BABCOCK UNIVERSITY

THIS THESIS SUBMITTED BY

NAME: AKANNI, ADENIYI W.

WAS ACCEPTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS FOR THE DEGREE OF DOCTOR OF PHILOSOPHY IN THE DEPARTMENT OF COMPUTER SCIENCE SCHOOL OF POSTGRADUATE

BABCOCK UNIVERSITY

THE EFFECTIVE DATE OF THE AWARD IS

_____

_____ _____

Date SECRETARY

SCHOOL OF POSTGRADUATE STUDIES

International Journal of Scientific & Engineering Research
ISSN 2229-5518

CLOUD-BASED HYBRID AUTHENTICATION FOR MOBILE BANKING

AKANNI, ADENIYI W.

CLOUD-BASED HYBRIDAUTHENTICATION FOR MOBILE BANKING

NAME: AKANNI, ADENIYI W.

MATRIC NO.: PG/11/0334

DEGREE(S) EARLIER OBTAINED: BSc(Maths), MBA (Fin. & Acc.), PGD (Comp Sc.), MSc (IT)

BEING A THESIS SUBMITTED IN THE DEPARTMENT OF COMPUTER SCIENCE, SCHOOL OF POSTGRADUTE STUDIES, IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF THE DEGREE OF DOCTOR OF PHILOSOPHY, BABCOCK UNIVERSITY, ILISHAN-REMO, OGUN STATE, NIGERIA

2016

CERTIFICATION

This Thesis is titled, Cloud-Based Hybrid Authentication for Mobile Banking prepared and submitted by Akanni, Adeniyi W. in partial fulfillment of the requirements of the degree of DOCTOR OF PHILOSOPHY (Computer Science) is hereby accepted.

_____ (Signature and Date)

Sunday O. Idowu

Principal Supervisor

(Professor)

_____ (Signature and Date)

Oludele Awodele

Co-Supervisor

(Professor)

_____ (Signature and Date)

Olutayo Ajayi

Co-Supervisor

(Ph D)

Accepted as partial fulfillment of the requirements for the degree of DOCTOR OF PHILOSOPHY (Computer Science)

_____

Dean, School of Postgraduate Studies

DEDICATION

To ’Bunmi Akanni, a woman of substance

ACKNOWLEDGEMENTS

The gratitude goes to the Lord God Almighty, the owner of wisdom. Thanks to His Son, Jesus Christ – the Rock where I was hewed and the Blessed Holy Spirit who leads and directs me at every step of the way.I am greatly indebted to many people who contributed in one way or the other to make this degree a reality. Notable among them are my immediate family members (Bunmi, Deolu, Segun and Tolu) who solidly stood behind me.

I am grateful to my supervisors – Prof Sunday Idowu, Prof Oludele Awodele and Dr. Olutayo Ajayi for their patience and understanding during the review and discussion sessions. Their immense contributions helped in this work. My sincere gratitude also goes to Prof. Ben Ogunmoyela, a renowned industrialist and an academia, who happens to be a distant mentor. His rich blend of industrial expertise and academic excellence inspired me to carry on when the road was rough. It is a great privilege to know him through his wife - my beloved boss Mrs. Yemi Ogunmoyela.

I also acknowledge the contributions of several friends and colleagues who supported in some respects. The list includes: Messrs Nelson Uduak, Kanu Okereke, Tijani Akinkunmi, Michael Ola, Malachy Anekwe, Ramoni Alabi, Olumide Adedeji, Kayode Adejumo, Tosin Olakitan, Adedeji Adegbenle, Ibukun Falaye, Akin Fagbohun and Mrs. Gina. Lastly, I want to appreciate the efforts of the men and women of God who encouraged and supported me in prayers for successful completion. They include Pastors J. Asemota, Esho, P. Oluwi, R. Bamgbayan, J. Akinwande, B. Adesina, D. Oladele, O. Oyewole, S. Balogun, O. Dolapo, A. Onokor, C. Arowolo, E.Udeogwu, A. Agada, T. Akinwale, T. Atolagbe, Sisters Abike Adeniyi, Christiana Owolabi among others. I saw their

hands in prayers. Their moral support also assisted in no small way. In all, the Lord has been so good to me despite various challenges and seemingly contrary winds. He saw me through. Blessed be His Holy Name. Amen.

ABSTRACT

One of the current scientific advancements is manifested in Internet Compliance Technology (ICT) of mobile devices with commercial transactions. Software entrenched in cloud technology facilitates cost reduction by the banking sectors through resource opportunity. Inadequacy of the existing security in mobile banking is a threat to ICT by this sector. There is a perceived compromise of Personal Identification Number (PIN) and unauthorized access to cloud data.

With the advancement in technology, the technology of mobile devices has made it easy for operations and activities to be carried out anywhere at any time. The acceptance of mobile technology has received a wide span over the years; the banking sector is not left out in this coverage. This brought about the introduction of mobile banking, a conventional way of carrying out banking transactions by customers anywhere they are at any time. All a customer for now needs is a mobile device with the mobile banking app (application) on it, and a four digits PIN as well as optional password. Then, the customer will have access to the account details as well as balance in the account and with this could transfer funds, pay for goods and services without being in the banking premises.

As acceptable as this mobile banking is, it comes with associated challenges which needs to be addressed, one of which is the issue of identity theft. From various research works reviewed it has been identified that the security of mobile banking application can easily be compromised. This is as a result of ease of impersonation that has been discovered, customers may be careless with both device and PIN, or there could be loss of device which will make an unauthorized user gain easy access to a customer's account. This is because all that is needed to access the account is just a four digit PIN which can be guessed by anyone. Hence, the need to introduce a more robust means of identification and authentication of the account owner not just with PIN alone but with feature such as biometrics features of the owner of the account.

In light of the above, this research looks at securing mobile banking by introducing a cloud based hybrid authentication approach to the mobile banking system. This was accomplished by designing a system that adopts the PIN authentication harnessed with a biometric authentication system.
Touch Base Dynamic Authentication model was adopted. This allows the users PIN and fingerprint to be captured. They are routed via the internet to the authentication servers, PIN and fingerprint Authentication servers hosted in the cloud.
The design adopted binary search for matching of the query PIN with the PIN stored in the cloud. Also for the fingerprint biometric authentication, the research employed image processing technique, where the fingerprint was made to go through various stages of image processing and the important features were extracted using an extraction Algorithm and these extracted features were stored in the cloud. Minutiae were the feature considered on the fingerprint. Matching of a

query fingerprint biometric was done with that which has be stored in the cloud and access is granted based on perfect match found in all.

The proposed solution was found to be adequate and effective compared with the existing PIN-based authentication method. The obtained results show that the False Acceptance Rate (FAR) was o% while the False Rejection Rate was found to be 98%. This implied that it was impossible for an intruder to gain access to the mobile application whereas there was a chance of 2% of denial of access to authentic user. Therefore, this system is recommended for use by banking sectors as well as commercial firms that rely on other mobile application in their regular transactions. For further studies, it is recommended that a sync of other biometric feature with that which has been used may be considered.

Keywords: Hybrid, cloud, biometrics, fingerprints and mobile banking.

TABLE OF CONTENTS

Content **Page**

Content **Page**

Content **Page**

Content **Page**

Content **Page**

Content **Page**

Content **Page**

ABBREVIATIONS

1. Sms – short message service
2. SIM – Subscriber's Identification Module
3. PIN – Personal Identification Number
4. PAN – Primary Account Number
5. App - Application
6. SPOF – Single Point of Failure
7. FAR – False Acceptance Rate
8. FRR – False Rejection Rate
9. 24/7 – every moment
10. RBAC – Role-Based Access Control

Chapter One

INTRODUCTION

1.1 Background to the Study

Technology has become a major enhancement tool in driving banking processes and activities. According to Anyasi and Otubu (2009), Banks that effectively harness their technological resources enjoy real competitive edge among the peers. A typical example is that of mobile banking which offers users the convenience of time and place in carrying out banking transactions (Bankole et al., 2011). Mobile devices, on which m-banking run, have experienced a lot of phenomenal changes in recent times. These may be borne out of a lot of other uses and improvement associated with the technology.

Mobile devices are used to make calls, surf the internet, take pictures, store vital details, and carry out e-commerce activities among others. There are quite a good deal of application over the year that has promoted and encouraged both the development and acceptance of mobile devices. The ease at which processes are carried out without waste of time or need to access a distance location has made is more acceptable. Various institutions have tapped into this technology in making life easier for their clients, as most practice with the various institutions have been made mobile. Banking sector is not left out; this technology made way for mobile banking, which now widely acceptable. As pervasive as this is, there are challenges that are associated, sure as unauthorized user accessing an account via the mobile device and carrying out transaction, loss of mobile device may subject the account to an impostor. Hence, there is need for building necessary safety measure to address the security concerns associated with mobile banking especially, in managing the identity of users (Bamoriya and Singh, 2011; Higgris, 2013; Siciliano, 2013; Wang, 2011).

The main focus of this study is on employing IT to secure mobile banking activities such that necessary controls can be in place at the user's end and for the bank as regards identity management. Mobile devices are used for carrying out financial transactions without making any physical contact with a branch after initial set up. The primary link is the mobile phone number which is susceptible to cloning. Hence, additional controls in form of strong authentication should be in place towards ensuring the security triad of Confidentiality, Integrity and

Availability (CIA) is preserved. Mobile device has become a good companion that houses so many personal details inclusive of bank account information. Loss of such phones may lead to compromise of stored details which can be used to carry out financial transactions as if the impostor is the authentic owner leading to loss to be borne by either the bank or the customer depending on where the liability shifts (Adegbenle, 2013; Akash, 2015; Kim and Hong, 2011; Kossman, 2013). Personal Identification Number (PIN) and biometrics were used to tackle this in this research. Biometrics, on which this research leveraged, was of tremendous usage in taking care of Identity Management challenges due to its uniqueness.

1.2 Statement of the Problem

From practice and review of literature, it has been shown that the existing control that makes use of Identification and Authentication (I & A) method via mobile number and Personal Identification Number (PIN) is not reliable due to possible compromise. Managing and securing the identity of users of mobile devices has been a problem. It becomes more pronounced and complicated when it involves direct movement of funds. Ability to establish that a banking transaction, carried out from a mobile device, is initiated from the rightful owner of the account has been a major challenge for banks. These security implications underscore the need for a resilient identity management structure to authenticate and ensure secure mobile banking transactions. The research is focusing on designing a novel model for authenticating users on mobile banking with respect to cloud computing. This is to provide a 2-factor authentication to guarantee more security in mobile banking.

1.3 Aim and Objectives

The main aim of this work is to design and implement a hybrid PIN and bimodal biometric authentication to further strengthen the current security control challenge being faced in mobile banking transactions resulting from identity theft. The research sets out to achieve the following specific objectives:

1. To present a critical analysis of existing system of mobile banking.
2. To develop a hybrid model for mobile banking authentication using PIN and bimodal biometric feature.

3. To implement and evaluate the efficiency of the new system with regards to false acceptance rate (FAR) and false rejection rate (FRR).

1.4 Methodology Overview

To achieve the aim and objectives of this study, a review of existing related work was carried out to critically analyze the existing systems on mobile banking. Study of biometrics, bimodal biometrics and image processing as well as cloud computing were equally done. Thereafter, a hybrid authentication system that relies on both PIN and bimodal biometric features was designed.

The design was such that two different systems were harnessed together in the cloud for the purpose of user or person identification. The first is that which works based on *what the user knows*, that is a system that recognizes the PIN of a client. While the second is the biometric system which works based on *what the user has*, this is the fingerprint system. This system adopts the process of image processing, in extracting the biometric feature of a fingerprint. All these features are store in the cloud. The image processing technique involves various stages, such as data capturing, noise-remover, segmentation/classification as well as extraction of biometric features. Various algorithms were employed at each stage.

The designed system was developed using a client-server arrangement. The server was hoisted in the cloud (Public) with a Microsoft SQL Server 2008 R2 Express Edition used for the database and Windows Server 2008 for the Operating System (OS). The latter was used due to its high degree of stability and security while the former was adopted because of its resource sharing capability. Java language was used for the development of the codes because it is fast and highly responsive by reducing the network latency – the time it takes to request from a server and get feedback and it is platform independent. Infrastructure-as-a-Service model was used in a public cloud for the purpose of cost reduction.

The implementation stage involves the enrolment of customers. The enrolment involved capturing customer names, phone number, PIN and fingerprint. These details were linked to the account number on the core banking application. This was for the purpose of storing customers'

details in the database for matching. All the acquired information were stored in the cloud storage.

At the testing stage, matching of the PIN at the database level was carried out using binary search method to reduce time taken for searching of details. The biometric input was routed to a database where matching of both the PIN and the biometric features were done. Then, Minutia-Based Fingerprint Matching Algorithm (MBFMA) was adopted for comparing the Minutiae of the fingerprints input during matching. At this point, if successful, the core banking application was updated; otherwise, an error message was flagged.
The efficacy of the system was measured by using standard performance metrics of the False Acceptance Rate (FAR) and False Rejection Rate.

Minutiae are prominent local ridge characteristics in fingerprint which typically consists of ridges and valleys. The MBFMA approach stores only a small number of minutiae points in order to reduce the storage requirement as well as the network overhead. MBFMA is notably used in portable device that requires small storage space. Thus, it was suitable for this research. The next stage is the identification and authentication of users. Before a transaction can be consummated, both the PIN and fingerprint are supplied through the mobile device. one after the other with the corresponding stored PIN and fingerprint. This was done by reviewing

1.5 Scope and limitation of the study

The research was restricted to mobile devices with internet, PIN and bimodal biometric facilities. For the purpose of this study, client-server arrangement was used due to inability of Telco to open up their infrastructures for experimental analysis. Fingerprints and facial images of fifty people were taken at random from a First Generation Bank for easy accessibility by the researcher. Each of the ten fingers of the samples taken shall be captured while any of them could be sufficient for authentication. A predetermined sample of mobile banking users with smart phones is taken and enrolled. Another set of fifty different people who were not pre-enrolled will be tested for possibility of acceptance or otherwise. For security reasons from the bank and Telecommunication Company view points, the study shall be limited to security of the

mobile banking transactions through biometric authentication taking for granted that the end-to-end encryption would be done ordinarily across the channels.

1.6 Significance of the Study

This study is significant in the following ways:

1. The research will help in contributing to the body of knowledge for future researchers who can leverage on the outcome of this work as a benchmark for mobile banking authentication studies. This is because the hybrid authentication solution is relatively new in mobile banking.
2. The work can be of tremendous benefit to nations embarking on Cash-less Policy through secure mobile banking to further strengthen the security around mobile banking. Mobile banking has huge potentials to enjoy better patronage once the security concerns are properly addressed.
3. This research can provide useful input to societies embarking on Identity Management Project by leveraging on the efficient database that can aid tracing.
4. With a little tweaking, the solution can be used for electronic voting where an electorate will only be entitled to only one vote. However, this may involve a secure centralized database for storing capture data.
5. Outcome of this research can be used for boarder control to prevent illegal migrants.

1.7 Spiritual Insight

The Omniscient God knows all and can identify all. The riddle of identity is not a problem to Him since He can identify even as minute as every single hair on our heads (Matthew 10:30). He has also freely given us all things including wisdom. Tapping from His fountain of knowledge, the riddle of identity theft that dates back to the days of Esau and Jacob (Genesis 27:1-38) can be solved. An attempt was made by the Gileadites (Jugdes 12:5,6) to Identify Ephraimites through a passphrase: “Shibboleth” but an Ephraimite would say “Sibboleth”. Through that means, Gilead could identify and killed forty two thousand Ephraimites. Just as White (1892) admonished us to study God’s Word because it gives illumination beyond human comprehension. God speaks in all ages but mortals would neither listen nor learn from His wisdom to do all things. She added that:

"The noble power of the mind may be so dwarfed by lack of exercise on the themes worthy of their concentration as to lose their ability to grasp the deep meaning of the word of God. The mind will enlarge if it is employed in tracing out the relation of the subjects of the Bible, comparing scripture with scripture and spiritual things with spiritual." In actual fact, there is nothing more calculated to strengthen the interest of the intellect than to study the Scriptures. As we read and meditate our darkened hearts become enlightened. If we will give earnest attention to His voice of wisdom and at the same time ask Him for wisdom then will He fill us without measure.

1.8 Definition of Terms

Mobile device is a small handheld device computing device, typically having a display screen with touch input and/or a miniature keyboard.

Input is the term denoting either an entrance or change which is inserted into a system and which activate or modify a process.

Output anything that comes out of a computing device.

Social engineering refers to psychological manipulation of people into performing actions or divulging information.

Hybrid means combination of features

Identity theft is a form of stealing someone's identity in which pretends to be another by assuming that person's identity.

Failure is the state or condition of not meeting a desirable or intended objective.

Single point of failure (SPOF) is a part of a system that, if it fails, will stop the entire system from working.

False acceptance rate (FAR) is the measure of the likelihood that the biometric security system will incorrectly accept an access attempt by an unauthorized user.

False rejection rate (FRR) is the measure of likelihood that the biometric security system will incorrectly reject an access attempt by an authorized user.

Equal error rate (ERR)the rate at which both the rejection and acceptance errors are equal

Accuracy is the condition or quality of being true, correct, or exact.

Identification is the process whereby a network element recognizes a valid user's identity.

Authentication is the process of verifying the claimed identity of a user.

Multifactor authentication is a method of computer access control which a user can pass by successfully presenting authentication factors from at least two of something you know, something you have or something you are.

Cashless Policy refers to the measures and guidelines put in place to reduce cash transactions in an economy.

Wallet is an electronic purse or account where funds can be deposited or loaded to allow for usage.

Know Your Customer (KYC) is a term used to describe the duty that a bank has to perform on its customer

Biometric features are measurements or metrics relating to human characteristics.

Bimodal biometric features are combination of two human characteristics for measurement.

1.9 Outline of the Thesis

The remaining part of the thesis is broken down into four chapters.

Chapter two contains the Review of literature of related work. This is where in-depth study and review of existing work is brought to fore. Chapter three is about the methodology used in the thesis before generating the results. Chapter four covers the results and discussion. This is the section where the obtained results were analyzed and discussed. Chapter five gives the summary, conclusions, recommendations and future work. It is the last chapter of the work. It gives a concise summary as well as conclusions and made necessary recommendations with an eye on future work.

Chapter Two

REVIEW OF LITERATURE

2.1 Introduction

This section provides an extensive literature review for both the security concerns on similar as well as closely related works on authentication. While much work had been done on second factor authentication, not much has been done on cloud-based hybrid solution (of PIN and fingerprint) for authentication of mobile banking transactions.

2.2 Previous Research

2.2.1 Cloud Computing

Mell et al (2011) defined cloud computing as a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (such as network, servers, storage, applications and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. Paladi, 2012 saw cloud computing as a centralized provisioning of computational resources to multiple remote clients. The benefits of cloud computing are enormous ranging from cost to expertise (Zimmermann, 2001; Brooks, 2010; Badger et al, 2010). It has observed that there is always a measure of difficulty in assessing the required bandwidth and hence, payment for the Internet Service Provider (ISP) especially in bandwidth intensive environment may be quite difficult to measure. However, John, (2013) explained that Cloud computing provides a leeway out this. A major derivable benefit from cloud computing is the ability for the cloud service customer to cede the responsibility of providing infrastructure to the CSP. Hence, banks need not bother about whether or not the bandwidth paid for is being sublet to other customers.

In essence, resources such as network, servers, space can be provisioned and released with minimal management effort or service provider interaction (Al Shehri, 2013). There are basically, five characteristics**,** three service models, and three deployment models in cloud computing. The service models are: Software as a Service (SaaS), Platform as a Service (PaaS) and Infrastructure as a Service (IaaS).

The following are the main Deployment Models in cloud computing (fig 2.1):

1. Private cloud: this is a type of model deployed for the use of a single organization comprising multiple consumers
2. Public cloud: this is for the use of the general public. Ownership may be by individuals or business or government organization, or some combination of them.
3. Hybrid cloud: this is made up of two or more distinct cloud infrastructures (private, community, or public) that remain unique entities, but are bound together by standardized or proprietary technology that enables data and application portability (e.g., cloud bursting for load balancing between clouds).
4. Cloud computing, like other forms of virtualization, has some negative security implications (Wooley, 2011). According to Scarfone et al (2011), systems that involve of layers of technologies would have complicated security implications. Unless well attended to with reasonable assurance, it will always be a concern to business owners. Giacomo and Brunzel (2010) have however indicated that traditional outsourcing is similar to cloud computing. Thus, if security concerns can be addressed traditionally then they can in the cloud. A recent survey conducted by a consulting outfit, emphasized that despite the security reasons surrounding cloud computing more than 30% of the respondents agreed to move to cloud within the next 18 months (KPMG, 2013). The survey further disclosed that about 70% of the respondents believed that cloud computing is already delivering its benefits to the users. This goes further to say that Cloud computing is a way to go. Although there may be some security issues, they should be addressed to fully reap the potentials. Arnesen (2013) further corroborates this fact by explaining that he was not aware of any specific case of data compromise traceable to Cloud vendors. This is so because significant resources are being deployed to safeguard information assets placed in the cloud. It should therefore motivate more companies to consider migration to cloud.

