been put into use on real mobile payment systems or tested widely with different groups of people. It assumes that users know how to use smartphones normally and doesn't make special considerations for people with disabilities or support for languages other than English. The study was done in Nigeria, so the results might not apply to other countries that have different rules, user habits, and banking systems.

5.3 Recommendations

Based on the findings and limitations of this study, several recommendations are proposed for different stakeholders in the mobile payment ecosystem.

1. For Regulatory Bodies: The Central Bank of Nigeria and Nigerian Communications Commission should think about requiring all mobile payment platforms to follow minimum verification standards and set maximum time limits for processing reversals across the industry. Consumer protection efforts should include educating the public on mobile payment security, asking platforms to clearly share how often reversals succeed and how long they take, and making sure they are open about the fees involved in reversing transactions.
2. For System Developers: Security should be a top priority, which means using strong encryption to protect all transaction data, adding fingerprint or face recognition for big transactions, and creating ways to prevent tampering during the hold period.

5.4 Suggestions for Future Work

1. Advanced verification technologies can include using fingerprints or facial recognition for important transactions, using voice checks for sending money over the phone, and building smart systems that learn how people usually spend money. Better ways to protect users might look into small insurance plans that help if someone sends money by accident and guarantees from the platform that let users ask to undo a transaction.
2. Long-term studies should look at how much money is actually saved from mistakes, how sending money through mobile apps becomes more popular, and whether it's worth it for the companies running these platforms.

5.5 Conclusion

This research tackled an important problem by creating and putting into use an integrated Transaction Verification and Reversal Request Model. This model combines both preventive and recovery methods in a simple, easy-to-use mobile app. The Verify-Hold-Confirm model gives a practical way to stop transfer mistakes from happening in the first place. The risk assessment tool,

the hold timer, and the reversal request feature all worked as planned, which shows that the main technical ideas are valid. In summary, the Transaction Verification and Reversal Request Model suggested in this research is a solid, complete solution to the ongoing issue of wrong mobile money transfers in Nigeria.

By stopping most errors before they happen and offering good ways to fix any mistakes that do occur, the system can greatly improve user experience, build more trust in mobile payments, and help grow Nigeria’s digital economy.

REFERENCES

Journal Articles, Conference Papers, Reports & Patents

Abdirahman, A. A., Hashi, A. O., Dahir, U. M., Elmi, M. A., & Rodriguez, O. E. R. (2024). Machine learning-based fraud detection across prominent wallet platforms. *SSRG International Journal of Electronics and Communication Engineering, 11*(3), 110. https://doi.org/10.14445/23488549/ijece-v11i3p110

Åkesson, J., Gathergood, J., & Quispe-Torreblanca, E. (2023). *Preventing payments fraud in the FinTech era: New evidence from a behavioural experiment.* Social Science Research Network. https://doi.org/10.2139/ssrn.4532757

Alfaridzi, M. G., Hanggara, B. T., & Az-Zahra, H. M. (2023). Usability testing and user interface improvement of mobile banking application: Livin' by Mandiri. *Matics: Jurnal Teknik Informatika, 15*(1). https://doi.org/10.18860/mat.v15i1.22919

Ayodele, E. (2020). *Investigating blockchain-based smart contracts for cross-border payment settlement, regulatory compliance and risk reduction in international finance* [Research report].

Beckman, M. R., Belsky, S., Marshall, V., & Sohm, R. R. (2020). *Secondary fraud detection during transaction verifications* [Patent]. US Patent Office.