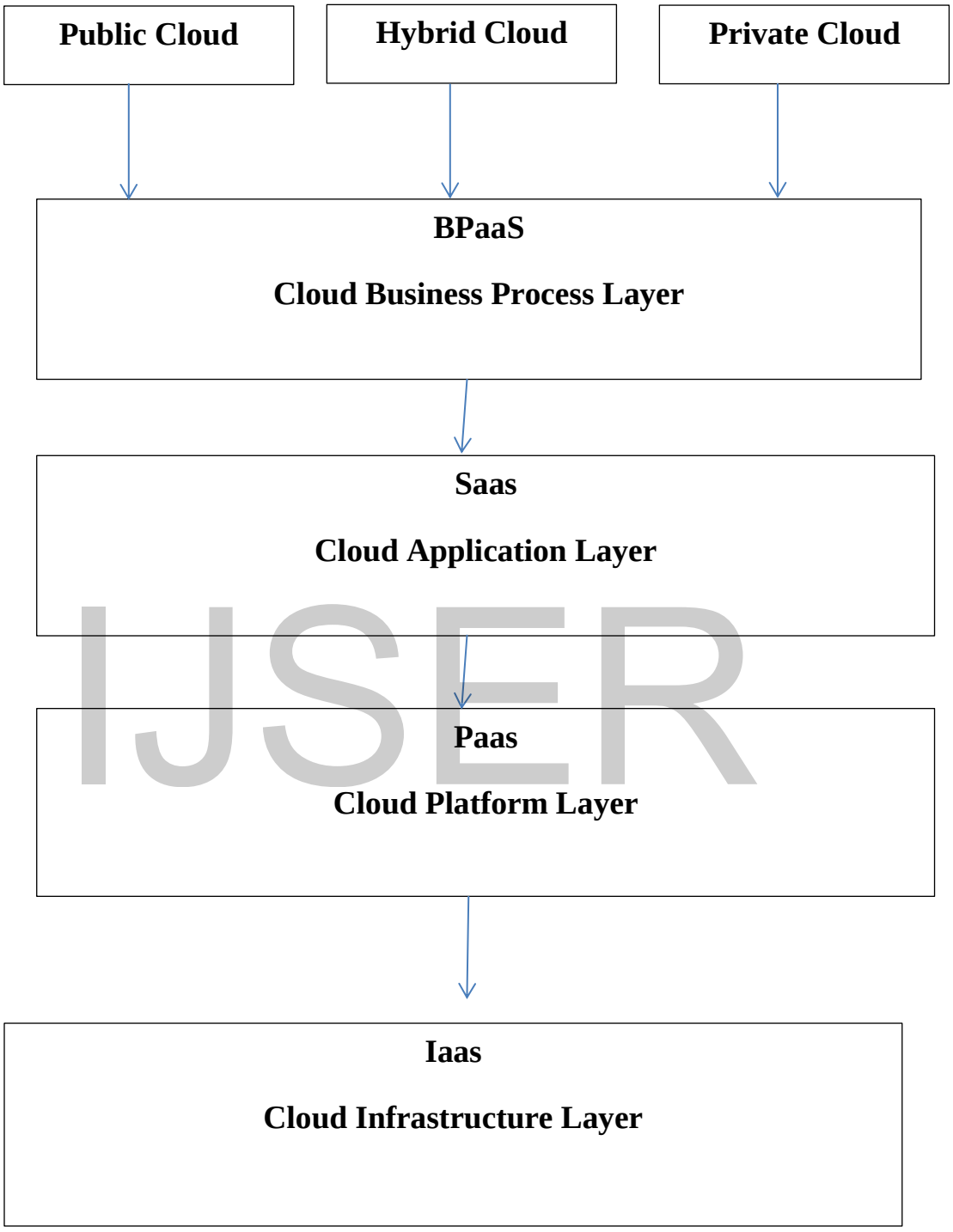

Fig. 2.1: Cloud services and deployment models

(Source: Al Shehri, 2013)

2.2.2 Information Security

Information are the details about a resource or an asset. The resource can be about human or computer. Information Security thus, deals with protecting the details from unauthorized access. In other words, information security is the whole process involved in safeguarding information assets such the Confidentiality, Integrity and Availability (CIA) are preserved to preventing abuse (Pesante, 2008, SANS). Singleton (2007), explained these three basics. Confidentiality relates to keeping sensitive data from unauthorized access. The data can be in form health records, user's log, corporate or personal information such that only those who a reason and with appropriate permission can access. Another component is the Integrity. Integrity has to do with data the representing what it was meant without alteration or modification made to it by unauthorized person. It starts with feeding data into systems, processing of data and securing the data over time to ensure that no unauthorized changes occur. Integrity may be affected if modified before being fed into system. This is called data diddling. It may also occur while processing, storing or retrieval of such data. The last component, Availability, relates to making information available at any time it is needed. Some details are required every moment of the day (24/7). It also includes retrieval of stored data especially, during an incident. He illustrated these components with two basic examples in the table 2.1.

2.2.3 Computer Security

Computer security is concerned with risks related to computer (as an asset) with the aim of preserving the CIA. Computer, in any form, should be guided by professional ethical guidance to prevent any form abuse or disruption. When content of a file (soft or hard copy) is accessed by an unauthorized person, then the confidentiality is compromised. Integrity is lost if data can be manipulated to convey another meaning instead of that which was intended. Information assets should always be readily available any time it is needed otherwise, availability is lost (Dittrich, et al., 2009; Kenneally, et al., 2010).

Table 2.1: Analysis of Two Objects Using the Infosec Triangle

(Source: Singleton, 2007**)**

Information Security Object	**Confidentiality**	**Integrity**	**Availability**
Internet connection	Access to personal details (high risk)	Unauthorized access (high risk)	Degree of reliance on IT (high risk)
Portable storage device	Access to personal details (medium risk)	Unauthorized access (low risk)	Degree of reliance on IT (low risk)

2.2.4 Access Control

Access to a resource describes the permission granted to make use of such a resource. Access control is the restriction of access to a place, facility or resource. Gaining access may therefore be by permission or authorization. Access control mechanism is a component that serves to receive the access request from the subject to decide and to enforce the access decision (Hu, et al, 2014). Access control can be physical or logical. Physical control can be in form of deadman door, security manning or door with a padlock. Logical access control involves usage of one or combination of user id, token, password, PIN and biometrics. According to Huth et al (2012), passwords are very important means of accessing information. They should be well and adequately protected so that wrongdoers do not capitalize on the weakness to steal users' identities. According to Onankunju (2013), there are three main types of access control models which are: Mandatory Access Control (MAC), Discretionary Access Control (DAC) and Role Based Access Control (RBAC). MAC involves granting access to computer resource in a hierarchical form usually at the Operating System level as defined and set up by the System Administrator (SA). Hence, it is not possible for any user to change the access control credentials to a resource. DAC allows users to control access to their own data. It entails an access control list (ACL) that contains a list of users and groups to which the user has permitted access together with the level of access for each user or group. For instance, a user can create a file and grant a Read-Only access to other users. Therefore, other users can only read without the permission to write or alter it. RBAC also called Non-DAC. This is based on job function required by a user. Roles are defined by the organization with permission that matches each usually on a need-to-have or need-to-know bases. For instances, an Auditor's role is different from an SA's. Hence, while permission to write may not granted to former, the latter may have such rights. Usually, there is a role matrix defined which does not allow for overlapping of roles. This is such that an operation group user does not have a right to that of an Accountant neither will a normal user possesses the same right with an SA. RBAC was therefore, used in this research. However, each user must have credentials on the resource that uniquely identifies him and securely managed.

a. Identity Management (IDM)

Identity management (IDM) is a way of identifying individuals in a system such as a country, a network or an organisation and controlling the access to the resources in that system by placing

restrictions on the established identities (Gunjan et al, 2011). Identity theft is a fast growing crime whereby dishonest individuals illegally gain access to an unsuspecting person's account as if the rightful owner did (USA Social Security Administration). Access to the storage should be on need-to-have or need-to-know bases. Both users and services accessing cloud information should always be reviewed and amended accordingly when the need arises to be secured (Gopalakrishna 2009; Thornton, 2012). There are risks associated with mobile IDM similarly to those of the mobile devices. These include: identity theft, eavesdropping, spyware, phishing and lack of user awareness (Papadopouli, 2009). Gunjan, et al, (2012) defined identity management (IDM) as a way of identifying individuals in a system such as a country, a network or an organisation and controlling the access to the resources in that system by placing restrictions on the established identities. Akram and Hoffmann (2008) identified minimal disclosure of information for a constrained use as one of the ways of securing identity. This is to say that personal information should be kept by mobile banking users and bank officers are also not to access the database at the same level. Access to the storage should be on need-to-have or need-to-know bases.

b. Authentication

Authentication is a process of ascertaining that the identified person is the actual person he claimed to be. Prior to authentication, a resource user must have been recognized through the identification process. It can be in terms of user id. Authentication thus, can be through "something you know" like a password, "something you have" like a token (soft or hard) and "something you are" like biometrics. Using only one of "something you know", "something you have" and "something you are" is term as a single factor authentication. Where it uses two of these is called two-factor. If more than two, it is known as a multi-factor. Symantec 2011, opined that two factor authentication is embraced by corporate bodies due to its relative ease of use. Various two factor authentication method have been used by different banks (Northcutt, 2014). Some of these include online keyboard, complex ID's and passwords, software tokens that are generated and sent to mobile phones. However, the problem observed is that they are prone to shoulder surfing when the owner is keying in the details. Ravi, et al (2009) developed a fingerprint matching algorithm through Fingerprint Recognition using Minutia Score Matching method (FRMSM). The approach involved the following stages: image binarization, ridge

thinning and noise removal. A comparative analysis revealed this to achieve higher accuracy through the false matching ratio.

Aloul, et al (2009) researched into using multifactor authentication to secure online banking as well as ATM terminals for secure transactions. They used software token. A set of numbers would be generated and sent via short message services (sms) to a mobile phone. These numbers are copied out within a time space and used for authenticating transactions. This approach is secured as long as the phone is with the account owner. The problem here is that transactions cannot be consummated in the event that the phone is lost. More so, theft of the mobile phone can lead to fraud because it makes the fraudster the direct recipient of the token.

c. Biometrics

Biometrics is the measurement and analysis of unique physical or behavioural characteristics. It is usually for the verifying identity whether in form of genetical dispute resolution or access control. The latter is employed in this research as a form of second level authentication. It has some key benefits that made it readily useful in today's technology as an access control mechanism. Jacobs and Poll (2010) explained biometrics as the use of physical characteristics, behaviour or skills to identify a person. These include palm, finger, iris, voice, DNA, face and a host of others. The basic idea about biometrics is that its features such as palm, iris and face become permanent shortly after birth and cannot be shared like password. For convenience and security reasons, accesses are preferably controlled by biometrics (Nandakumar et al 2009). A comparative study carried out on biometric features places high reliability on finger print as shown in table 2.2. There are various biometrics that have been previously tested with some degree of reliability. There are cases where two or more biometric features are considered in order to increase the degree or level of reliability of an authenticating system. When two biometric features are combined to form an access control, it is known as bimodal biometric feature. When the combined features are more than two, it is regarded as a case of multimodal biometric features for access control. However, this study adopted the bimodal biometrics (fingerprint and facial biometrics) in order to increase the security level of authentication.

Table 2.2: Comparative study of biometric technology

Biometric Technology	Accuracy	Cost	Devices Required
Iris Recognition	**High**	**High**	**Camera**
Retina Scan	**High**	**High**	**Camera**
Facial Recognition	**Medium-Low**	**Medium**	**Camera**
Voice Recognition	**Medium**	**Medium**	**Microphone, Telephone**
Hand Geometry	**Medium-Low**	**Low**	**Scanner**
Fingerprint	**high**	**Medium**	**Scanner**
Signature Recognition	**Low**	**Medium**	**Optic pen; touch panel**

(Source: Nandakumar et al, 2009)

2.3 Matching Algorithm

Algorithm is a procedure for solving a problem. There can be various ways to solve a problem, however, the best is always preferred. Depending on the nature problem vis-à-vis the ultimate goal of the person, there may be need to concede on one part to gain on another. For instance, an algorithm may not be cost effective but may be fast whereas some others may not be as fast but not costly to adopt. Matching algorithm is a way of comparing items such as PINs, phone numbers or fingerprints. In most fingerprint projects, fingers are captured twice such there is a pre-match before storage. The main purpose of this is to reduce false rejection rate occasioned by placement of fingers. Examples of matching algorithms are minutiae-based algorithm and Binary search algorithm. Chen and Gao, 2007 considered minutiae-based matching algorithm as very fast. The approach utilized phase correlation to determine the alignment between two sets of minutiae. The similarity was measured between the template minutiae set and the aligned input set. The method was simple and did not search for corresponding minutiae pairs. The study only made use of the locations and directions of sparse minutiae points in fingerprints. This is a good development in terms of speed and space, invariably, cost. Binary search algorithm is one of the

fundamental algorithms in computer science. It involves sorting data in a sequence. It is fast (Topcoder, 2015). Many of these have been used with proven high level of reliability.

2.4 Mobile Banking Overview

Mobile banking is an Information and Communications Technology (ICT) application considered to be of vital use among people in various countries who are likely to have dissimilar cultural backgrounds. Research into the use and adoption of mobile banking has shown varied findings in different countries across the globe. This can be attributed to the diversity of the cultural landscape in different countries. The development of mobile banking in a country is likely to be determined by some characteristic factors which are unique to that country. For instance, Nigeria running a cash-based economy, aims at reducing cash in circulation through integration of the numerous unbanked citizens introduced mobile banking. Furthermore, to actualize the vision 2020:20, Nigeria considers mobile banking as a necessity. Hence, the introduction of cash-lite into the Nigerian economy (Odumeru 2013, CBN). Similarly, M-PESA was introduced in Kenya and the effect has been quite impressive (Ayo et al 2012). Many unbanked citizens in the country were allowed to have banking relationship. For instance, the nomadic cattle farmers, who spend most part of their time in the forest where no banking facilities can be availed, were able to carry out transactions through their mobile devices. Simotas et al (2011) consider mobile banking model as a major issue to be critically looked into. Two business models notably exist for mobile banking as it relates to channels:

1. Mobile banking as a separate channel;
2. Integrating it with other channels.

Various authors attempted to define and explain mobile banking. In the context of this research, Donner and Tellez (2008) referred to mobile banking as a set of applications that enable people to use their mobile devices to manipulate their bank accounts, store value in an account linked to their handset, transfer funds, or even access credit or insurance products. Klein and Mayer (2011) maintained that mobile banking is gaining a remarkable speed all around the globe. They stressed that it is going to revolutionize the way banking is run in developed and developing

nations. Ondiege (2010) saw mobile banking a way of taking banking to people. While a vast majority of ordinary people remained unbanked, mobile banking is the only way of achieving financial inclusion since this population also makes use of mobile phones.

Framework shows relationship with subject matter most especially in showcasing the underlying principle of the research – PIN and bimodal biometric authentication for mobile banking. Conceptual model consists of access control with respect to mobile banking. It uses both PIN and fingerprint biometric to secure access to mobile banking transaction. While PIN is supplied through the keypad of a mobile device and fingerprint is added through touch-based feature to further strengthen the mobile banking transaction. This position was supported by Meng (2013).
According to Mobile Marketing Association (2009), people are already used to online banking. However, by the end 2010, more than 30% of U.S. population will migrate to mobile banking. The trend continues all over the world. For the fact that mobile devices are more affordable compared to internet facility, more users are bound to embrace mobile banking. Different programmes (such as financial inclusion and rural banking) are being organized to educate the populace on the need to adopt mobile banking.

Mobile banking is rolled out as a banking product to facilitate banking transactions anytime, anywhere. It is most likely going to take care of ATM transactions sooner than later. Most banks run mobile banking with services such as alerts, account balances, bills payments, funds transfers and cash withdrawals at ATM points. In Nigeria, mobile banking are carried out using two main platforms: either through a bank account or an electronic wallet. AuWerter (2012) itemized the capabilities of carrying out so many activities with mobile devices but added that customers of banks are scared of making banking out of these devices because of security worries. However, solving the security issues in terms of IDM will lay more credence to mobile banking transactions. Ullrich (2012) saw mobile banking as an emerging channel with a lot of benefits. However, a major identified challenge why customers are not embracing it is lack of confidence in the security of the services especially, when the devices are lost or PINs are compromised.

2.4.1 Identity Management in a Banking Environment

Banking activities can be broadly viewed under two broad categories, for the purpose of this discussion, Government-oriented Bank and Public-oriented Banks. The Government-oriented is the Apex Bank in any country. It is otherwise known as the Government Bank. It performs functions such as printing of currency notes, mopping up liquidity in the circulation and controls other banks in the country of domiciliation. It also moderates the activities of the Public-oriented Banks through various policies and guidelines issued depending on the goal it sets to achieve. In Nigeria, for instance, the Apex Bank is called the Central Bank of Nigeria (CBN). It issues policies and guidelines at various times to guide the operations of banking activities in Nigeria.

The Public-oriented Banks are basically the Commercial Banks, Merchant Banks and Development Banks. They are the ones being used by individuals and corporate bodies. They accept deposits, grant loans to borrowers and also allow withdrawals. However, to maintain banking relationship with a public bank, a customer is expected to be identified. The commonest type of public bank is called the commercial bank. A commercial bank is expected to be able to identify its customers through the "Know-Your-Customer" (KYC) policy of the CBN. The CBN gave three classes of this. According to Chukwu (2013), they are tagged Tier 1, Tier 2 and Tier 3. The classes are based on the level of identity expected to be kept of the customers by the banks.

Tier 1 KYC also known as Low-Value Accounts. These require minimum KYC documentations. These include: Name, place and date of birth, gender, address and telephone numbers. Verification of such details is not required. They came into existence as a result of CBN policy on financial inclusion such that everyone can be banked. Artisans or normandic cattle rearers, who may not have any valid means of identification (such as Driver's License, International Passport, National Id Card or voter's card). Such accounts must have the following features:

1. Must be Savings account only.
2. Must be for local operations alone – not allowed do foreign transfers.
3. Account opening documents can be sent electronically.
4. Can do mobile banking.

5. Allow deposit into the account other people but withdrawals must be by the account owner.
6. Can do ATM transactions
7. Single maximum deposit must not exceed ₦20,000
8. Maximum cumulative balance must not exceed ₦200,000
9. Maximum single transaction limit of ₦3,000 and a daily limit of ₦30,000 on mobile banking

Tier 2 KYC also called Medium-Value Accounts. They require medium level documentations. The basic information supplied by the account holders must be verified. These accounts can be opened personally at a bank branch by the account holders. The accounts have the following features:

1. Must be Savings account only.
2. Valid means of identification required.
3. Must be for local operations alone – not allowed do foreign transfers.
4. Account opening documents can be sent electronically.
5. Can do mobile banking and fund transfers.
6. Allow deposit into the account other people but withdrawals must be by the account owner.
7. Can do ATM transactions
8. Single maximum deposit must not exceed ₦50,000
9. Maximum cumulative balance must not exceed ₦400,000
10. Maximum single transaction limit of ₦10,000 and a daily limit of ₦100,000 on mobile banking

Tier 3 KYC known as High –Value Accounts. The accounts have no balance restriction placed. They must be opened by physical presence of the customers. Other features include:

1. The accounts can be either Savings or Current.
2. Valid means of identification required.
3. Can do both local and foreign transfers.
4. Can do mobile banking and fund transfers.

5. Can do ATM transactions
6. Maximum single transaction limit of ₦100,000 and a daily limit of ₦1,000,000 on mobile banking

2.4.2 Mobile Banking and Internet Banking

The mobile banking is an aspect of banking services to customers on their mobile devices with a view to making transacting at any convenient time and place - specifically the operation of bank current and deposit or savings accounts – less strenuous (Archana and Vineet, 2012). E-banking comprises of mobile banking, internet banking as well as ATM. It is referred to as the provision of banking products and services through electronic delivery channels. There are benefits derivable from e-banking which include: cost reduction, time saving, convenience and easy access to account information (Chan, 2013). However, according to Chikom et al, 2006, there are attendant challenges with e-banking. These are: lack of trust in epayment system, low level of computer literacy, lack of or limited regulations on ebanking and cross-border issues. Mobile banking involves carrying out banking activities through mobile devices. This may not involve the use of internet. Risks in online banking mainly revolve round privacy, financial and social risk (Lee, 2008). Conversely, internet banking uses internet facility to transact. It may consist of usage of mobile device with internet facility. Mobile payment can be viewed as a subset of m-commerce which provides a method for conducting micropayment to facilitate mobile commerce transactions. It is a way of using a mobile device for transferring money from payer to receiver through an intermediary (Abdulah, 2012; Chavan, 2013; Khan, et al., 2009; Mallat, 2007; Masocha, et al., 2011; Zhao, et al, 2013,).

2.5 Mobile Devices

A mobile device is a pint-sized computing device with a mini keyboard and a touch or non-touch screen. It is referred as a hand-held computer (Viswanathan, 2015). Examples of mobile devices are: mobile phones, smartphones, Personal Digital Assistants (PDAs), Pagers and Personal Navigation Device. PDAs and smartphones are the most preferred mobile devices. Smartphones are ubiquitous due to their extensive usage. As at 2012, the mobile phones produced were far above 6 billion. It was further expected to grow by a yearly figure of 1 billion. By extrapolation, the mobile phones in circulation should be running to about 9 billion. They are in high demand

due to value attached to them different users of different culture and ages. Mobile phones have quite a number of other uses apart from normal calls (Swan, et al., 2007; Woodill, 2012). In much the same way, tablets are not just for viewing materials. They can, as well, be used to collect data in various formats and transfer them to other devices or systems. However, US-CERT, 2010 in her technical paper on cyber threats highlighted various threats against mobile devices. These include social engineering, exploitation of social networking, mobile botnets and exploitation of m-commerce among others. Efforts should therefore, be channels towards mitigating these threats and associated risks.

2.6 Review of Closely Related Works

Bank of America, (2006) implemented Sitekey application (app) and made bold to tell customers that they are 100% guaranteed in the event of any fraud loss. SiteKey was developed by RSA Data Security Company. It was meant to provide a mutual authentication between a website and a user using cookies. A user is identified just when the site opens. Where the browser does not contain a client-side taken from a previous visit, user is made to answer some security question. If correctly answered, the site displays the company logo and the previously configured phrase to give him the assurance that he is at the genuine site. Otherwise, it is a phishing site. So, he is identified. The user is then required to verify that he is the user he claims to be by supplying the password and further authentication done with facial recognition. Human face was implemented at the apps level. Facial recognition was used to provide the needed authentication. There are issues with this. Firstly, in terms of accuracy, it is on medium level. Secondly, there is no back up for human face and a slight change may lead to denial. Another weakness of this solution is that whether or not the security question is correctly answered, a phrase is displayed. Many users do not pay attention especially when they are in a hurry. Bank of America, in May 2015 has hinted its numerous customers that SiteKey would discontinued before the end of the year.

Scotiabank, (2010) implemented a software token that is stored on the mobile device as a second level means of authentication. Software token is a type of two-factor authentication security device. It is usually stored on an electronic device like a desktop computer, a laptop or a mobile device. It can also be duplicated and cheaper than hardware token. In comparison with a hardware token, its battery does not run out. A user accesses the software token from the device

where it is installed after supplying the PIN or password, it generates a set of digits to be used as a second factor authentication. Software token are reliable. The issue of no back up was taken care of. However, because it was stored on the device, an outright theft can lead to fraud.

Edsbacker (2011) maintained that today's mobile devices make use of SIM cards which are forms smartcards. They are capable of carrying network identity information as well as storing different applications. They can be seen as computer on their own. They can process and store data. We can then leverage on this to store mobile banking apps which interfaces with the bank's host. The only shortcoming is that each time SIM card is lost or damaged, the app would need to be re-installed.

Adegbenle (2013) investigated fortifying the existing control on ATM (Automated Teller Machine). In addition to the existing four digit PIN and card, he introduced the fingerprint as a biometric means of authentication. When a customer approaches an ATM terminal, inserting his card, some basic checks are run on the chip for identification purpose. The cardholder is then required to supply his PIN before adding the fingerprint as a second factor authentication. The solution gave a high degree of comfort because there was a near impossibility of a fraudster gaining access to a cardholder's account. However, it was designed for an ATM environment.

Kinsbruner (2013) stressed the relevance of preserving identity in mobile banking through voice as a second factor. The solution involves accessing the mobile banking application with PIN and further authenticated by the user's voice. Before usage, the customer must have been asked to speak a word or phrase. Same is recorded for subsequent use. This is a major breakthrough and gives so much comfort. However, false rejection rate (FRR) is high with this solution. May be due to change in whether or cough or voice cracks, the voice tends to change thereby causing a mismatch between the pre-recorded and that of the life user (Krinsbruner, 2013). Thus, making it quite unfriendly for users.

Technological advancement has also impacted on the design and features of mobile devices for more user friendliness and security. So much of these are witnessed especially, in iphones and Samsung products. They have added biometric features for an additional security. A review of a recent development in Apple's iPhone5 and Samsung S5 was carried out (Kypreos, 2014 and

Cnet, 2015). Both devices (iphone5 and S5) allow fingerprint authentication on the device. This technology was embraced by some UK banks such as Natwest and RBS. Some African banks like DiamondBank and Zenith International bank of Nigeria have also implemented it in Nigeria. The devices have brought some improvement by employing Touch ID technology. The technology also permits enrolment of up to eight fingers (five for iPhone, eight for S5) to be captured on the device where authentication is done. The goal is to securely lock the device. This approach affords users the ease of usage and the choice of biometrics is made optional. However, the technology faces the risk of a single point of failure (SPOF) seeing that a compromise at the device level is enough to commit irregularities. This reason has made it to come under heavy criticism by users that the banks are just attempting to shift liability to the account holders such that in the event of any loss arising from the usage of the devices, the banks can absolve themselves of the blame. It implies there is need for more careful study.

Central Bank of Nigeria (CBN) in conjunction with the Nigerian Inter-Bank Settlement System (NIBSS) recently embarked on a biometric enrolment project. They are working towards a near-perfect system of uniquely identifying banks' customer that will in turn strengthen the cashless initiative. These days IDM is being done on various platforms using biometrics.

Trewin, et al (2012) opined that password or PIN has its own merits but has its own demerits which must be enhanced in mobile banking. Biometric authentication helps in overcoming issues relating to (inadvertent) compromise through password or PIN. Table 2.3 gives a summary of related work.

Table 2.3: Summary of Closely Related Works

Author/Year	Methodology	Strength	Weakness
Bank of America (2006)	SiteKey Authentication between a website and its user through cookies through facial recognition.	Effective for internet banking to prevent phishing.	1. No back up for the face in the event of mark. 2. Facial recognition is not highly effective.
Scotiabank (2010)	Implementation of a Software Token (SoftToken) on the same device for mobile transaction.	Effective security on mobile banking	It is susceptible to Single Point of Failure (SPOF) in the event that device is lost.
Edsbacker (2011)	Ability to store applications (such as mobile banking apps) on SIM	High processing and storing capabilities	1. Damaged SIM would lead to a repeat of the entire process. 2. Loss of SIM can lead to compromise.
Adengbenle (2013)	Combination PIN and fingerprint on ATM	Prevents intruders from using the card when PIN is compromised	Done on ATM only.
Kinsbruner (2013)	Voice authentication on mobile baking	Effective security on mobile banking	High False Rejection Rate due to voice distortion through cold.
Kypeos (2014) **Cnet (2015)**	Fingerprint Authentication for mobile devices on sophiscated devices like iphone5, S5 and	Effective locking of mobile devices even when PIN is compromised.	It is only for locking the mobile devices but not transaction. Susceptible to Single Point of Failure

	higher versions		(SPOF).
CBN/NIBSS (2015)	Implementation of Bank Verification Number (BVN) by capturing customer details – names, face and fingerprints. Irrespective of number of Banks where a customer maintains accounts, only one BVN applies.	Reducing the risk of identity of theft	Applicable to transactions within the banking halls – such as withdrawals and accessing loan facilities.

Appreciable studies had been carried out in the area of identity management, mobile banking and authentication of transactions as revealed in section 2.0, yet there are security challenges. These gave reason for this research to address the identified gaps in terms of security and effectiveness. This work on hybrid model provides a resilient business model for banks. Since ability to manage the identity of users of mobile banking is critical to its adoption, bank can rely on the solution proffered by the researcher.

The new system addressed the identified weaknesses above which include:

1. Low accuracy level occasioned by high FRR
2. Control over SPOF
3. Countermeasure against compromise without user's involvement
4. Safeguard against shoulder surfing
5. Loss or theft of authentication device (such as OTP)

Chapter Three

METHODOLOGY

3.1 Introduction

This chapter discusses the methodology employed to achieve the set objective stated in the first chapter.. The design was such that two different systems were harnessed together in order to authenticate access into a mobile banking application (app) on a mobile device. The first is that which works based on *what the user knows*, that is a system that recognizes the PIN of a client. While the second is the bimodal biometric system which the works based on *what the user has,* this is the fusion of fingerprint system that adopts Touch Based Dynamic Authentication (TBDA)..All these features are store in the cloud.

In a live scenario, the mobile banking application is stored on the user's mobile device. Enrolment is done at the bank's premises for security reasons. Details obtained include First name, Last Name, Middle Name (optional), phone number, PIN and fingerprint. They are linked to the customer's account number and stored on the bank's server. Once the enrolment is done, customer can then use his mobile device identified (by the SIM) on the database but requiring further authentication via PIN. Upon validation, the installed app opens up for him to carry out his transaction.

3.2 Design of the Proposed System

All above mentioned details were captured to facilitate effective testing. The main challenge was to get the parties (mobile devices manufacturers, the Telecommunication companies and banks) to open up their infrastructures for testing. This has been pretty difficult for security reasons. Even when Samsung made the System Development Kit (SDK) available for S5-and-above variants, the code was not opened to allow transfer of fingerprint input against intended purpose of storing and validating it on the device. The researcher thus used a simulation approach to mimic the authentication made with hybrid authentication of mobile banking users. A client server environment was then used. The database server was hosted in the cloud - representing the server room of the bank while the client signified the mobile device. A Domain Name was assigned for the purpose of unique identification. While a system with the application installed is

used to enroll, captured details are routed via the internet to the database server for secure storage. Ideally a phone with fingerprint facilities (such as iphone5 or Samsung S5 and higher versions) would have been used but for reason earlier mentioned, they were not opened for such experimental usage. In place of this, a system with a SecuGen Scanner attached and Internet facility was used to receive Phone number, PIN and fingerprint input. Then route same to the database server for matching.

3.2.1 Phase One of the System

The first phase of the design is for matching of the PIN. Matching of the PIN at the database level was carried out using binary search method to reduce time taken for searching of details.

3.2.2 Phase Two of the System

This segment of the proposed system handles the fused bimodal biometric features using image processing technique.

The steps to image processing were introduced into the design images/data set were captured and made to go through the various stages.

i. Capturing of sample dataset – Image acquisition stage

A total of fifty (50) mobile banking users were randomly selected from a first generation bank (First Bank Nigeria Limited). This bank was selected based on proximity and easy accessibility to the researcher. Mobile banking users were selected based on the experience they had gathered while using the existing solution. The users had each of their ten fingers captured. Ten fingers were selected for the purpose of back up whereas any of the fingers was sufficient to authenticate a user. Each enrolled user has his own distinct phone number and at liberty to select his 5-digit PIN. At enrolment also, other details captured are the names of individual users which may not necessarily be unique since there can be cases where names are the same. This does not affect the outcome.

ii. Image Pre processing

This is the process or stage at which the acquired images are made ready for features to be extracted from them. The preprocessing includes normalization which is a process that changes the range of the pixel intensity values. This is done in order to bring the images into a range that is more familiar or normal. It is also to reduce poor contrast or make images more robust for recognition. For the normalization of the fingerprint biometric Histogram equalization was adopted.

Given an image, I(x, y) either for face or finger biometric, the probability, P(i) is given by:

$$P(i) = n_i/N$$

Where I = 0, 1,2,…, k-1; n_i denotes the number of pixels in I(x, y)

With the grey level value of i, the mapping from a given intensity value i to a transformed one i_{new} is defined by

$$i_{new} = \sum n_i/N$$
$$= \sum P(i) \text{ where } i = 0, 1, 2, \ldots, k-1$$

The Algorithm for the normalisation

Input: noisy images of both fingerprint, I(x, y)

Output: normalized images, F(x, y)

Begin

1. Read in noisy images (raw), I(x, y)
2. For each (pixel_value in pixel_value)

   Position = (pixel_value + x)

   New value = int (position*y')

End.

iii. Image Segmentation

After both images have been normalized, segmentation was done. The goal of segmentation is to cluster pixels into salient image regions. That is, portioning of image into multiple regions (set of pixels) such that each region is homogenous. Divers algorithms exist that can be adopted for segmentation but this research considers the clustering algorithm which segments the images into clusters. Clustering is the process of partitioning a set of pattern vectors into subsets called clusters (Shapino and Stockman, 2000). This research employed K-means clustering algorithms for the clustering process.

The K-means is a simple method for estimating the mean (vectors) of a set of K groups. There are K clusters $C_1, C_2, \ldots, C_k$ with means $m_1, m_2, \ldots, m_k$

Least Square Error or Sum of Square Error which measures how close the data are to their assigned clusters is defined as:

$$D = \sum\sum||x_i - m_k||^2$$

for I = 1, 2, 3, …, k; $x_1 \epsilon C_k$

Algorithm for the K means clustering

Input: normalized images F(x, y)

Output: Clustered vectored images d(F(x, y), μ).

Begin

1. set ic (iteration count) to 1
2. Choose randomly a set of K means $m_1(1)$, $m_2(1)$, …, $m_k(1)$
3. For each vector x_i compute $D(x_i, m_k(ic))$; for each k = 1, 2, …, k and assign x_i to the cluster C_i with the nearest mean
4. Increment ic by 1 and update the means to get a new set $m_i(ic)$, $m_2(ic)$, …, $m_k(ic)$
5. Repeat steps 3 and 4 until $C_k(ic) = C_k(ic +1)$ for all k

End.

iv. Biometric Feature Extraction Stage

At this stage, the important features are extracted. These include ridges and valleys from the fingerprint. These factors are as binaries where grey regions are represented as zeros and the white regions as ones. The extraction is done using image processing extractor algorithms. There are various algorithms that can be used for the extraction but this research adopted the Principal Component Analysis (PCA). PCA is a useful statistical technique for recognition and image compression. Mathematically, PCA depends upon the eigen-decomposition of positive semi-definite matrices.

Algorithm for Feature Extraction using PCA

Input: Clustered images X = d(F(x, y), μ)

Output: Extracted vectors,

Minutiae of fingerprints and

Eigen vectors or Eigen faces of face biometrics V_i

Begin

1. Given X normalize to a linear vector image
2. Return $x = (x_1, x_2, x_3, \ldots, x_N)$
3. Calculate mean centred image

   $w_1 = x_1 - m$ where m is mean calculated and

   $m = 1/N \sum x_i \quad i = 1, 2, \ldots, N$
4. Matrix A of the mean centred vector is created

   $A = \{w_1, w_2, w_3, \ldots, w_N\}$
5. Compute the covariance matrix, C which is

   $C = A.A^T$
6. Compute the Eigenvector v_i of the covariance matrix
7. Compute the Eigenvector e_i of the transposed covariance matrix
8. Return feature vector $v_i = A\ e_i$

End.

v. Authentication/ Matching Stage

The system requests the user to present fingerprint for the system to confirm if the user is who he or she claims to be at this stage. Matching is now done by comparing the test or query fingerprint and face with that which has been extracted and stored in the cloud. For comparison, there are different algorithms too that are useful for matching arranged features and the arranged features and the test features and the test features for correlation.

For matching, this research adopts the Fourier Transform. It used specifically, the Fast Fourier Transform (FFT) because of its improvement over other Fourier Transform algorithms. FFT is a discrete Fourier Transform algorithm which reduces the number of computations needed for N points from $2N^2$ to 2N1g N where 1g is the base 2 algorithm. FFT algorithm computes the Discrete Fourier Transform (DFT) of a sequence or its inverse. Fourier analysis converts a signal from its original domain to a representation in the frequency domain. The DFT for T is defined thus:

Let $x_0, x_1, \ldots, x_{N-1}$ be complex numbers

$X_k = \sum x_n e^{-i2\pi kn/N} \qquad n,k = 0, 1, \ldots, N-1$

The Matching Algorithm

Input: v_i are Eigen vectors

M_{Ax} minutiae given A(x) of degree ≤ n-1where n is a power of 2ω

Output: True Match fingerprint.

Begin

1. if ω = 1, return A(1)
2. express A(x) in the form A_e(x^2) and A_O(x^2) where A_e are the even powers and A_O are the odds.
3. call FFT(A_e, ω^2) to evaluate A_e at even powers of N
4. call FFT(A_O, ω^2) to evaluate A_O at even powers of N
5. for j = 0 t0 n-1

Compute A(ω^j) = A_e(ω^(2j)) + ω^j(A_O(ω^(2j)))

6. return A(ω^O), …, A^(n-1)

End

3.3 **Overview of the System**

The main focus of this research was to design and implement an authentication model that leverages on TDBA (fig 3.1) model using both PIN and bimodal biometrics as means of authentication. The codes were written in Java language. The entire hybrid authentication process involves capturing of fingerprints using SecuGen Hamster due to its high-performance and maintenance-free optical sensor for fingerprints. SecuGen scanner is designed in a way that allows for vertical placement of fingers. Therefore, two impressions per finger were made. Sample impressions are shown in fig 3.2. Irrespective of which of the two impressions is adopted, the essential ridges are obtained and a fingerprint database is created. Matching is then done by comparing the ridges of the finger being scanned with the ones stored in the database using a binary search method. Since fingerprint involves supplying input by touching, this study will also leverage on the Touch Dynamics-Based Authentication (TDBA) technology (fig 3.1) to develop a hybrid model that is focused on input received into the system by touching (fig 3.4).

The designed system is in two phases: the training phase and the testing phase (fig 3.3).

i. Training Phase

This is the phase where customers or potential users of the mobile banking app are made to register their details and necessary information pertaining to them are requested for including fingerprints. The information are then kept in the cloud storage together with their account details. As each person's data is captured, it goes through all the image processing phases and then kept in the cloud.

ii. Testing Phase

It is the second phase after implementation where a test image was used to compare with what has been stored in the cloud. Fingerprint as well as PIN of a customer were matched against what has been kept in the cloud for identification and authentication.

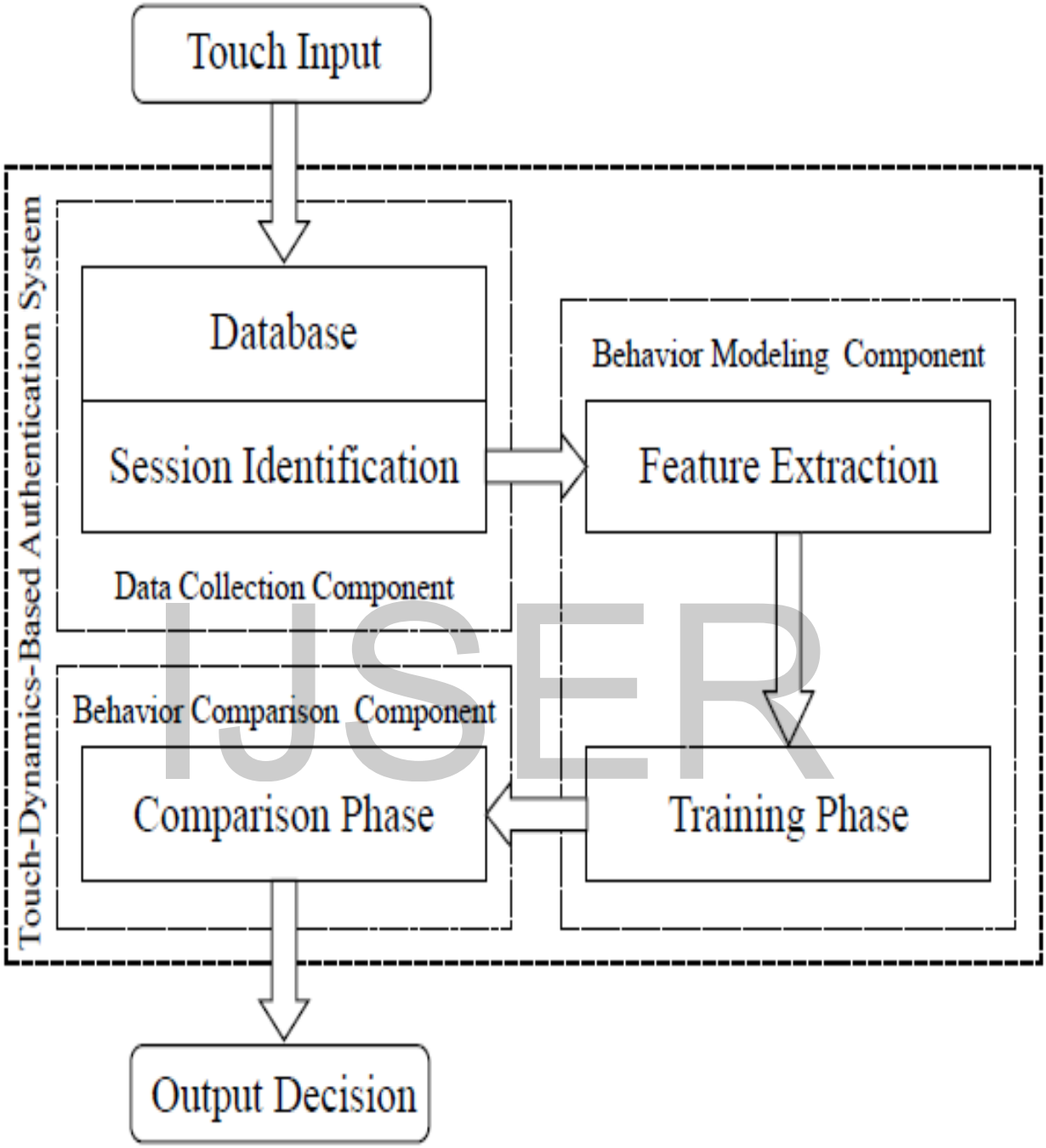

Fig 3.1: Touch dynamics based authentication (TDBA) technology

(Source: Meng 2013)

Fig 3.2: Impressions of a finger and corresponding ridges

(Source: Chen and Gao, 2007)

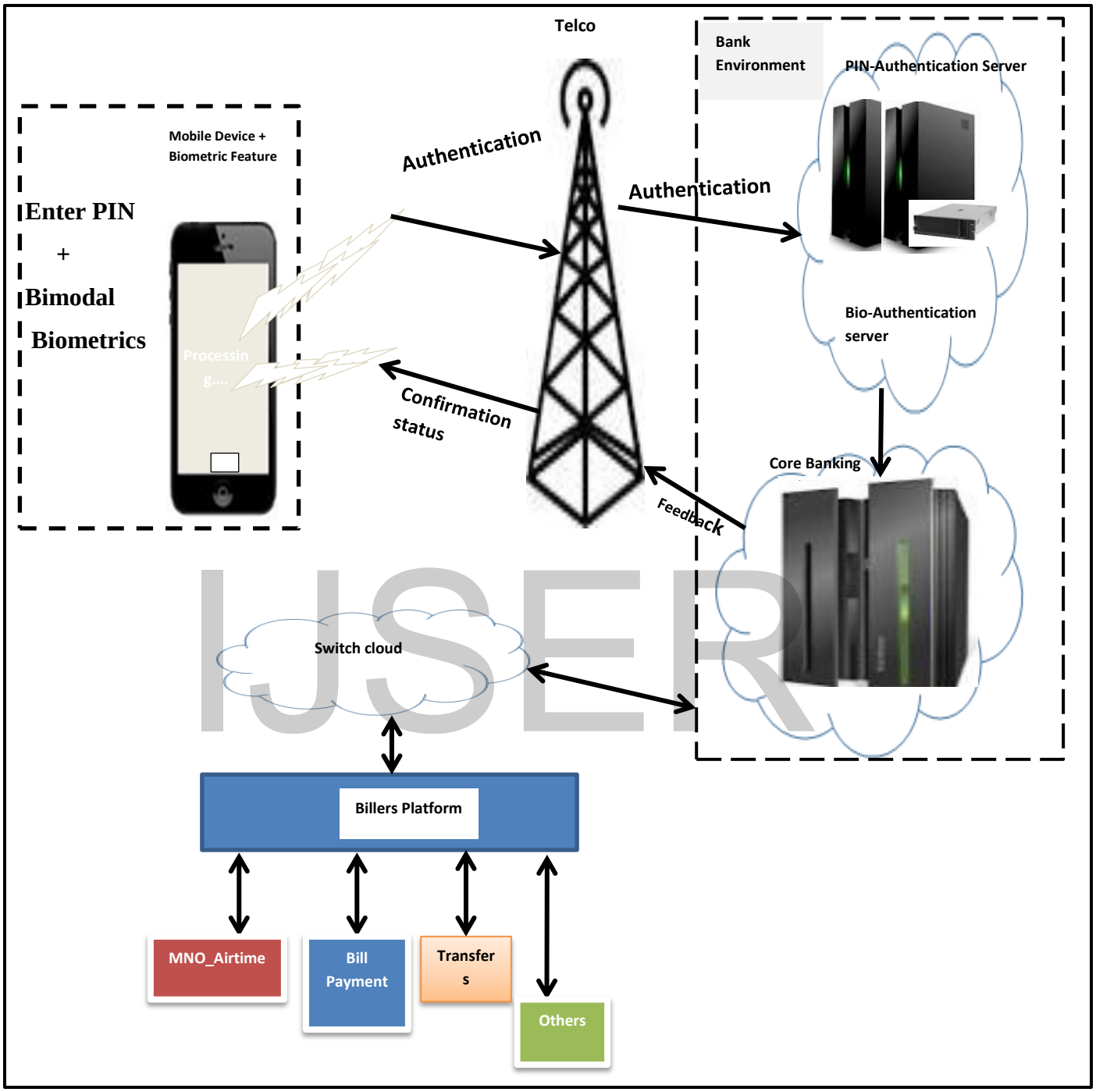

Fig 3.3: New Hybrid Authentication Architecture

Users' acceptability of the new system was also determined. This was done by reviewing the False Acceptance Rate (FAR) and False Rejection Rate.