Bui-Huu, D., Le-Nhat, T., & Nguyen-An, K. (2024). Blockchain-powered e-wallet: Enhancing security and fraud detection in online payments. *IEEE Conference Proceedings.* https://doi.org/10.1109/vcris63677.2024.10813393

Dugauquier, D., Van Bochove, G., Raes, A., & Ilunga, J. J. (2023). Digital payments: Navigating the landscape, addressing fraud, and charting the future with Confirmation of Payee solutions. *Journal Article.* https://doi.org/10.69554/mmwu3803

Hartono, N., Rizaldy, A., & Lestari, N. A. (2022). Studi literature sistem keamanan biometrik untuk verifikasi dan transaksi dompet digital. *Journal Article.* https://doi.org/10.24252/shift.v2i2.30

Oguntimilehin, A., Akukwe, M. L., Olatunji, K. A., Abiola, O. B., Adeyemo, O. A., & Abiodun, I. (2022). Mobile banking transaction authentication using deep learning. *Conference Proceedings.* https://doi.org/10.1109/ITED56637.2022.10051553

Owoade, S. J., Uzoka, A., Akerele, J. I., & Ojukwu, P. U. (2024). Automating fraud prevention in credit and debit transactions through intelligent queue systems and regression testing. *International Journal of Frontiers in Engineering and Technology Research*, *7*(2), 48. https://doi.org/10.53294/ijfetr.2024.7.2.0048

Roosli, C. (2022). Security enhancement for SMS verification code in mobile payment. In *Proceedings of the 2022 11th International Conference of Information and Communication Technology (ICTech)* (p. 8). https://doi.org/10.1109/ictech55460.2022.00008

Salman, M., & Mishra, R. (2024). AI-enhanced secure mobile banking system utilizing multi-factor authentication. *International Journal of Experimental Research and Review, 45,* 12. https://doi.org/10.52756/ijerr.2024.v45spl.012

Sowon, K., Luhanga, E., Cranor, L. F., & Fanti, G. (2024). The role of user-agent interactions on mobile money practices in Kenya and Tanzania. *IEEE Conference Proceedings.* https://arxiv.org/pdf/2309.00226

Wang, Y. (2023). Application of big data technology in mobile payment security. *Journal of Research in Social Science and Humanities, 12*(4). https://doi.org/10.56397/jrssh.2023.12.04

Web and Online Sources

Central Bank of Nigeria. (n.d.). *Payments system.* https://www.cbn.gov.ng/PaymentsSystem/

DPI Africa. (n.d.). *Mobile apps transforming financial inclusion in Nigeria.* https://dpi.africa/mobile-apps-transforming-financial-inclusion-in-nigeria/

Koriat Law. (n.d.). *Procedure for recovery of erroneous transfer of funds in Nigeria.* https://koriatlaw.com/procedure-for-recovery-of-erroneous-transfer-of-funds-in-nigeria/

LawPavilion. (n.d.). *Erroneous bank transfers in Nigeria: The costly mistakes, fraud risks, and legal remedies.* https://lawpavilion.com/blog/erroneous-bank-transfers-in-nigeria-the-costly-mistakes-fraud-risks-and-legal-remedies/

Trusted Advisors Law. (n.d.). *Erroneous bank transfer: How to recover mistakenly transferred funds legally.* https://trustedadvisorslaw.com/erroneous-bank-transfer-how-to-recover-mistakenly-transferred-funds-legally/

TrustedDecision. (n.d.). *The ultimate guide to payment reversal: Everything you need to know.* https://trustdecision.com/resources/blog/ultimate-guide-payment-reversal-everything-you-need-to-know

Tymlova, P. (n.d.). *Top mobile payment innovations changing the way Nigerians transact.* https://www.tymlova.com/blog/top-mobile-payment-innovations-changing-the-way-nigerians-transact