The existing control over mobile banking requires only PIN to authenticate transaction. The PIN is then routed through the internet to an authentication server to confirm genuineness or otherwise. The hybrid authentication solution required both the PIN and the fingerprint to be supplied and routed through the internet to the authentication server. Some routing authentication standard, capable of adding the biometric authentication to the existing infrastructure can be used to achieve this. Example of such is the BioAPI (from Entrust). Ten fingers of both hands are enrolled at the capturing or enrolment stage. Any of the ten pre-enrolled fingers of both hands is sufficient for authentication. The supplied input is matched with the stored data in the Authentication Server using binary search algorithm to verify the authenticity of input.

In real life scenario, a mobile device with biometric feature (such as Apple's iphone5, Samsung S5 or higher versions) is preferably used to receive input from user and matched against already stored biometric details linked to the customer's account number. This is because this type of devices has facilities to receive fingerprint input. For this research, a sample of users was taken to try their fingerprints on the new solution. The resultant output was evaluated to determine the False Rejection Rate and False Acceptance Rate. Conclusion was then drawn from the Rates to decide on the effectiveness or otherwise of the new solution.

3.3.1 The Model: Hybrid authentication model (HAM)

The hybrid authentication model (HAM) used a combination of both the PIN and biometrics for authentication. The transaction to be carried out would be selected from the application. The customer was challenged to supply the Phone number, then the PIN as well as the fingerprint. Each of this was verified. Where the supplied PIN or Phone number or fingerprint was incorrect, an error message was displayed. Where they all match, the transaction proceeded by checking the balance and updating relevant accounts.

3.3.2 Touch Dynamics Based Authentication (TDBA) Technology

This study required a technology model that can safeguard user's identity and user friendly. To guarantee these attributes, I leveraged on TDBA technology to develop a more resilient model. The TDBA model best suits physical contact with the device by touching (Meng 2013). It allows users to be identified through IMEI, authenticated by PIN, carry out transactions and authenticate such transactions by fingerprint. Fig 3.5 shows physical contact by touching using TDBA technology.

3.3.3 Possible Hindrances to fingerprint enrolment

The following may affect fingerprint enrolment and should be taken into consideration to ensure smooth capturing:

Imaging conditions. Sensor should be clean before capturing for optimal result

Finger condition. The fingers should be dry and clean to prevent any form of distortion.

Finger placement. At the capturing stage, each finger should be properly placed.

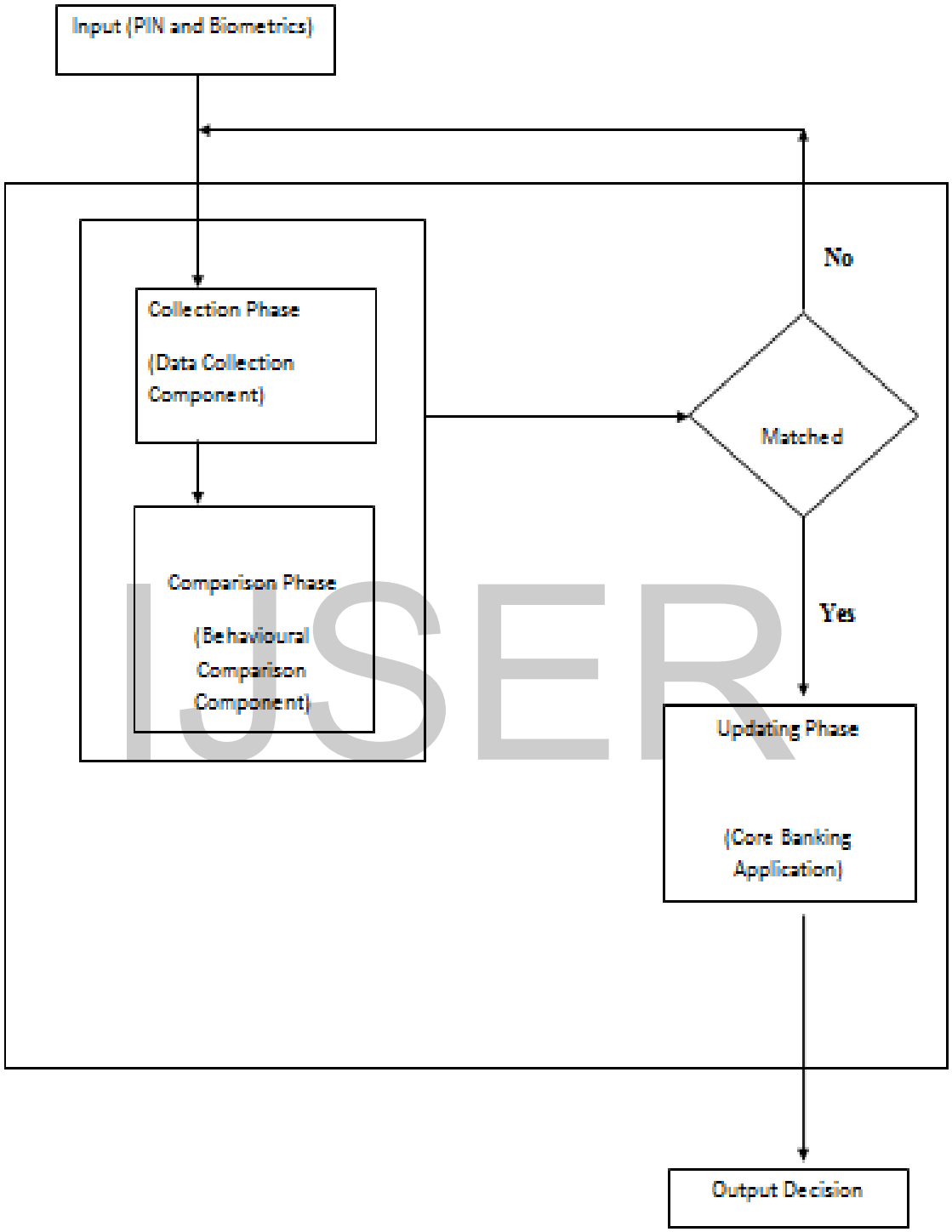

Fig 3.4 New Hybrid Authentication Model (Flowchart)

3.3/3 Components of the HAM

Input phase. The Phone number, PIN and the biometrics are supplied to the system at this phase. They must be correct and previously supplied to at the enrolment stage. The phase is very important to the integrity of the data. Any wrong capturing at the initial stage would lead to wrong denial of authentic user or data mismatch. Organisations always prefer to restrict physical and logical access while performing this task. All things being equal, supplied data (Phone number, PIN and Fingerprints) are allowed to the next phase called the collection phase.

Collection phase. This is a transient phase between the input and the matching phases. Phone number, PIN and the fingerprint received from the input phase are sent to the comparison phase for correctness or otherwise of each, one after the other.

Comparison phase. At this phase, orderly arranged data in the database are compared with the collected data. The Collected PIN is compared with the stored PIN and then the fingerprint in a similar manner. The searches become simplified because the SIM serves as the unique identifier to narrow down the search. In other words, mobile devices are identified, validated with PIN then further authenticated by fingerprint.

Updating phase. This phase occurs at the bank's core banking applications end. It does not take place until due verification has taken place. The verifications include confirming the existence of the phone number in the database, matching of PIN and fingerprint and financial position of the customer's balance. Where all matching is done and the customer has sufficient balance to accommodate the transaction amount, his account is immediately debited before going to credit the biller or merchant. This is in line with basic accounting principle of **"debit and credit."** Where there is insufficient balance, despite passing the comparison phase, value is denied.

Output phase. This is the last phase. It can also be called the feedback phase. This phase relays to the user of the outcome from input to the updating phases. The feedback includes the success or failure of the input, matching and updated position.

3.4 Cloud Storage

A typical cloud storage system consists of a Master Central Server and several storage servers. There are hundreds of different cloud storage systems. Some have specific usage such as for email, pictures or for storing digital data. There are different cloud storage service providers. Some known names include Dropbox, Sugarsync, Database Mart and a host of others. Storing data into cloud involves different stages which also considered in this research. The first step is to choose a provider that suits the need at the moment. The next step is to make one's data stored in the cloud.

Factors considered before choosing the storage provider:

i. appropriateness of security standards

ii. size of data to be stored

iii. data encryption at uploading stage, downloading stage and in stored state

iv. access sharing with cloud folder.

v. available options should the cloud fail or hacked into

Steps considered for securing stored data

i. Password: cloud services required a master password to get into a stored file. So, a strong password with a combination of alphanumeric and special characters was used. Other password policies were also adhered to such as: enforce period change, disallow re-use within a 12-month-cycle and discouraging password sharing.

ii. Back up: the service provider also has adequate backup/restore process in place.

Having considered all these, this research chose a public cloud storage provider – Database Mart – for it met the necessary requirements and it is affordable.

3.5 Design

The design is cloud-based. The hybrid system has two main modules for capturing and for matching. While the former is used by the Administrative (Admin) user who enrolls other users on the system, the latter is for normal users who carry out transactions requiring authentication.

The research decided to host the database server in the cloud to represent bank's datacenter where the database is kept.

In a real life scenario, the Admin user will be a designated staff of the bank. Necessary controls would be effected to prevent any security lapse. For instance, the enrolment must be done in a restricted arena to ensure the confidentiality and integrity of data is preserved. The duties should also be properly segregated to ensure to no single individual can single-handedly complete an enrolment process. He is also the custodian of his login details.

3.5.1 Admin User's Activities

The Admin user has specific tasks to perform on the new system that begins with his access to the system through to when he finishes capturing the entire ten fingers of each normal user. This is further broken down in the following subsections.

3.5.1.1 Capturing or Enrolment stage

At this stage, the Admin user accesses the system by clicking the Admin button (Fig 3.5) to allow him access to capture the details of normal users. In real life, the doctrine of least privilege is applied to ensure that a user does not have more access rights than required to perform his assigned or designated task. His access is also protected by his own login credentials.

3.5.1.2 Admin User Menu

The Admin user is presented with the Admin user menu (Fig 3.6) the moment he accesses the new system. He has the facility to select "Registration" which enables him to capture details of normal users or exits the application for one reason or the other. At go-live, the Admin user menu may include other menu such as password maintenance. This is to enable him change his password at will especially, when a compromise is suspected.

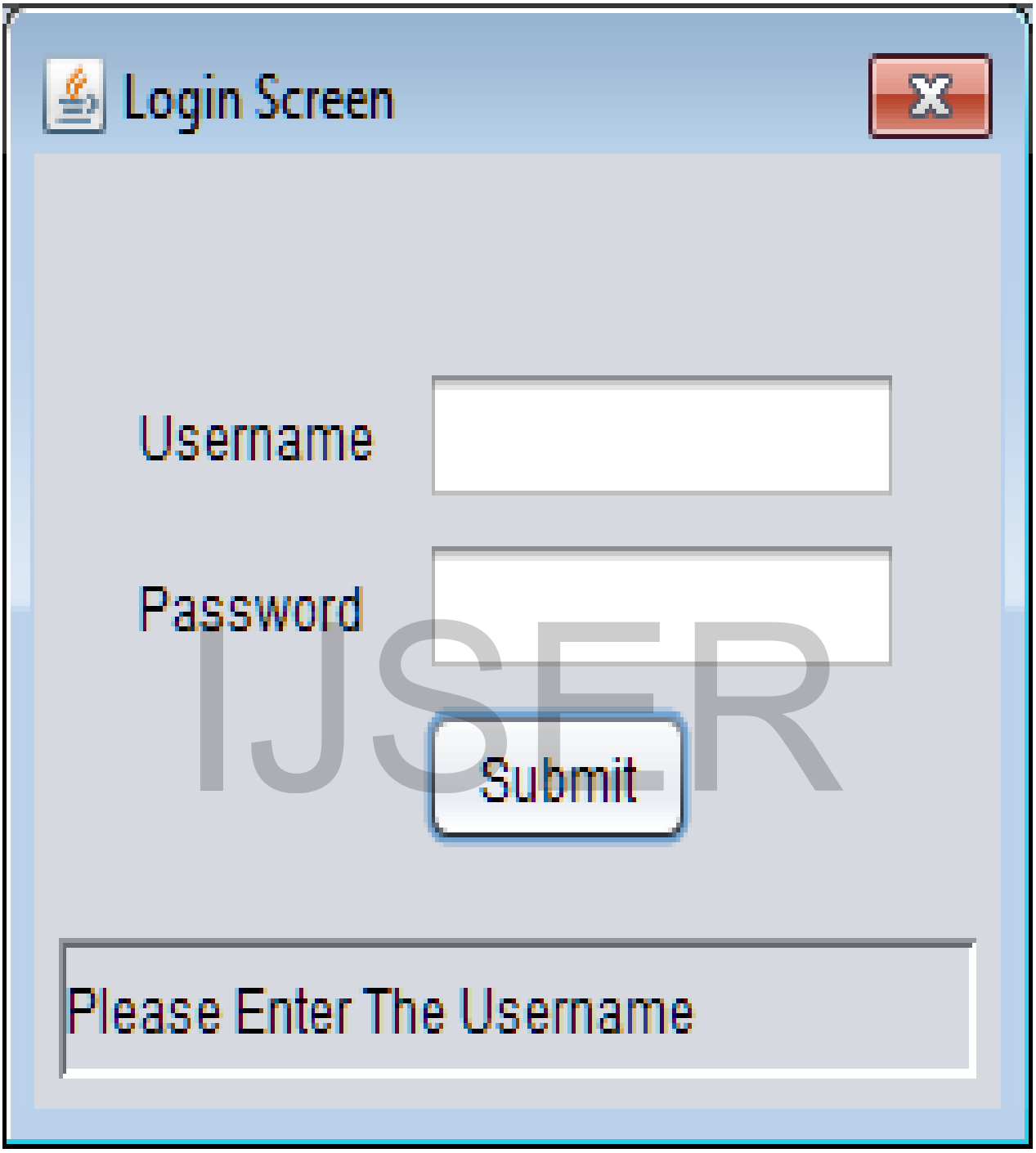

Fig 3.5 New Login Window for Admin user

Fig 3.6 New Admin User Module

3.5.1.3 Bio-data and Fingerprint Details Capturing

The Admin user enrols the bio-data of a normal user. Details to enrol include: First Name, Middle Name (optional), Last Name, Phone number and PIN. The PIN is entered twice for the purpose of confirmation (fig 3.7). After entering these details, the "Init" button is pressed to initialise and prepare the new system for scanning the ten fingers one after the other. "Enrol" button is pressed after scanning each finger to ensure the scanned finger is stored and also the system to receive the next finger.

3.5.1.4 Submission of Captured Details

The process of scanning the fingers is iteratively done until the ten fingers have been captured. The "Submit" button is then pressed to store the details in the database (Fig 3.8). Each finger is captured twice and the system immediately compare the two instances before enrolling them. This to reduce cases of mismatch. The moment the ten fingers have been captured, the new system does not allow for additional ones.

3.5.2 Normal User's Activities

A normal user represents a mobile banking user who has mobile device the installed mobile banking application. Here, he accesses the new system by supplying his phone number and PIN which were pre-enrolled with the Admin user (fig 3.9) and presses the "OK" button. Validation is carried out on each of these two details supplied. For the purpose of experiment, a feedback is allowed to know which of the details is wrong (fig 3.10 and 3.11). In practice, this is discouraged in order to avoid giving a fraudster a lead.

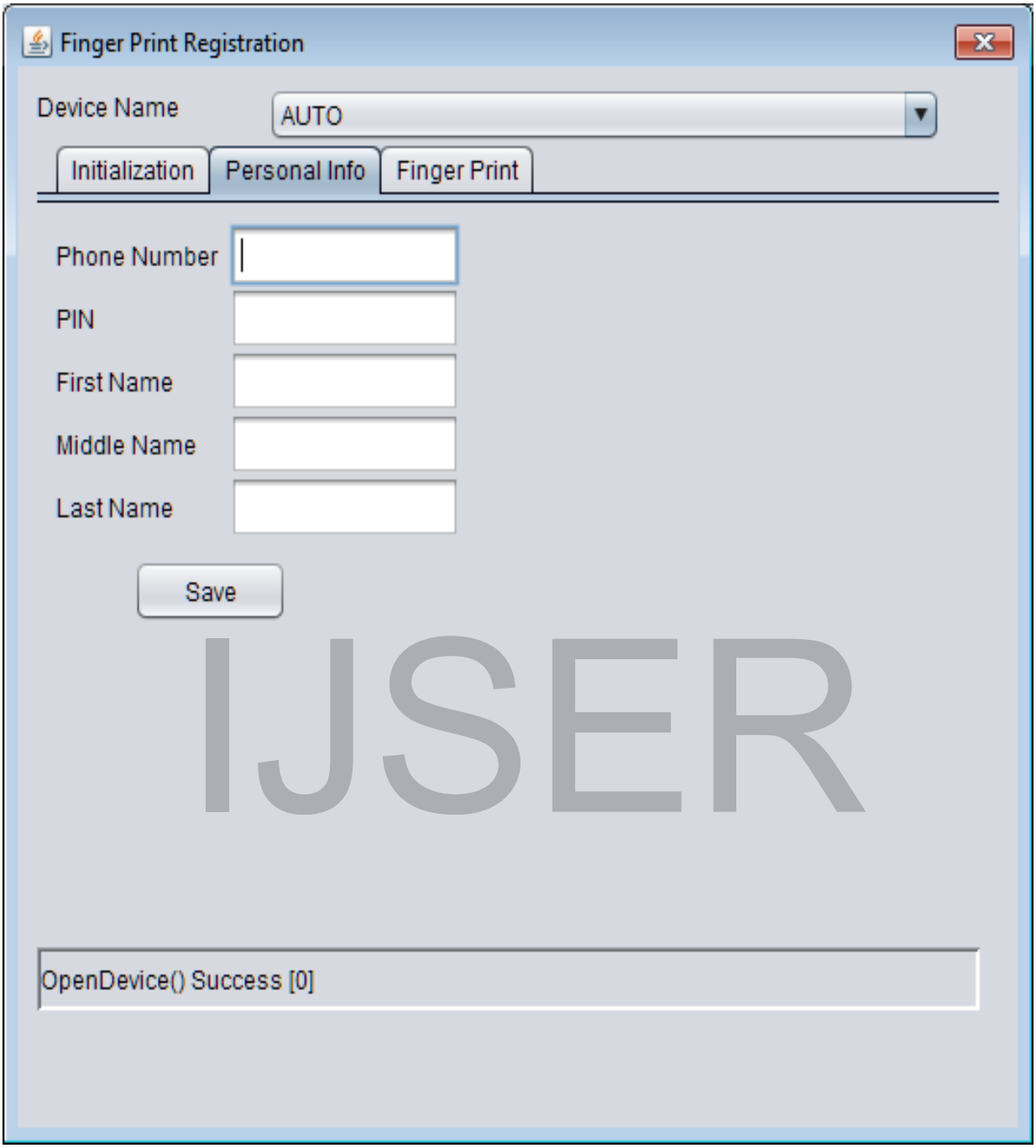

Fig 3.7 Capturing Details on the New System

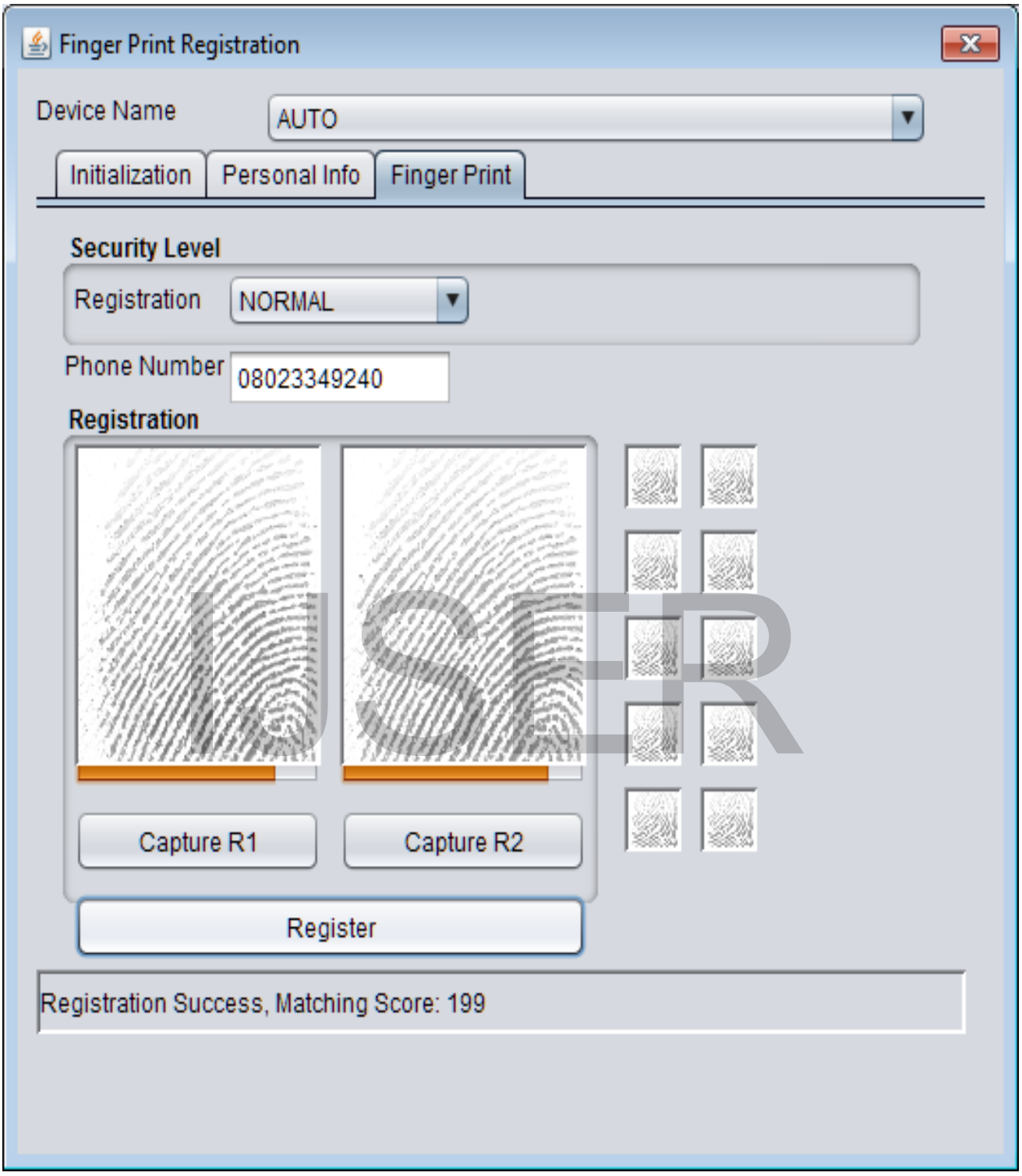

Fig 3.8 Saving Captured Details on the New System

Fig 3.9 Normal User Login Page on the New System

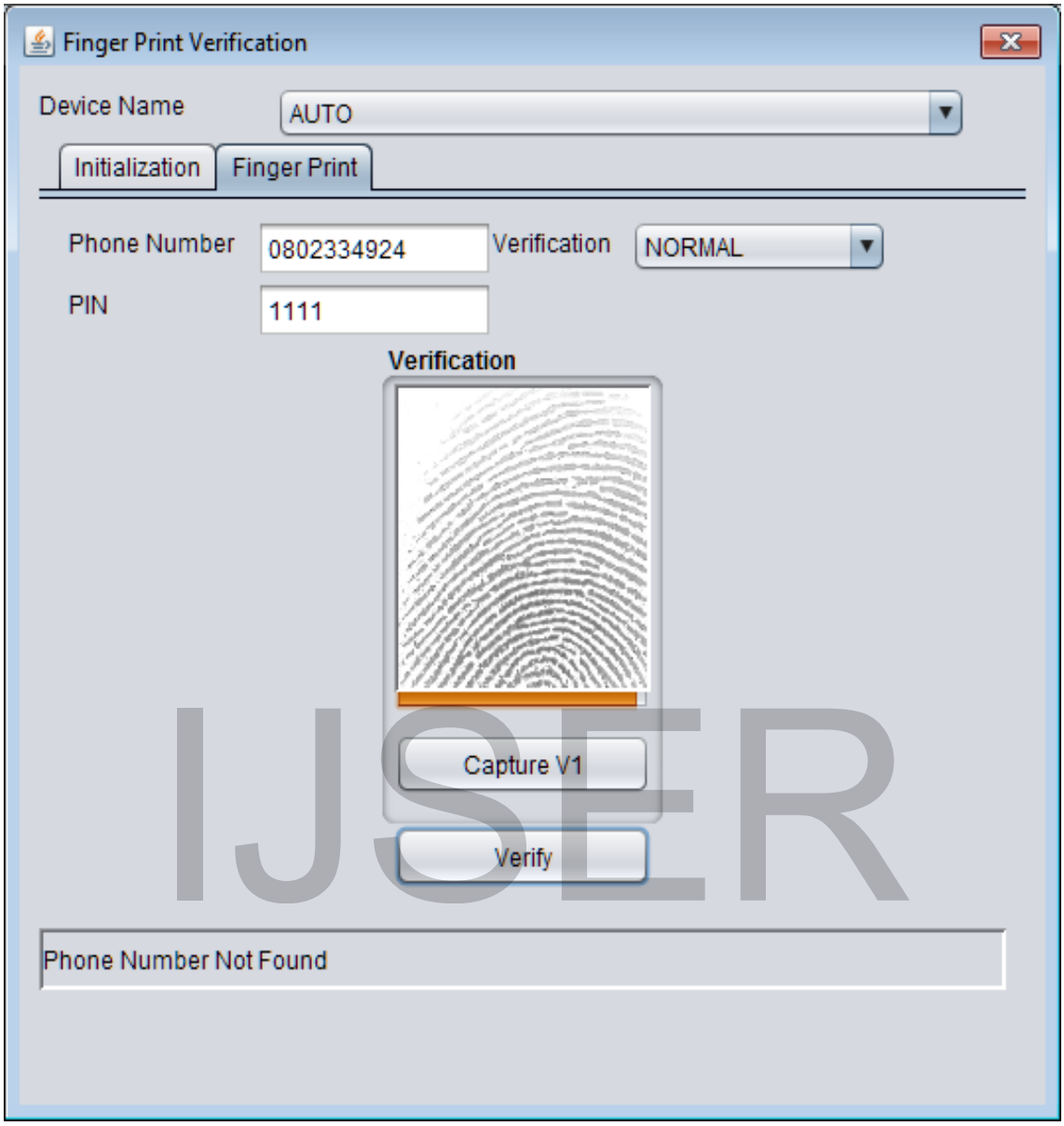

Fig 3.10 Wrong Phone Number Supplied to the New System

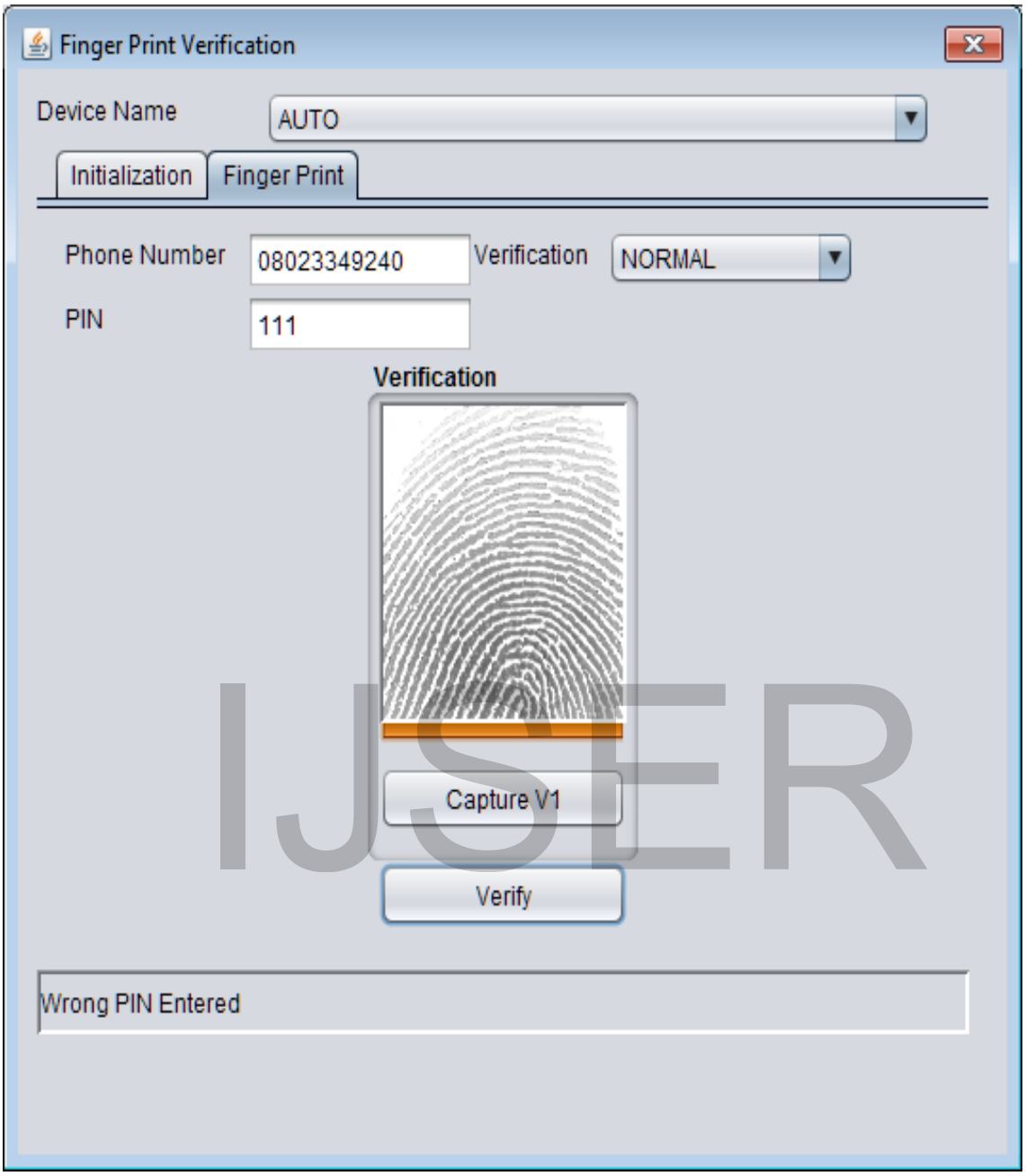

Fig 3.11 Wrong PIN Supplied to the New System

3.5.3 Fingerprint Verification Menu

When both the phone number and the password supplied are correct, the new system prompts for fingerprint to be supplied. This process begins with the initialization for the system's readiness to receive the finger print (fig 3.12).

3.5.3.1 Fingerprint Verification

The fingerprint, which is the additional authentication, is verified by pressing the "verify" button (Fig 3.13). The verification or authentication process takes the supplied input and compares it with the stored fingerprint in the database. Any of the fingers can conveniently be used to authenticate.

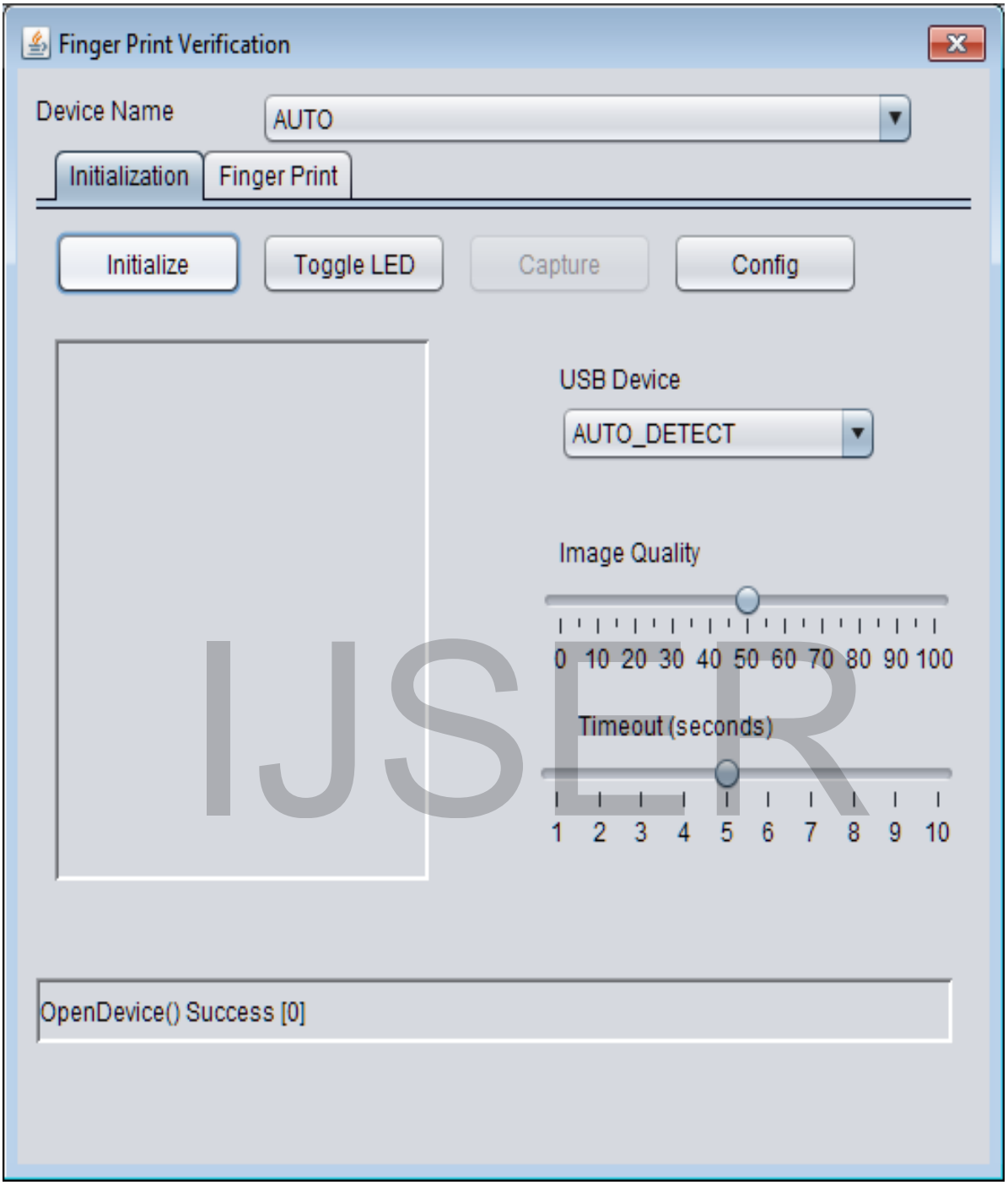

Fig 3.12 Initialization Process on the New System

Fig 3.13 Verification on the New System

3.5.2.2 Failed Fingerprint Authentication

In the process of verifying the fingerprint, the input (finger to be verified) is supplied. The input is then compared with the scanned and stored fingerprints in the database. The search is however narrowed down because the phone number with the PIN had been verified and found to truly exist. Further confirmation, to ascertain the ownership of the correct PIN and phone number presented, is done with the fingerprint matching. Where it fails, an error message is flagged (Fig 3.14). At go-live, the displayed message can always be managed to suit the bank's policy without given an unnecessary lead to an intruder. In most cases, the unsuccessful attempts are also tracked by the internal control mechanism and then analyzed to obtain certain useful details that can aid profiling as well as decision making.

3.5.2.3 Successful Fingerprint Authentication

Where fingerprint input finds a matched in the database, access is granted to do any of the listed mobile banking transactions (Fig 3.15).

Fig 3.14 Failed Authentication on the New System

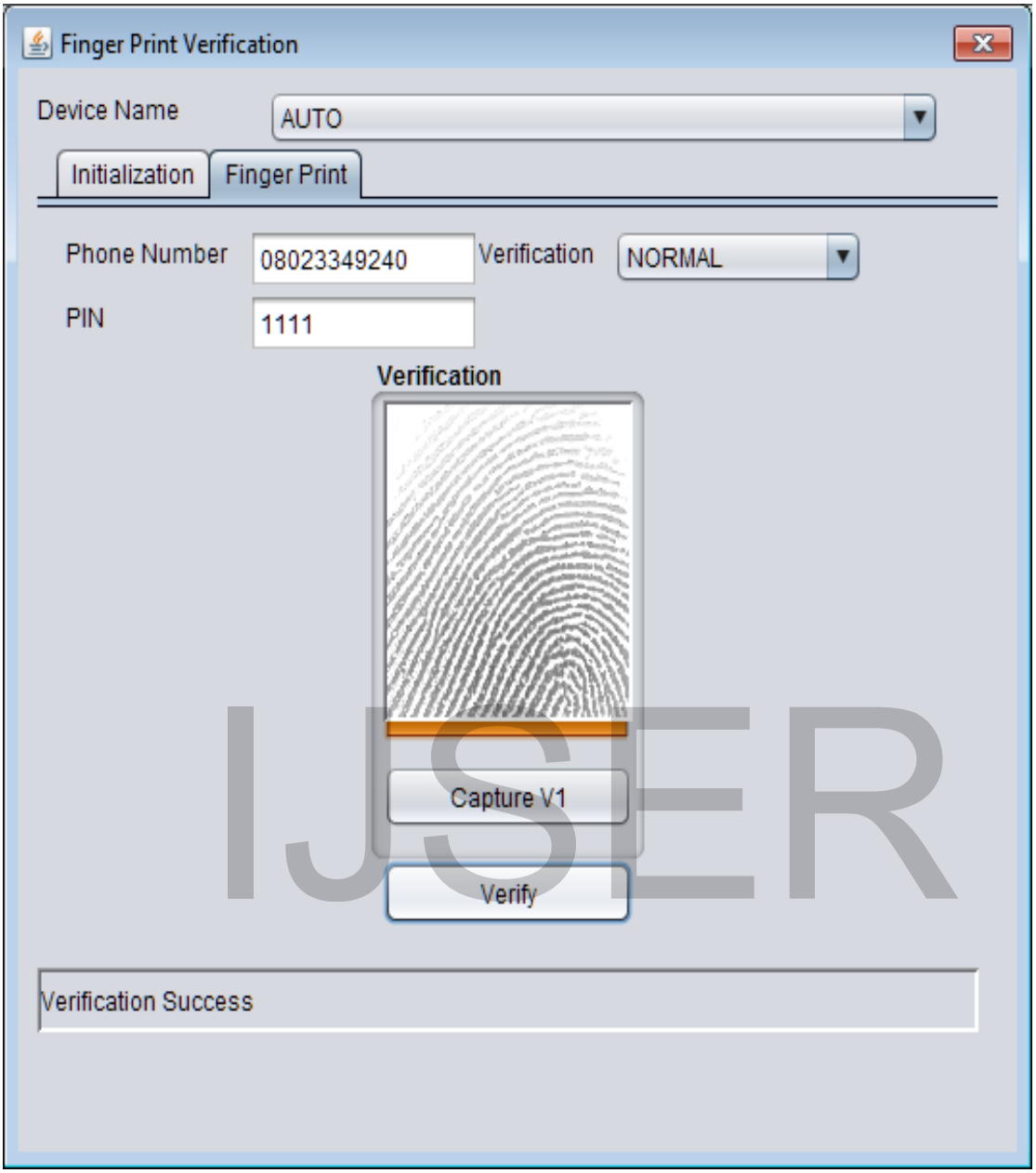

Fig 3.15 Access to Mobile Banking Transaction on the New System

3.6 Identification and Authentication (I & A)

Identification and Authentication represent two terms that are often used together. They describe the initial phases of gaining access to a system. While identification is the process of availing an identity to the system, authentication is the process of verifying that the user is who he claims to be. I & A have been of tremendous use and the concept is used to secure many applications and platform. The ultimate goal is to be sure that unauthorized users are denied access to the system resources while granting access to legitimate ones. Mostly, unique user identities (employee staff numbers) are used for identification purposes. They are called user ids. Confirmation to show that the person whose id is supplied represents the right user, he is prompted to add the authentication details. The authentication details can be a password (something you know), a biometric input (something you are), a PIN (something you know) or a Token (something you have). Where it involves only the id and any of the authentication details, it is termed a single factor authentication. It is called a two-factor authentication if it involves any two of the authentication details.

3.7 Benefits of Biometric Authentication

There are various benefits derivable from biometric authentication especially when compared with other means of authentication. These include:

Cost. Biometric system that uses smart phones to receive fingerprint input does not require additional cost. Overhead cost of password reset is taken off.

High security. Since no two individuals have exactly the same biometric features, it reduces chances of frauds.

Anti-spoofing measures. Various forms of anti-spoofing tactics are employed thereby making it difficult to perpetrate frauds. For instance, heat measurement or electrical conductivity is used in sophisticated readers to be sure the finger is live. Blinking is also ensured for facial biometrics to detect if a picture is used.

Readily available. Biometric features are always readily available unlike token that needs synchronization and re-synchronization or PIN that can be forgotten.

High Accuracy. Biometrics usage as means of authentication have been quite impressive when used to grant access to facilities. This is because most of them have low FAR.

Anti-shoulder surfing. Careless and uneducated users have fallen victims of shoulder surfing while keying in their PIN. Biometric input method has helped to take care of this class of users since it cannot be imitated.

Ease of usage. Biometric input does not involve committing anything to memory. The user just presents himself after the initial enrolment.

3.8 Fingerprint Capturing

The Admin user logs on to the new application with his credentials and then enrolls the normal users by taking the bio-data and then their fingerprints. The enrolment stage is very critical to effectiveness of the new system because if wrong details are stored against wrong user, it can lead to false denial of an authentic user or false acceptance of an illegitimate user. In most cases, the Admin user uses a restricted area that cannot be easily accessed either physically or logically. For this research, the ten fingers were captured (for the purpose of providing back up should one or more fingers be damaged) and stored in the database that can be logically accessed by normal user at fingerprint verification.