APPENDIX

SOURCE CODE

LOGIN SCREEN

```
import React, { useState } from 'react';
import { StyleSheet, Text, View, TextInput, TouchableOpacity, StatusBar, Alert } from 'react-native';
import { useFonts, Poppins_400Regular, Poppins_600SemiBold, Poppins_700Bold } from '@expo-google-fonts/poppins';
import { loginUser } from '../database/db';
export default function LoginScreen({ navigation }) {
  const [phoneNumber, setPhoneNumber] = useState('');
  const [pin, setPin] = useState('');

  let [fontsLoaded] = useFonts({
    Poppins_400Regular,
    Poppins_600SemiBold,
    Poppins_700Bold,
  });
  if (!fontsLoaded) {
    return null;
```

```
}

const handleLogin = () => {

  // Validation

  if (!phoneNumber || !pin) {

    Alert.alert('Error', 'Please enter phone number and PIN');

    return;

  }

  if (pin.length !== 4) {

    Alert.alert('Error', 'PIN must be 4 digits');

    return;

  }

  // Attempt login

  const result = loginUser(phoneNumber, pin);

  if (result.success) {

    // Navigate to Home with user data

    navigation.navigate('Home', {

      userId: result.user.id,

      userName: result.user.name,

      userBalance: result.user.balance

    });
```

```
  } else {
    Alert.alert('Login Failed', 'Invalid phone number or PIN');
  }
};
return (
  <View style={styles.container}>
    <StatusBar barStyle="dark-content" />
    <View style={styles.signupContainer}>
      <Text style={styles.signupText}>Don't have an account? </Text>
      <TouchableOpacity onPress={() => navigation.navigate('SignUp')}>
        <Text style={styles.signupLink}>Sign Up</Text>
      </TouchableOpacity>
    </View>
    <View style={styles.inputContainer}>
      <Text style={styles.countryCode}>+234</Text>
      <TextInput
        style={styles.phoneInput}
        placeholder="801 234 5678"
        placeholderTextColor="#C4C4C4"
        keyboardType="phone-pad"
```

```
    value={phoneNumber}
    onChangeText={setPhoneNumber}
    maxLength={11}
  />
</View>
<View style={styles.inputContainer}>
  <Text style={styles.pinLabel}>PIN</Text>
  <TextInput
    style={styles.pinInput}
    placeholder="● ● ● ●"
    placeholderTextColor="#C4C4C4"
    keyboardType="number-pad"
    secureTextEntry={true}
    maxLength={4}
    value={pin}
    onChangeText={setPin}
  />
  <TouchableOpacity>
    <Text style={styles.forgotText}>FORGOT</Text>
  </TouchableOpacity>
```

```
      </View>
      <TouchableOpacity style={styles.loginButton} onPress={handleLogin}>
        <Text style={styles.loginButtonText}>Login</Text>
      </TouchableOpacity>
  );
}
```

SEND MONEY SCREEN

```
export default function SendMoneyScreen({ navigation, route }) {
  const { userId, userName, userBalance } = route.params;
  const [accountNumber, setAccountNumber] = useState('');
  const [amount, setAmount] = useState('');
  let [fontsLoaded] = useFonts({
    Poppins_400Regular,
    Poppins_600SemiBold,
    Poppins_700Bold,
  });
  if (!fontsLoaded) {
    return null;
  }
  const handleContinue = () => {
```

```
    // Validation
    if (!accountNumber || !amount) {
      Alert.alert('Error', 'Please enter phone number and amount');
      return;
    }
    if (accountNumber.length < 10) {
      Alert.alert('Error', 'Please enter a valid phone number');
      return;
    }
    const amountNum = parseFloat(amount);
    if (isNaN(amountNum) || amountNum <= 0) {
      Alert.alert('Error', 'Please enter a valid amount');
      return;
    }
    // Check if user has enough balance
    if (amountNum > userBalance) {
      Alert.alert('Insufficient Balance', `You only have
₦${parseFloat(userBalance).toLocaleString()} available`);
      return;
    }
```

```
// Check if recipient exists
let recipientName = 'UNKNOWN USER';
let isNewRecipient = true;
if (recipient) {
  recipientName = recipient.name.toUpperCase();
  isNewRecipient = false;
} else {
  recipientName = accountNumber; // Show phone number if user not found
}
// Calculate risk level
let riskLevel = 'LOW';
if (isNewRecipient) {
  riskLevel = 'HIGH';
}
if (amountNum > 20000) {
  riskLevel = 'HIGH';
}
if (amountNum > 10000 && isNewRecipient) {
  riskLevel = 'HIGH';
}
```