3.9 Testing of the Solution

This research adopted a simulation approach due to challenges encountered with the parties involved (Telecommunication companies, mobile devices manufacturers and banks). They were not favourably disposed to opening up their platforms for such experimental works of this nature due to security reasons. Beside this, simulation approach was helpful because it is cost effective.

Banks. Banks could not allow the testing for the fear of exposing identified weakness in the financial institutions to the outside to world to capitalize on. In place of this, a database server was hosted in the cloud. It was remotely accessed by a client.

Mobile Device Manufacturers. These are companies producing mobile devices such as Samsung, Apple, Techno among others. The phones with fingerprint facility were either not making their SDK readily available or makes it extremely difficult to open the SDK to route the fingerprint input from the device to a distant server. A client was made to connect to the database

server via the internet while a SecuGen Hamster was connected to the client to supply the fingerprint input.

Telecommunication Companies. Telecommunication companies (Telcos) are usually involved in routing mobile banking transactions through their platforms since each SIM belongs to a Telco. In order to uniquely identify a device, application was designed to include phone number at the capturing stage.

3.10 Pre-Testing of Application

The application was tested via a localhost. A SecuGen Hamster was connected to a system. Several users were enrolled and subsequently tested for matching. It was discovered that of the 5 samples randomly selected the False Acceptance Rate (FAR) was 0 while False Rejection Rate (FRR) was also 0. This gave some level of confidence that the expected solution could be met.

3.11 Characteristics of Sample

The population was made up of staff and customers of FirstBank Karu Branch, Abuja, Nigeria. This is due to the proximity reason for the researcher.

3.12 Method of data analysis

In determining the error rate, various scenarios were considered. These were tested based on PIN, SIM and fingerprint availability. FAR and FRR are then obtained and analyzed to determine the reliability of the new solution.

3.12.1 PIN Testing – PIN testing was done to represent a situation whereby the PIN was guessed. As long as the right PIN was not obtained, the solution would deny access to the intruder even when the mobile device is stolen. However, fraudsters have made frantic efforts to obtain PIN through social engineering, shoulder surfing or through cracking (on rare occasions). This was the reason for the hybrid solution that this work proffered.

3.11.2 SIM Testing – Phone number was used for the test to stand for the mobile device. When a wrong phone number was supplied to the solution, access was denied. In a real life situation, a

mobile device can be stolen even with the mobile application installed, access would still not be granted when the registered SIM is missing.

3.12.3 Fingerprint Testing – Ten fingers of both hands were captured although any of them could be sufficient for validation. The whole essence of capturing all fingers was for necessary backups should there be a damage done any of the fingers.

3.12.4 PIN verification

The existing mobile banking solution uses the 5-digit PIN which can be supplied through the keypad of the mobile device in use and routed to the bank's server for verification. If it is successful, the user is allowed to carry out his transaction. Otherwise, access is declined. The new Hybrid solution leverages on this but added another layer of security to further strengthen the control. This is done such that even in the event that the mobile device a mobile banking user is stolen and the PIN is also obtained, frauds can be prevented.

3.13 Verification of Phone Number and the Fingerprint

Both the phone number and the Fingerprints are entered and stored away in the database for use at the verification stage. The Hybrid solution has the capability to efficiently search the phone number from the pool. This is similar to the Primary Account Number (PAN) used in cards. Where the phone number exists on the database, it narrows down the search and only awaits authentications of PIN and Fingerprint. Each phone number has the following attributes:

1. They are distinct. No duplicate exists. The phone numbers are uniquely assigned to users and hence, no overlap.
2. Each phone number is linked to owners account numbers that are also uniquely created.

The stored phone numbers are orderly arranged in the database from the least to the highest.. the middle phone number is selected from the database. The middle number divides the stored phone numbers into two say, the upper and the lower halves. The middle number thus becomes the biggest number for the lower half. Comparison is done between the lowest or the first number and the highest or last number in the lower half. Where it matches, then, the phone number is pre-enrolled and exists in the database. Where it is smaller than the lowest number, then input

does not exist on the database. Where it is greater than the highest number in the first half, then the second half is considered and similar process is undergone. The following scenarios play out:

Scenario 1 – Phone Number not existing in the database

If number does not exist in the second half having tried the first, then a conclusion is drawn that the phone number is not captured which is the reason for the non-match. There is no reason to proceed further to request for PIN or fingerprint. It may imply that an intruder wants to use a stolen detail such as stole PIN and or device with installed mobile banking application to perform some irregularities.

Scenario 2 – Phone Number is found in the database

If the phone number exists, the solution then request for PIN in order to confirm the identified phone number. This is similar to the user name and password access method. When the supplied PIN is incorrect, the solution disallows further activities. This may represent a situation whereby the mobile device is lost with the SIM inside but the fraudster does not have the PIN. However, where phone number exists and the correct PIN is used the solution requires a fingerprint to authenticate the user. This authentication process ascertains that the user that is found to exist (through the phone number matched with the captured data in database) and identified (with the correct PIN) is the authentic user by requesting for the fingerprint.

3.14 Normalization

Normalization is the process of organizing the attributes and tables of a database so as to minimize data redundancy. It involves refactoring a table into less redundant without losing any information. Two tables (details_tab and finger_tab) are involved in this research. They are related by the phone number which assisted in reducing the redundancy.

3.15 Relationship between Model and Matching

Table 3.1: Tabulated results from the new system

"v" REPRESENTS CORRECT DETAILS

"×" REPRESENTS WRONG PARAMETER

PIN	SIM	FINGERPRINT	NO OF ATTEMPT	NO OF SUCCESS	NO OF FAILURE
v	v	V	50	49	1
×	v	V	50	0	50
v	v	×	50	0	50
×	×	V	50	0	50
×	×	×	50	0	50
v	×	×	50	0	50
v	×	V	50	0	50
×	v	×	50	0	50
Sum			400	49	351
Mean			50	6.125	43.875
Standard Deviation			0	4.0256	4.0256

Sum = $\sum X$

Mean = $(\sum X)/N$

Standard Deviation = $(\sum X^2/N)^{1/2}$

Optimal performance of the solution is assumed to be 100%. This is set as the benchmark against when the proposed solution is measured. Mean square error was used to test the efficiency or otherwise of the solution. A solution is termed efficient when it is unbiased and has a minimum variance when compared another unbiased estimator.

The success or failure rates were measured upon the introduction of the mix of variables of PIN, SIM and Fingerprint. Instances of introducing the correct details of the three variables or various mixes of wrong details were tested to estimate the success and failure rate. These are directly linked the error rate.

3.15.1 Estimates for Number of Success

Using Mean Square Error (MSE),

$MSE = Var(\grave{u}) + Bias^2(\grave{u})$……(3.14.1)

Where ù is the bias of the estimator defined as the expected value (of the estimator) less the mean "μ" being estimated;

Var (ù) is the variance.

Thus, $Bias = E(\grave{u}) - \mu$…..……..(3.14.2)

An estimator is unbias if it has a zero bias, otherwise it is biased.

$Bias = E(6.125) – 6.125$

$= 0$

Substitute for Bias in equation 3.14.1:

$MSE = Var(\grave{u}) + Bias^2(\grave{u})$

Hence, $MSE = 4.0256 + 0^2$

$= 4.056$

3.15.2 Estimates for Number of Failure

From equation (3.14.2),

Bias = E(ὺ) - μ

Hence, Bias = E(43.845) – 43.845

= 0

Using equation (3.14.1),

MSE = Var (ὺ) + $Bias^2$(ὺ)

Therefore, MSE = 4.056 + 0^2

= 4.056

Since the solution has zero bias and has the least variance value, the solution is efficient.

3.16 Algorithm Development.

Declarations

Step 1: Initialize whole numbers F_R, N

Step 2: Let N be taken between 1 and 10

Step 3: Set N^{th} finger to F_{RN}

Step 4: Input fingers and store in F_R

Step 5: Set the length of the fingers F_R, of a normal user to 10

Step 6: Set F_{R1} to the first image in F_R

Step 7: Set $F_{R2}$to the matched finger on the rest of F_R

Step 8: If F_{R2} matches, set F_{R1} to matched finger.

Step 9: Otherwise, return F_{R2}

Step 10: If finger matches F_{R2}, set F_{R2} to matched finger

Step 11: Otherwise, return match not found

Step 12: using equation 3.16.2,

Bias = $E(\grave{u}) - \mu$

Step 13: Then substitute for Bias in equation 3.16.1:

MSE = Var $(\grave{u}) + [E(\grave{u}) - \mu]^2$

Step 14: Display MSE for step 9

Step 15: Display MSE for step 11

3.17 Comparison of the PIN-Authentication Mobile Banking (PAMB) with the Hybrid Authentication Mobile Banking (HAMB)

There are major differences between PAMB and HAMB which basically revolve around security. These are hereunder stated:

3.17.1 Cloning. This involves copying basic features of mobile device for fraudulent usage. With respect to mobile banking transactions, there are basic threats that can make mobile banking vulnerable to attacks. Mobile Marketing Association (2009) stressed that Mobile banking are susceptible to the following: cloning, hijacking, malicious code, malware, man-in-the-middle-attack, redirecting, SMiShing, spoofing and vishing. Most issues highlighted by Mobile Marketing Association are preventable through proper user education and strong encryption. Rajendran (2014) stated that there were more than 20,000 registered cases of illegal SIM cloning in Delhi and Punjab yet none of the major telecom operators can immediately track existence of duplicate SIM cards. Thus, cloning of SIM details remains a major problem that the existing PAMB has not been able to solve. This describes a situation where a SIM number is cloned and used as if it emanated from the victim. However, HAMB adequately takes care of cloning. Even when cloning occurred, further authentication of fingerprint input will foil the fraud. This was clearly illustrated in this research.

3.17.2 PIN Compromise. PIN compromise describes a process of fraudulently obtaining the five-digit number being used to consummate a transaction. This could be achieved in divers ways ranging from non-technical to technical such as social engineering, mere guesing, should surfing and PIN cracking. When PIN is obtained, a fraudster may have the chance of committing frauds thereby making PAMB highly insecure whereas HAMB has effectively taken care of this by requesting for fingerprint for further authentication.

3.17.3 Loss/Stealing of Mobile Device. These days, mobile devices are used for some activities other than making calls or just carrying out financial activities. Some personal details are equally stored. Some schools of thoughts believe that there is no use memorizing anything (including password or PIN) when it can be safely stored. Loss of such a device can expose the owner to frauds before effort can be made to block. This is only possible with PAMB. It is a major difference between PAMB and HAMB. The device can afford to stay for as long as he wills before notifying his financial institution of such a loss or theft and his account remains intact due to the fact that additional fingerprint request is prompted when a device stolen with the both the SIM and PIN. Thus, HAMB offers a more resilient security for mobile banking. Even a more recent "tap and go" method otherwise known as PayAttitude is not immune from these vulnerabilities. PayAttitude is an extension of PAMB which offers more assurance of availability even in remote locations where there could be network challenges that may affect mobile banking transactions. PayAttitude makes use of a chip attached to the mobile device and used for transactions at a close range or from distant location. PayAttitude is also prone to SPOF which HAMB effectively prevents.

Chapter Four

Implementation and Discussion

4.0 Introduction

This chapter deals with the implementation of the new hybrid system, results obtained from the experiment and critically looking at the outcomes. These are examined under three subsections.

4.1 DEVELOPMENT ENVIROMENT

This section examined the basic environment in terms of the hardware and software configurations used in the new solution.

4.1.1 HARDWARE ENVIRONMENT

A laptop with an Intel (R) Core T6400 processor based system running at 2.00 GHz was used to connect to a virtual server through the internet.

Other required hardware/tools include:

1. Secugen Fingerprint Scanner
2. External Hard Drive for Backup of the source code and other data that might be needed in case of system re-installation.

4.1.2 SOFTWARE ENVIRONMENT

The new solution is web-based. The virtual server has the following configurations:

4.1.2.1 OPERATING SYSTEM

Windows Server 2008 R2 Standard Edition x64 was used as the Operating System (OS) because it is stable. It also has high capacity for networked-systems management and resource sharing.

4.1.2.1 Database Structure

In designing and developing the Microsoft SQL Server 2008 R2 Express Edition was used. This is it is readily compatible with the Windows Server 2008 used as the OS. It is most suitable for a Client-Server environment.

All biometric solutions basically follow the same process of secure capturing (also known as enrolment), storing the captured data with restricted access and matching of details. In this research, fingerprint is used to fortify the existing PIN security on mobile banking. A digital form is presented for the Admin user to capture a mobile banking user. For the purpose of this research, a mobile user is called a Normal user to distinguish between the individuals doing the enrolment and being enrolled. The created Admin user logs on to the new solution with his credentials and captures the Bio data of the normal user. The bio data details include: first name, middle name, last name, PIN and phone number. These are stored in a table called details_tab. Then, the fingerprints of the ten fingers are captured with the phone number are captured into the finger_tab. Worthy of note is the fact that the phone serves as the primary account number that uniquely identifies a customer in a bank. It is always traceable to the accounting records of the bank. Thus, the phone number becomes the link between the two tables – details_tab and finger_tab (Fig 4.1). However, all fields in the two tables are not mandatory to allow for cases whereby any of the fingers or middle name does not exist. The process of verification therefore, involves the extraction of the relevant template from the details_tab and comparing it with the fingerprint from the finger_tab.

Fig 4.1: Database Structure of the New System

4.2 Data Presentation

Table 4.1 shows the outcome of fifty attempts made by normal users of the new system.

Table 4.1: Analysis of result from the new system

"v" represents correct details

"×" represents wrong parameter

PIN	SIM	fingerprint	no of attempt	no of success	no of failure	success rate	failure rate
v	v	V	50	49	1	0.98	0.02
×	v	V	50	0	50	0	1
v	v	×	50	0	50	0	1
×	×	V	50	0	50	0	1
×	×	×	50	0	50	0	1
v	×	×	50	0	50	0	1
v	×	V	50	0	50	0	1
×	v	×	50	0	50	0	1

4.3 Analysis

Table 4.1 clearly shows that out of 50 trials made by legitimate users whose complete details had been taken stored in the database, only 1 of them could not be permitted to access the server. In a similar vein, none of the 50 trials made by persons who were never enrolled could be permitted to access the server. These are presented below:

1. False Rejection Rate, FRR (all parameters were correct, yet user denied access) =1/50 = 0.02 which is strictly less than 0.1
2. False Acceptance Rate, FAR (at least one of the parameters was wrong, yet user granted access) = 0/50 = 0 which is strictly less than 0.1

With the confidence level of 0.1% or confidence interval of 99% set for the test as seen from the result displayed in the Table 4.1, the probability of a false rejection after 50 trials was 0.02 which makes the probability of correctly verifying a registered customer 0.98

The probability of correctly accepting an unregistered customer stands at 0%. Based on this result we can deduce that a registered customer will be correctly verified and will be able to use the new system more than 49 times out of 50 while an unregistered user has no chance of being correctly verified.

Chapter Five

SUMMARY, CONCLUSION AND RECOMMENDATIONS

5.1 Introduction

This chapter presents the summary of the research, conclusion drawn from the simulation, recommendation based on the obtained results and other areas of focus for future researchers.

5.2 Summary

The research has been able to demonstrate the efficiency of hybrid PIN and fingerprint –based authentication in a real life environment. Over time biometric tokens have proved secure and reliable in various applications. This fact has also been corroborated by the result of this research. The unique characteristics of biometric features as designed by nature made them secure. Since no two human beings, no matter how identical they seem, would have exactly the same biometrics such as iris, face, palms and fingerprints. The main focus of this work was on using PIN and then providing additional authentication through biometric fingerprint. This has made it difficult for a third party, a fraudster, to match the distinctiveness offered by biometrics.

5.3 Conclusion

The implementation of hybrid PIN and fingerprint authentication solution for mobile banking leveraged on acceptability, stability, reliability and cost effectiveness of biometric fingerprint characteristics. While the existing PIN authentication has not been able to prevent identity theft, the hybrid solution offers a more secure platform for mobile banking.

The solution leveraged on the technological advancement in mobile devices that incorporates biometric input facilities. Thus, making the entire system safer, more reliable and also affords users the opportunity of easy usage without having to carry other extensions about. Biometrics has become an important of our security systems nowadays due to its varying usage. It is used in schools to effect necessary checks before exams to ward off impersonation. To a very great extent, this has been successful. It is also used in iphones to lock and unlock screen.

This research has been able to meet its set out objectives of designing an efficient model that combined both PIN and fingerprint. It also built an efficient database that enable and reduced the FAR and FRR. The efficiency of the solution was also tested by comparing it with an existing application. Major achievements of the work include:

1. Developing a hybrid model for mobile banking authentication using traditional PIN and biometrics. Fingerprint was used as a second factor authentication to further strengthen the prevailing PIN. In cases where a mobile device is stolen from the owner, even when the PIN is obtained or guessed, transactions can still not be consummated unless the fingerprint is added.
2. Building an efficient solution that reduced the error rates while authenticating users. Mobile banking application users have their details such as phone numbers, PIN and fingerprints stored in the bank's database. All these details are as well tied to the customers' account numbers stored in the bank's core application. Even though there are so many accounts, each customer is uniquely identified by an account number to which other mobile banking details are linked. The search is done by, first of all, accessing the mobile application stored on the mobile device. The application prompts for the first level authentication of supply of PIN. Since the SIM is already identified with the device, the system only needs to verify that it is the device owner – through the SIM – who is accessing the mobile application. Supplying the correct PIN completes the I & A stage and links it with the specific account number among the numerous accounts. Since the fingerprints are linked (with phone number and PIN), the search is limited to the particular account number. The system only awaits fingerprint input for matching to further authenticate that no one else but the account owner accesses the account once it (fingerprint) matches the one stored in the database for that account.
3. Implementation of a hybrid authentication system using both PIN and biometric fingerprint. This research was able to implement the PIN and fingerprint authentication to further strengthen the security of mobile banking transactions.

This solution has been able to achieve a broad objective of securing the mobile banking application user, the bank and extension, the economy by eliminating identity theft on mobile banking transactions which has been a major reason for customer's inability to embrace this

initiative. If this solution is deployed, the customer will be able to securely carry out their mobile banking transactions (such as: funds transfer, purchases, checking balances among others). The banks will also be free of financial loss and win their customers' loyalty. The economy as well will also reap the benefits of cashless economy.

While it can be out rightly used on its own in some applications, it may need to be supported by other compensating control mechanisms. Management of organizations may have to review their process vis-à-vis the cost implication to determine the suitable and profitable approach.

5.4 Recommendations

Having realized the importance of biometric tokens, it is also noteworthy to bear in mind that no security systems can be wholly full proof. Thus, a holistic approach is recommended to ensure that each stakeholder (individual users, banks and government) plays his role. We recommend the followings:

5.4.1 Individual users

1. Users should take proper custody of their mobile devices.
2. Cases of lost or theft should be promptly reported to the users' bank for a prompt action.
3. Users' details should not, for any reasons be divulged to a third party.

5.4.2 Banks

1. Enrolment should be carried out in secured environment. Enrolment is very key to the success of any biometric application. Hence, banks should ensure that access is restricted to the enrolment arena.
2. Access to the application should be on a least privilege basis. Users to the application have different roles to play such as creation, enrolment, approval, review among others. Therefore, they should be properly defined and access should not be granted beyond what a user requires.
3. Anti-fraud rules should be in place to safeguard users from being defrauded by setting necessary triggers to alerts designated officers to take appropriate steps. Necessary

apparatus should be in place to monitor pattern of customers' transactions and set appropriate rules based on predefined parameters such as location, time, value or volume beyond which further clarifications may be made.

4. Banks should also ensure that there mechanisms in place to determine possible intruders into the system. This may depend on the biometric token in use. Some applications are fortified with high level of intelligence to detect sharp practices. Sometimes, the heat of the fingerprint may be checked to ensure that it is not replay a stolen image. Some applications also checkmate the facial identity theft by ensuring that the user blinks or changes position of the head. Otherwise, it is taken a mere picture and thereby denied access.
5. The application should enforce auto-lock for screen after a certain period of idleness. Based on internal review, minutes of idleness should be set at the application level. Mostly, this is set to 3. There is no hard and fast rule to this.
6. Customers' education or awareness should be carried out from time to time to prevent social engineering.
7. Limit should be set on transaction per day beyond which necessary approval should be obtained.
8. Insurance policy should also be in place for risk to be transferred when the need arises.
9. Banks should ensure that end-to-end encryption is in place to protect data as it travels from users through the telecommunication infrastructures to the banks servers.

5.4.3 Government

1. The government through its agencies (such as CBN, Commissions or other regulatory bodies) should designed appropriate policies subject to timely review.
2. Should be involved in awareness campaign.

5.5 Contribution To Knowledge

This research has contributed to the existing security control on Mobile banking in the following ways:

1. Strengthening the authentication process through the introduction of fingerprint as an additional security method for mobile banking. Fingerprint is unique to each user and it is difficult to steal. Thus, making it difficult for an intruder or a fraudster to steal a mobile device and defraud an authentic user.
2. In comparison with the existing platform, the new solution offers better security by protecting user's mobile transaction even where the device is lost with the PIN thereby making it safer for user to protect his identity.
3. The solution is efficient since it has very low FAR and FRR thereby making it more friendly to users.

5.6 Suggestion for further Study

The role played by biometrics in security cannot be over-emphasized. Each of the biometric tokens can be very useful. This research employed fingerprint as a means of second level authentication. For shortness of time, the application was restricted to a single scanner as the only input device. It was not subjected to stress test. Future researchers may want to study how the application would respond with multiple scanning devices supplying input all at the same time. In real life scenario, different mobile devices would access same application and invariably, the database simultaneously. The result of this study can be used for electronic voting where an electorate will only be entitled to only one vote as well as boarder control. However, these may involve a secure and centralized database for storing captured data.

REFERENCES

Abdulah, S. (2012). Electronic payment systems. www.pbs.plymouth.ac.uk. Retrieved May 16, 2015

Adegbenle, A. (2013). Reducing Automated Teller Machine (ATM) fraud using fingerprint biometric technique. M.Sc. Project. Babcock University

Akash, K. (2015). Ten risks of mobile banking transactions. www.business-standard.com. Retrieved 3rd January, 2016.

Akram, H. and Hoffmann, M. (2008). Laws of Identity in Ambient Environments: The HYDRA Approach. IEEE Computer Society, 978-0-7695-3367-4/08

Al Shehri, W. (2013). Cloud database – database as a service. International Journal of Database Management Systems (IJDMS) Vol. 5, No. 2, April 2013.

Aloul, F., Zahidi, S. and El-Hajj, W. (2009). Multifactor authentication using mobile phone. International Journal of Mathematics and Computer Science 4(2009), no 2, 65-80.

Anyasi, F. and Otubu, P. (2009). Mobile Phone Technology in Banking System: Its Economic Effect. Research Journal of Information Technology 1(1): 1-5, 2009 ISSN: 2041-3114

Archana, S. and Vineet, K. (2012). Mobile banking as technology adoption and challenges: a case of m-banking in India. www.ijsrp.org Retrieved 15th May, 2015

Arnesen,S. (2013). Is a cloud ERP solution right for you? www.softresources.com. Retrieved June 9, 2013

AuWerter, S. (2012). Is mobile banking really safe? www.money.cnn.com. Retrieved January 20, 2014

Ayo, C. , Ukpere, W., Oni, A., Omote, U. and Akinsiku, D.(2012). A prototype mobile money implementation in Nigeria. ISSN 1993-8233 ©2012 Academic Journals. African Journal of Business Management Vol. 6(6), pp. 2195-2201, 15 February, 2012

Badger, L., Bohn, R., Chandramouli, R., Grance, T., Karygiannis, T., Patt-Comer, R. and Voas, J. (2010). Cloud computing use cases. www.nist.gov. Retrieved May 11, 2015

Bamoriya, P. S. and Singh, P. (2011). Issues & Challenges in Mobile Banking In India: A Customer's Perspective. Research Journal of Finance and Accounting. ISSN 2222-1697 Vol. 2 No 2, 2011

Bank of America (2006). Mobile banking security from Bank of America. www.bankofamerica.com. Retrieved January 20, 2014

Brooks, C. (2010). How to build an application for the cloud. www.searchcloudcomputing.techtarget,com Retrieved May 12, 2015

Central Bank of Nigeria (CBN). Regulatory framework for mobile payments services in Nigeria. www.cenbank.org. Retrieved May 3, 2013.

Chan, J. (2013). Internet banking – benefits and challenges in an emerging economy. International Journal of Research in Business Management. Vol 1, issue 1,June 2013 pp 19-26

Chen, W. and Gao, Y. (2007). A minutiae-based fingerprint matching algorithm using phase correlation.

Chikomo, K., Chong, M. K., Arnab, A. and Hutchison, A. (2006) Security of Mobile Banking. www.researchgate.net. Retrieved May 13, 2015

Chukwu, C. (2013). Introduction of the three-tiered know your customer (KYC) requirements. FPR/DIR/CIR/GEN/02/001

Cnet. Major UK banks Natwest and RBS to use iphone touch id fingerprint scanner. www.cnet.com Retrieved March 2, 2015

Dittrich, D., Bailey, M. and Dietrich, S. (2009). Towards community standards for ethical behavior in computer security research. Technical Report 2009-1, Stevens Institute of Technology

Donner, J. and Tellez, C. (2008). Mobile banking and economic development. www.libra.msra.cn. Retrieved May 13, 2015

Edsbacker, P. (2011). SIM Cards for Cellular Networks**:** An introduction to SIM Card Application Development. www.miun.diva-portal.org. Retrieved May 13, 2015

Gunjan, K., Sahoo, G. and Tiwari, R. (2012). Identity management in cloud computing –A Review. International Journal of Engineering Research & Technology (IJERT) Vol. 1 Issue 4, June – 2012.

Gopalakrishna, A. (2009). Cloud computing identity management. SETLabs Briefings Vol. 7 No 7, 2009

Higgins, K. (2013). Weak Security In Most Mobile Banking Apps. www.darkreading.com. Retrieved January 4, 2014

Hu, V., Ferraiolo, D., Kuhn, R., Schnitzer, A., Sandlin, K., Miller, R. and Scarfone, K. (2014). Guide to Attribute Based Access Control (ABAC) definition and considerations.

National Institute of Standards and Technology. www.nvipubs.nist.gov. Retrieved May 13, 2015

Huth, A., Orlando, M. and Pesante, L. (2012). Password security, protection and management. www.us-cert.gov. Retrieved July 12, 2015

Jacobs, B. and Poll, E. (2010). Biometrics and Smart Cards in Identity Management. www.cs.ru.nl. Retrieved May 13, 2015

John, S. (2013). Efficient bandwidth drives cloud computing. www.Nigeriacommunicationsweek.comng. Retrieved June 12, 2013.

Kenneally, E., Bailey, M. and Manghan, D. (2010). A framework for understanding and applying ethical principles in network and security research. FC.LNCS vol 6054, pp 240-243

Khan, M., Mahapatra, S. and Sreekrumah (2009). Service quality evaluation in internet banking: an empirical study in India, International Journal of India Culture and Business Management, 2(1), pp 30-46

Kim, J. and Hong, S. (2011). A method of risk assessment for multi-factor authentication. Journal of Information Processing Systems, vol. 7, No. 1.

King James Bible. Luke 12 verse 7. www.kingjamesbibleonline.com. Retrieved January 20, 2014

Klein, M and Mayer, C. (2011). Mobile banking and financial inclusion. www.openknowledge.worldbank.org. Retrieved May 13, 2015

Kossman, S. (2013). Ten dangers of mobile banking. www.money.usnews.com. Retrieved 8th January 2016

KPMG (2013). The cloud takes shape. www.kpmg.com. Retrieved June 9, 2013

Krinsbruner, E. (2013). Key Considerations for Testing Voice Recognition in Mobile Banking Applications. www.banktech.com. Retrieved May 13, 2015

Kypreos, E. (2014). Galaxy S5 Fingerprint Scanner vs iPhone 5S Touch ID. www.trustedreviews.com. Retrieved April 13, 2015

Lee, M. (2008). Factors influencing the adoption of internet banking: An integration of TAM and TPB with perceived risk and perceived benefits. www.csc-studentweb.lr.edu. Retrieved May 13, 2015

Mallat, N. (2007). Exploring consumer adoptionof mobile payments - qualitative study. Journal of Strategic Information Systems. Vol 16 pp 413-432

Masocha, R., Chiliya, N. and Zindriye, S. (2011). E-banking adoption by customers in rural miliues of South Africa: A case of Alice, Eastern Cape, South Africa. African Journal of Business Management, 5(5), pp 1857

Mell, P. and Grance, T. (2011). The NIST definition of cloud computing. www.nist.gov. Retrieved May 11, 2015

Meng, Y. (2013). Design of behavioral biometric based authentication with an adaptive mechanism on mobile phones. City University of Hong Kong. www.kaspersky.com, Retrieved May 13, 2015

Miguel, C., Luis, P. and Dominugo, M. (2007). Bimodal biometric person identification system under perturbations.Springer-Verlag Berlin Holdelberg pp 114 -127.

Mobile Marketing Association, MMA (2009). Mobile banking overview. www.mmaglobal.com Retrieved May 28, 2015

Nandakumar, K., Ross, A. and Jain, A. (2009). Biometric fusion: does modeling correlation really matter? IEEE 3rd Intl. Conf. on Biometrics: Theory, Applications and Systems (BTAS 09), Washington DC

Northcutt, S. (2014. Two factor authentication for online banking. www.sans.edu. Retrieved December 7, 2014

Odumeru, J. (2013). Going cashless: adoption of mobile banking in Nigeria. Arabian Journal of Business and Management Review (Nigerian Chapter) Vol. 1, No. 2, 2013

Onankunju, B. (2013). Access control in cloud computing. International Journal of Secience and Research Publications, Vol. 3, Issue 9.

Ondiege, P. (2010). Mobile banking in Africa: taking the bank to the people. www.afdb.org. Retrieved May 13, 2015

Paladi, N. (2012). Trusted computing and secure virtualization in cloud computing. Lulea University of Technology, dept. of computer science.

Papadopouli, M. (2009). Mobile identity management. www.eniza.europa.eu. Retrieved May 16, 2015

Pesante, L. (2008). Introduction to information security. www.us-cert.gov. Retrieved May 16, 2015

Rajendran, M. (2014). Security alert: Frauds can clone your SIM, use your credit card. www.hindustantimes.com. Retrieved November 28, 2015

Scarfone, K., Souppaya, M. and Hoffman, P. (2011). Guide to security for full virtualization technologies. National Institute of Standards and technology. Special publication 800-125

SANS Institute. Information security resources. www.sans.org. Retrieved May 14, 2015

SANS Institute. Computer security resources. www.sans.org. Retrieved May 14, 2015

Scotiabank (2010). Mobile Banking Security and Privacy. www.scotiabank.com. Retrieved January 20, 2014

Siciliano, R. (2013). What is Mobile Banking? Is it Safe? www.huffingtonpost.com. Retrieved January 20, 2014

Simotas, G., O'Driscoll, J. and Linder, J. (2011). Pursuing Mobile Banking Solutions. www.pdfvioh.org. Retrieved May 13, 2015

Singleton, T. (2007). What every IT auditors should know about information security. Information Systems Control Journal Vol. 2

USA Social Security Administration. Social Security.Identity theft and your social security number. www.socialsecurity.gov. Retrieved April 9, 2013

Shappiro, S.and Stockman, S. (2000). Image segmentation. IEEE conference on computer vision and pattern recognition 2000, 1-51

Shih, F. (2010). Image processing and pattern recognition, Institute of Electrical Electronics Engineers. John Wiley and Sons Inc, Hobokin, NewJersey.

Silver, B. (2000). An introduction to digital image processing cognex corporation, modular vision system division Natick

Swan, K., Kratcoski, A. and Hooft, M. (2007). Highly mobile devices, pedagogical possibilities and how teaching needs to be reconceptualized to realize them. www.researchgate.net. Retrieved May 13, 2015

Symantec (2011). Two-factor authentication: a TCO viewpoint. www.symantec.com Retrieved July 7, 2015

Thornton, G. (2012). Global projects in identity management and infrastructure security. www.isaca.org. Retrieved May 17, 2015

Topcoder (2015). Data science tutorials. www.topcoder.com. Retrieved April 9 2015.

Trewin, S., Swart, C., Koved, L., Martino, J., Singh, K. and Ben-David, S. (2012). Biometric Authentication on a Mobile Device: A Study of User Effort, Error and Task Disruption. www.researcher.ibm.com. Retrieved May 13, 2015

Ullrich, J. (2012). Security of Mobile Banking and Payments. www.sans.org. Retrieved May 13, 2015

US-CERT (2010). Cyber threats to mobile devices. Technical Information Paper – TIP-10-105-01. www.us-cert.gov. Retrieved May 16, 2015

White, E. (1892). Steps to Christ. PP85 -90. www.whiteestate.org Retrieved May 12, 2015

Woodill, G. (2012). Using mobile devices as research tools. www.floatlearning.com. Retrieved May 14, 2015

Wooley, P. (2011). Identifying cloud computing security risks. M.Sc. Project. University of Oregon.

Zhao, Y. and Kurnia, S. (2013). Exploring mobile payment option in China. www.pacis-net.org. Retrieved May 15, 2015.

Zimmermann, R. (2011). Towards cloud computing. www.future-internet.com. Retrieved May 12, 2015

Zuva, T., Esan, O. and Ngwira, S. (2014). Hybridization of Bimodal biometrics for access control authentication. International Journal of Future Computer and Communicator. Vol. 3 No. 6, 2014 444-451

Appendix

Server Code

```
/*

 * To change this license header, choose License Headers in Project Properties.

 * To change this template file, choose Tools | Templates

 * and open the template in the editor.

 */

package Hybrid;



import SecuGen.FDxSDKPro.jni.JSGFPLib;

import SecuGen.FDxSDKPro.jni.SGFDxErrorCode;

import SecuGen.FDxSDKPro.jni.SGFDxSecurityLevel;

import java.sql.ResultSet;

import javax.jws.WebService;

import javax.jws.WebMethod;

import javax.jws.WebParam;



/**

 *

 * @author Niyi

 */

@WebService(serviceName = "NewWebService")
```

```
public class NewWebService {


    DB_OPS ops_4_DB = new DB_OPS();

    JSGFPLib fplib = new JSGFPLib();

    String Errorstr;


    /**

     * This is a sample web service operation

     */

    public String hello(@WebParam(name = "name") String txt) {

        return "Hello " + txt + " !";

    }



    /**

     * Web service operation

     */

    @WebMethod(operationName = "verify")

    public String verify(@WebParam(name = "phonenum") String phonenum, @WebParam(name =
"PIN") String PIN, @WebParam(name = "finger") byte[] finger) {

        //TODO write your implementation code here:

        String ret_val ="-1";
```

```
//      ret_val = "3";

      // If wrong PIN, return

      try

      {

         ret_val ="-5";

         ops_4_DB.createConnection();

         int pin_result;

         pin_result = auth_PIN( PIN, phonenum ); //1 : pin found; 2 : wrong PIN; 0 : wrong phone number

          switch (pin_result)

         {

           case 0:   //Wrong Phone Number

           {

              ret_val = "0";

              break;

           }

           case 1:

             {

                int finger_result;

                finger_result = verify_finger(phonenum,finger);

                switch(finger_result)

                {

                   case 0 :   // Verification failed
```

```
                    {
//System.out.println("Verification Failed");
                       ret_val = "1";
                       break;
                    }
                    case 1 :  // Verify success
                    {
//System.out.println("Verification Success");
                       ret_val = "2";
                       break;
                    }
                    case 2 :  // SERVER ERROR
                    {
                       ret_val = "3";
                       break;
                    }
                    default:
                    {
                       ret_val = "-2";
                    }
                 }
                 break;
```

```
        }
      case 2:   //Wrong PIN
        {
          ret_val = "4";
          break;
        }
      case 4:   //ERROR
        {
          ret_val = "-6";
          break;
        }
      default:
      {
        ret_val = "-3";
        break;
      }
    }
return ret_val; //Server Error
}
catch(Exception ex)
{
   System.out.println("Verify : " + ex.getMessage());
```

```
        return "5"; //Server Error
    }
}

private int auth_PIN(String PIN, String phonenum)
{
    ResultSet results;
    results=null;

    String stmt;

    try
    {
        stmt = " select PIN from details_tab " +
            " where phone_num = '" + phonenum + "'";

        results = ops_4_DB.sel_data(stmt);

        if (results.next())
        {
            if (results.getObject(1).toString().equals(PIN))
```

```
            {
               return 1;
            }
            else
            {
               return 2;
            }
         }
         else
         {
            return 0;
         }
      }
      catch (Exception ex)
      {
         Errorstr = ex.getMessage();
         System.out.println( "AUTH PIN ERROR " + ex.getMessage());
         return 4;
      }
   }

   private int verify_finger(String phonenum, byte[] finger)
```

```
  {
      long iError;
      boolean[] matched = new boolean[1];
      matched[0] = false;
      byte[] regMin1 = new byte[400];
      byte[] thefinger = new byte[400];

            try
            {
               ResultSet results;
               String stmt;

               results = null;

//System.out.println("In Verify Finger : " + finger.length);
               stmt = "select finger_1, finger_2, finger_3, finger_4, "
                    + "finger_5, finger_6, finger_7, finger_8, finger_9, "
                    + "finger_10  from finger_tab "
                    + " where phone_num = '" + phonenum + "'";

               results =ops_4_DB.sel_data(stmt);
```

```
//                  res:
                  while (results.next() == true)
                  {
                    int i;
                     for ( i = 1;  i<=10; i++ )
                        {
                        regMin1 = results.getBytes(i);
                          if (regMin1 == null)
                            continue ;
//                              System.out.println("Here  finger :" + thefinger.length  + " DB finger : " +
regMin1.length);
//                          iError = fplib.MatchTemplate( regMin1, finger, 5, matched);
                          iError = fplib.MatchTemplate( regMin1, finger, SGFDxSecurityLevel.SL_NORMAL,
matched);
//System.out.println("In Verify Finger : IError " + iError + " Matched : " + matched);


                          if (iError == SGFDxErrorCode.SGFDX_ERROR_NONE)
                          {
//                              System.out.println("ERROR NONE");
                           if (matched[0])
                            {
//                              System.out.println("Matched");
```

```
//                                break res;

                                return 1;

                              }

                            }

                          }

                    }

                    return 0;

                }

                catch(Exception ex)

                {

//                    Errorstr = ex.getMessage();

                    System.out.println("Verify Finger : " + ex.getMessage());

                    return 2;

          //            this.jLabelStatus.setText("ERROR ! " + ex.getMessage() + "  " + regMin1 + "   " + vrfMin );

                }

    }

    @WebMethod(operationName = "hello")



    /**

     * Web service operation

     */

    @WebMethod(operationName = "error_msg")
```

```
    public String error_msg() {
        //TODO write your implementation code here:
        return Errorstr;
    }

}
```

Client Code

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package hybridclient;

/**
 *
 * @author OOO
 */
import SecuGen.FDxSDKPro.jni.*;
import java.awt.*;
import java.awt.image.*;
```

```
import java.sql.ResultSet;

import javax.swing.*;



/**

 *

 * @author  Administrator

 */

public class JSGD extends javax.swing.JDialog {



   //Private instance variables

   private long deviceName;

   private long devicePort;

   private JSGFPLib fplib = null;

   private long ret;

   private boolean bLEDOn;

   private byte[] regMin1 = new byte[400];

   private byte[] regMin2 = new byte[400];

   private byte[] vrfMin  = new byte[400];

   private byte[] tempmin = new byte[400];

   private SGDeviceInfoParam deviceInfo = new SGDeviceInfoParam();

   private BufferedImage imgRegistration1;

   private BufferedImage imgRegistration2;
```

```
    private BufferedImage imgVerification;
    private boolean r1Captured = false;
    private boolean r2Captured = false;
    private boolean v1Captured = false;
    private long registeroff;



    /** Creates new form JSGD */
    public JSGD(long offregister)
    {
//    public JSGD() {
        registeroff = offregister;
        bLEDOn = false;
        initComponents();
        disableControls();
/*
        if (registeroff == 1)
            this.jComboBoxVerifySecurityLevel.setSelectedIndex(4);
        else
            this.jComboBoxRegisterSecurityLevel.setSelectedIndex(4);
*/
        }
```

```
private void disableControls()
{
   this.jButtonToggleLED.setEnabled(false);
   this.jButtonCapture.setEnabled(false);
   this.jButtonCaptureR1.setEnabled(false);
   this.jButtonCaptureR2.setEnabled(false);
   this.jButtonCaptureV1.setEnabled(false);
   this.jButtonRegister.setEnabled(false);
   this.jButtonVerify.setEnabled(false);
   this.jButtonGetDeviceInfo.setEnabled(false);
   this.jButtonConfig.setEnabled(false);
}

private void enableControls()
{
   this.jButtonToggleLED.setEnabled(true);
   if (registeroff == 1)
      this.jButtonCaptureV1.setEnabled(true);
   else
   {
```

```
            this.jButtonCapture.setEnabled(true);
            this.jButtonCaptureR1.setEnabled(true);
            this.jButtonCaptureR2.setEnabled(true);
        }
        this.jButtonGetDeviceInfo.setEnabled(true);
        this.jButtonConfig.setEnabled(true);
    }
    private void enableRegisterAndVerifyControls()
    {
        if (r1Captured && r2Captured)
            this.jButtonRegister.setEnabled(true);
//      if (r1Captured && r2Captured && v1Captured)
        if (v1Captured)
            this.jButtonVerify.setEnabled(true);
    }

    /** This method is called from within the constructor to
     * initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is
     * always regenerated by the Form Editor.
     */
    // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents
```

```
private void initComponents() {

    jLabelStatus = new javax.swing.JLabel();
    jTabbedPane1 = new javax.swing.JTabbedPane();
    jPanelImage = new javax.swing.JPanel();
    jButtonInit = new javax.swing.JButton();
    jLabelImage = new javax.swing.JLabel();
    jComboBoxUSBPort = new javax.swing.JComboBox();
    jButtonToggleLED = new javax.swing.JButton();
    jButtonCapture = new javax.swing.JButton();
    jButtonConfig = new javax.swing.JButton();
    jLabel1 = new javax.swing.JLabel();
    jSliderQuality = new javax.swing.JSlider();
    jLabel2 = new javax.swing.JLabel();
    jLabel3 = new javax.swing.JLabel();
    jSliderSeconds = new javax.swing.JSlider();
    jPanelRegisterVerify = new javax.swing.JPanel();
    jLabelVerification = new javax.swing.JLabel();
    jLabelVerificationBox = new javax.swing.JLabel();
    jLabelVerifyImage = new javax.swing.JLabel();
    jButtonCaptureR1 = new javax.swing.JButton();
    jButtonCaptureV1 = new javax.swing.JButton();
```

```
jButtonRegister = new javax.swing.JButton();

jButtonVerify = new javax.swing.JButton();

jButtonCaptureR2 = new javax.swing.JButton();

jProgressBarV1 = new javax.swing.JProgressBar();

jTextFieldDeviceID = new javax.swing.JTextField();

jTextFieldFWVersion = new javax.swing.JTextField();

jTextFieldSerialNumber = new javax.swing.JTextField();

jTextFieldImageWidth = new javax.swing.JTextField();

jTextFieldImageHeight = new javax.swing.JTextField();

jTextFieldImageDPI = new javax.swing.JTextField();

jTextFieldBrightness = new javax.swing.JTextField();

jTextFieldContrast = new javax.swing.JTextField();

jTextFieldGain = new javax.swing.JTextField();

jButtonGetDeviceInfo = new javax.swing.JButton();

jComboBoxDeviceName = new javax.swing.JComboBox();

jLabelDeviceName = new javax.swing.JLabel();

jLabelSpacer1 = new javax.swing.JLabel();

jLabelSpacer2 = new javax.swing.JLabel();



jLabelphonenum = new javax.swing.JLabel();

jTextFieldPhonenum = new javax.swing.JTextField();

jLabelPIN = new javax.swing.JLabel();
```

```
        jPasswordFieldPIN = new javax.swing.JTextField();


        setTitle("Finger Print Verification");
        addWindowListener(new java.awt.event.WindowAdapter() {
            public void windowClosing(java.awt.event.WindowEvent evt) {
                exitForm(evt);
            }
        });
        getContentPane().setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());

/*      this.setMaximumSize(new java.awt.Dimension(400,400));
        this.setMinimumSize(new java.awt.Dimension(400,400));
        this.setPreferredSize(new java.awt.Dimension(400,400));
*/
        jLabelStatus.setText("Click Initialize Button ...");

jLabelStatus.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.L
OWERED));
        getContentPane().add(jLabelStatus, new org.netbeans.lib.awtextra.AbsoluteConstraints(10, 420,
490, 30));


        device_init_component();
```

```
        jTabbedPane1.addTab("Initialization", jPanelImage);


//      personal_info_components();
//      jTabbedPane1.addTab("Personal Info", jPanelPersonalDetails);


        jPanelRegisterVerify.setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());
//      if (registeroff == 1)
            verify_components();
//      else
//          reg_components();


//      jTabbedPane1.addTab("Register", jPanelRegisterVerify);


        jTabbedPane1.addTab("Finger Print", jPanelRegisterVerify);


//      device_info_components();
//      jTabbedPane1.addTab("Device Info", jPanelDeviceInfo);


        getContentPane().add(jTabbedPane1, new org.netbeans.lib.awtextra.AbsoluteConstraints(10, 35,
500, 420));
/*
```

```
        jComboBoxDeviceName.setModel(new javax.swing.DefaultComboBoxModel(new String[] { "AUTO",
"FDU05",  "FDU04", "FDU03", "FDU02" }));

        jComboBoxDeviceName.setMinimumSize(new java.awt.Dimension(350, 10));

        jComboBoxDeviceName.setVerifyInputWhenFocusTarget(false);

        */

//      getContentPane().add(jComboBoxDeviceName, new
org.netbeans.lib.awtextra.AbsoluteConstraints(130, 10, 350, -1));



        jLabelDeviceName.setText("Device Name");

//      getContentPane().add(jLabelDeviceName, new org.netbeans.lib.awtextra.AbsoluteConstraints(10,
11, 110, -1));



        jLabelSpacer1.setText(" ");

//      getContentPane().add(jLabelSpacer1, new org.netbeans.lib.awtextra.AbsoluteConstraints(510,
490, 10, -1));



        jLabelSpacer2.setText(" ");

//      getContentPane().add(jLabelSpacer2, new org.netbeans.lib.awtextra.AbsoluteConstraints(510, 10,
10, -1));



        pack();

    }// </editor-fold>//GEN-END:initComponents
```

```
    private void device_init_component()
    {
        jPanelImage.setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());

        jButtonInit.setText("Initialize");
        jButtonInit.setMaximumSize(new java.awt.Dimension(100, 30));
        jButtonInit.setMinimumSize(new java.awt.Dimension(100, 30));
        jButtonInit.setName("jButtonInit"); // NOI18N
        jButtonInit.setPreferredSize(new java.awt.Dimension(100, 30));
        jButtonInit.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButtonInitActionPerformed(evt);
            }
        });
        jPanelImage.add(jButtonInit, new org.netbeans.lib.awtextra.AbsoluteConstraints(10, 10, 100, 30));

jLabelImage.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBorder.LOWERED));
        jLabelImage.setMinimumSize(new java.awt.Dimension(200, 250));
        jLabelImage.setPreferredSize(new java.awt.Dimension(200, 250));
```

```
//      jPanelImage.add(jLabelImage, new org.netbeans.lib.awtextra.AbsoluteConstraints(10, 60, -1, -1));

/*

     jComboBoxUSBPort.setModel(new javax.swing.DefaultComboBoxModel(new String[] {
"AUTO_DETECT", "0", "1", "2", "3", "4", "5", "6", "7", "8", "9" }));

     jComboBoxUSBPort.setMaximumSize(new java.awt.Dimension(170, 27));

     jComboBoxUSBPort.setMinimumSize(new java.awt.Dimension(170, 27));

     jComboBoxUSBPort.setPreferredSize(new java.awt.Dimension(170, 27));

*/

     //      jPanelImage.add(jComboBoxUSBPort, new
org.netbeans.lib.awtextra.AbsoluteConstraints(280, 90, 170, 27));


     jButtonToggleLED.setText("Toggle LED");

     jButtonToggleLED.setMaximumSize(new java.awt.Dimension(100, 30));

     jButtonToggleLED.setMinimumSize(new java.awt.Dimension(100, 30));

     jButtonToggleLED.setPreferredSize(new java.awt.Dimension(100, 30));

/*      jButtonToggleLED.addActionListener(new java.awt.event.ActionListener() {

         public void actionPerformed(java.awt.event.ActionEvent evt) {

            jButtonToggleLEDActionPerformed(evt);

         }

     });  */

//      jPanelImage.add(jButtonToggleLED, new org.netbeans.lib.awtextra.AbsoluteConstraints(120, 10,
100, 30));
```

```
        jButtonCapture.setText("Capture");
        jButtonCapture.setMaximumSize(new java.awt.Dimension(100, 30));
        jButtonCapture.setMinimumSize(new java.awt.Dimension(100, 30));
        jButtonCapture.setPreferredSize(new java.awt.Dimension(100, 30));
/*      jButtonCapture.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButtonCaptureActionPerformed(evt);
            }
        });  */
//      jPanelImage.add(jButtonCapture, new org.netbeans.lib.awtextra.AbsoluteConstraints(230, 10,
100, 30));
        jButtonConfig.setText("Config");
        jButtonConfig.setMaximumSize(new java.awt.Dimension(100, 30));
        jButtonConfig.setMinimumSize(new java.awt.Dimension(100, 30));
        jButtonConfig.setPreferredSize(new java.awt.Dimension(100, 30));
 /*     jButtonConfig.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButtonConfigActionPerformed(evt);
            }
        });  */
```

```
//      jPanelImage.add(jButtonConfig, new org.netbeans.lib.awtextra.AbsoluteConstraints(340, 10, -1,
30));


        jLabel1.setText("USB Device");
//      jPanelImage.add(jLabel1, new org.netbeans.lib.awtextra.AbsoluteConstraints(280, 70, -1, -1));


        jSliderQuality.setMajorTickSpacing(10);
        jSliderQuality.setMinorTickSpacing(5);
        jSliderQuality.setPaintLabels(true);
        jSliderQuality.setPaintTicks(true);
        jSliderQuality.setName(""); // NOI18N
        jSliderQuality.setOpaque(false);
//      jPanelImage.add(jSliderQuality, new org.netbeans.lib.awtextra.AbsoluteConstraints(270, 170,
220, -1));


        jLabel2.setText("Image Quality");
//      jPanelImage.add(jLabel2, new org.netbeans.lib.awtextra.AbsoluteConstraints(280, 150, -1, -1));


        jLabel3.setText("Timeout (seconds)");
//      jPanelImage.add(jLabel3, new org.netbeans.lib.awtextra.AbsoluteConstraints(290, 230, -1, -1));


        jSliderSeconds.setMajorTickSpacing(1);
```

```
        jSliderSeconds.setMaximum(10);
        jSliderSeconds.setMinimum(1);
        jSliderSeconds.setPaintLabels(true);
        jSliderSeconds.setPaintTicks(true);
        jSliderSeconds.setValue(5);
//      jPanelImage.add(jSliderSeconds, new org.netbeans.lib.awtextra.AbsoluteConstraints(270, 250,
220, -1));

    }

    private void verify_components()
    {

        jLabelphonenum.setText("Phone Number");
        jPanelRegisterVerify.add(jLabelphonenum, new
org.netbeans.lib.awtextra.AbsoluteConstraints(15, 13, -1, -1));

        jPanelRegisterVerify.add(jTextFieldPhonenum, new
org.netbeans.lib.awtextra.AbsoluteConstraints(110, 10, 120, -1));

        jLabelPIN.setText("PIN ");
        jPanelRegisterVerify.add(jLabelPIN, new org.netbeans.lib.awtextra.AbsoluteConstraints(15, 43, -
1, -1));
```

```
        jPanelRegisterVerify.add(jPasswordFieldPIN, new
org.netbeans.lib.awtextra.AbsoluteConstraints(110, 40, 120, -1));


        jLabelVerification.setText("Verification");

//        jPanelRegisterVerify.add(jLabelVerification, new
org.netbeans.lib.awtextra.AbsoluteConstraints(230, 13, -1, -1));


//        jComboBoxVerifySecurityLevel.setModel(new javax.swing.DefaultComboBoxModel(new String[]
{ "LOWEST", "LOWER", "LOW", "BELOW_NORMAL", "NORMAL", "ABOVE_NORMAL", "HIGH", "HIGHER",
"HIGHEST" }));

//        jPanelRegisterVerify.add(jComboBoxVerifySecurityLevel, new
org.netbeans.lib.awtextra.AbsoluteConstraints(300, 10, 130, -1));

        jLabelVerificationBox.setBorder(javax.swing.BorderFactory.createTitledBorder("Verification"));

        jPanelRegisterVerify.add(jLabelVerificationBox, new
org.netbeans.lib.awtextra.AbsoluteConstraints(170, 70, 150, 240));


jLabelVerifyImage.setBorder(javax.swing.BorderFactory.createBevelBorder(javax.swing.border.BevelBor
der.LOWERED));

        jLabelVerifyImage.setMinimumSize(new java.awt.Dimension(130, 150));

        jLabelVerifyImage.setPreferredSize(new java.awt.Dimension(130, 150));

        jPanelRegisterVerify.add(jLabelVerifyImage, new
org.netbeans.lib.awtextra.AbsoluteConstraints(180, 90, -1, -1));
```

```
        jButtonCaptureV1.setText("Capture V1");
        jButtonCaptureV1.setActionCommand("jButton1");
        jButtonCaptureV1.setMaximumSize(new java.awt.Dimension(130, 30));
        jButtonCaptureV1.setMinimumSize(new java.awt.Dimension(130, 30));
        jButtonCaptureV1.setPreferredSize(new java.awt.Dimension(130, 30));
        jButtonCaptureV1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButtonCaptureV1ActionPerformed(evt);
            }
        });
        jPanelRegisterVerify.add(jButtonCaptureV1, new
org.netbeans.lib.awtextra.AbsoluteConstraints(180, 260, 130, 30));

        jButtonVerify.setText("Verify");
        jButtonVerify.setActionCommand("jButton1");
        jButtonVerify.setMaximumSize(new java.awt.Dimension(130, 30));
        jButtonVerify.setMinimumSize(new java.awt.Dimension(130, 30));
        jButtonVerify.setPreferredSize(new java.awt.Dimension(130, 30));
        jButtonVerify.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButtonVerifyActionPerformed(evt);
```

```
            }

        });

        jPanelRegisterVerify.add(jButtonVerify, new org.netbeans.lib.awtextra.AbsoluteConstraints(180,
305, 130, 30));



        jProgressBarV1.setForeground(new java.awt.Color(0, 51, 153));

        jPanelRegisterVerify.add(jProgressBarV1, new
org.netbeans.lib.awtextra.AbsoluteConstraints(180, 230, 130, -1));



    }

    private void jButtonVerifyActionPerformed(java.awt.event.ActionEvent evt) {//GEN-
FIRST:event_jButtonVerifyActionPerformed
//        long iError;
//        long secuLevel = (long) (this.jComboBoxVerifySecurityLevel.getSelectedIndex() + 1);
//        boolean[] matched = new boolean[1];
//        matched[0] = false;
       this.jLabelStatus.setText("");
        try
        {
          if (jTextFieldPhonenum.getText().length() == 0)
             this.jLabelStatus.setText("Please Enter Phone Number");
          else
```

```
        {
          if (jPasswordFieldPIN.getText().length() == 0)
              jLabelStatus.setText("Please Enter The PIN");
           else
          {

//              this.jLabelStatus.setText(hello("Oluwatosin"));
//                hello("Oluwatosin");

              String Verify_result;
              Verify_result = verify(jTextFieldPhonenum.getText(), jPasswordFieldPIN.getText(), vrfMin);
              int result_in_int;
              result_in_int=Integer.parseInt(Verify_result);
              switch (result_in_int)
              {
                 case 0 :
                 {
                    this.jLabelStatus.setText("Wrong Phone Number Entered " + Verify_result);
                    break;
                 }
                 case 1 :
                 {
```

```
    this.jLabelStatus.setText("Finger Not Matched " + Verify_result);
    break;
}
case 2 :
{
    this.jLabelStatus.setText("Finger Verification Success " + Verify_result);
    break;
}
case 3 :
{
    this.jLabelStatus.setText("SERVER ERROR, Please Try Again " + errorMsg());
    break;
}
case 4 :
{
    this.jLabelStatus.setText("Wrong PIN Entered " + Verify_result);
    break;
}
case 5 :
{
    this.jLabelStatus.setText("SERVER ERROR, Please Try Again" + errorMsg());
    break;
```

```
                }
                default:
                {
                   this.jLabelStatus.setText("ERROR " + errorMsg());
                   break;
                }

             }
          }
        }
      }
      catch(Exception ex)
      {
         this.jLabelStatus.setText("VERIFY ERROR : " + ex.getMessage());
      }

   }//GEN-LAST:event_jButtonVerifyActionPerformed

   private void jButtonCaptureV1ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-
FIRST:event_jButtonCaptureV1ActionPerformed
      int[] quality = new int[1];
```

```
byte[] imageBuffer1 = ((java.awt.image.DataBufferByte)
imgVerification.getRaster().getDataBuffer()).getData();

long iError = SGFDxErrorCode.SGFDX_ERROR_NONE;


iError = fplib.GetImageEx(imageBuffer1,jSliderSeconds.getValue() * 1000, 0,
jSliderQuality.getValue());

fplib.GetImageQuality(deviceInfo.imageWidth, deviceInfo.imageHeight, imageBuffer1, quality);

this.jProgressBarV1.setValue(quality[0]);

SGFingerInfo fingerInfo = new SGFingerInfo();

fingerInfo.FingerNumber = SGFingerPosition.SG_FINGPOS_LI;

fingerInfo.ImageQuality = quality[0];

fingerInfo.ImpressionType = SGImpressionType.SG_IMPTYPE_LP;

fingerInfo.ViewNumber = 1;


if (iError == SGFDxErrorCode.SGFDX_ERROR_NONE)

{

   this.jLabelVerifyImage.setIcon(new
ImageIcon(imgVerification.getScaledInstance(130,150,Image.SCALE_DEFAULT)));

   if (quality[0] == 0)

      this.jLabelStatus.setText("GetImageEx() Success [" + ret + "] but image quality is [" + quality[0] +
"]. Please try again");

   else

   {
```

```
            this.jLabelStatus.setText("GetImageEx() Success [" + ret + "]");


            iError = fplib.CreateTemplate(fingerInfo, imageBuffer1, vrfMin);
            if (iError == SGFDxErrorCode.SGFDX_ERROR_NONE)
            {
              this.jLabelStatus.setText("Verification image was captured");
              v1Captured = true;
              this.enableRegisterAndVerifyControls();
            }
            else
              this.jLabelStatus.setText("CreateTemplate() Error : " + iError);
          }
        }
        else
          this.jLabelStatus.setText("GetImageEx() Error : " + iError);


    }//GEN-LAST:event_jButtonCaptureV1ActionPerformed


    private void jButtonInitActionPerformed(java.awt.event.ActionEvent evt) {//GEN-
FIRST:event_jButtonInitActionPerformed


        try
```

```
{
int selectedDevice = jComboBoxDeviceName.getSelectedIndex();
switch(selectedDevice)
{
   case 0: //USB
   default:
      this.deviceName = SGFDxDeviceName.SG_DEV_AUTO;
      break;
   case 1: //FDU05
      this.deviceName = SGFDxDeviceName.SG_DEV_FDU05;
      break;
   case 2: //FDU04
      this.deviceName = SGFDxDeviceName.SG_DEV_FDU04;
      break;
   case 3: //CN_FDU03
      this.deviceName = SGFDxDeviceName.SG_DEV_FDU03;
      break;
   case 4: //CN_FDU02
      this.deviceName = SGFDxDeviceName.SG_DEV_FDU02;
      break;
}
fplib = new JSGFPLib();
```

```
if (fplib != null)
   {
      ret = fplib.Init(this.deviceName);
   }
if ((fplib != null) && (ret  == SGFDxErrorCode.SGFDX_ERROR_NONE))
{
   this.jLabelStatus.setText("JSGFPLib Initialization Success");
   this.devicePort = SGPPPortAddr.AUTO_DETECT;
   switch (this.jComboBoxUSBPort.getSelectedIndex())
   {
      case 1:
      case 2:
      case 3:
      case 4:
      case 5:
      case 6:
      case 7:
      case 8:
      case 9:
      case 10:
         this.devicePort = this.jComboBoxUSBPort.getSelectedIndex() - 1;
         break;
```

```
        }
        ret = fplib.OpenDevice(this.devicePort);
        if (ret == SGFDxErrorCode.SGFDX_ERROR_NONE)
        {
           this.jLabelStatus.setText("OpenDevice() Success [" + ret + "]");
           ret = fplib.GetDeviceInfo(deviceInfo);
           if (ret == SGFDxErrorCode.SGFDX_ERROR_NONE)
           {
              this.jTextFieldSerialNumber.setText(new String(deviceInfo.deviceSN()));
              this.jTextFieldBrightness.setText(new String(Integer.toString(deviceInfo.brightness)));
              this.jTextFieldContrast.setText(new String(Integer.toString((int)deviceInfo.contrast)));
              this.jTextFieldDeviceID.setText(new String(Integer.toString(deviceInfo.deviceID)));
              this.jTextFieldFWVersion.setText(new String(Integer.toHexString(deviceInfo.FWVersion)));
              this.jTextFieldGain.setText(new String(Integer.toString(deviceInfo.gain)));
              this.jTextFieldImageDPI.setText(new String(Integer.toString(deviceInfo.imageDPI)));
              this.jTextFieldImageHeight.setText(new String(Integer.toString(deviceInfo.imageHeight)));
              this.jTextFieldImageWidth.setText(new String(Integer.toString(deviceInfo.imageWidth)));
              imgRegistration1 = new BufferedImage(deviceInfo.imageWidth, deviceInfo.imageHeight,
BufferedImage.TYPE_BYTE_GRAY);
              imgRegistration2 = new BufferedImage(deviceInfo.imageWidth, deviceInfo.imageHeight,
BufferedImage.TYPE_BYTE_GRAY);
              imgVerification = new BufferedImage(deviceInfo.imageWidth, deviceInfo.imageHeight,
BufferedImage.TYPE_BYTE_GRAY);
```

```
            this.enableControls();
          }
          else
            this.jLabelStatus.setText("GetDeviceInfo() Error [" + ret + "]");
        }
        else
          this.jLabelStatus.setText("OpenDevice() Error [" + ret + "]");
      }
      else
        this.jLabelStatus.setText("Device Initialization Failed");
      }
      catch(Exception ex)
      {
        this.jLabelStatus.setText("Device Initialization Failed " + ex.getMessage());
      }



}//GEN-LAST:event_jButtonInitActionPerformed


/** Exit the Application */
private void exitForm(java.awt.event.WindowEvent evt) {//GEN-FIRST:event_exitForm
```

```
        System.exit(0);
    }//GEN-LAST:event_exitForm

    /**
     * @param args the command line arguments
     */
    public static void main(String args[]) {
//        new JSGD().setVisible(true);
    }

    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton jButtonCapture;
    private javax.swing.JButton jButtonCaptureR1;
    private javax.swing.JButton jButtonCaptureR2;
    private javax.swing.JButton jButtonCaptureV1;
    private javax.swing.JButton jButtonConfig;
    private javax.swing.JButton jButtonGetDeviceInfo;
    private javax.swing.JButton jButtonInit;
    private javax.swing.JButton jButtonRegister;
    private javax.swing.JButton jButtonToggleLED;
    private javax.swing.JButton jButtonVerify;
```

```
private javax.swing.JComboBox jComboBoxDeviceName;

private javax.swing.JComboBox jComboBoxUSBPort;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JLabel jLabelDeviceName;

private javax.swing.JLabel jLabelImage;

private javax.swing.JLabel jLabelSpacer1;

private javax.swing.JLabel jLabelSpacer2;

public javax.swing.JLabel jLabelStatus;

private javax.swing.JLabel jLabelVerification;

private javax.swing.JLabel jLabelVerificationBox;

private javax.swing.JLabel jLabelVerifyImage;

private javax.swing.JPanel jPanelImage;

private javax.swing.JPanel jPanelRegisterVerify;

private javax.swing.JProgressBar jProgressBarV1;

private javax.swing.JSlider jSliderQuality;

private javax.swing.JSlider jSliderSeconds;

private javax.swing.JTabbedPane jTabbedPane1;

private javax.swing.JTextField jTextFieldBrightness;

private javax.swing.JTextField jTextFieldContrast;

private javax.swing.JTextField jTextFieldDeviceID;
```

```
    private javax.swing.JTextField jTextFieldFWVersion;
    private javax.swing.JTextField jTextFieldGain;
    private javax.swing.JTextField jTextFieldImageDPI;
    private javax.swing.JTextField jTextFieldImageHeight;
    private javax.swing.JTextField jTextFieldImageWidth;
    private javax.swing.JTextField jTextFieldSerialNumber;

    private javax.swing.JLabel jLabelphonenum;
    private javax.swing.JTextField jTextFieldPhonenum;
    private javax.swing.JLabel jLabelPIN;
    private javax.swing.JTextField jPasswordFieldPIN;

    // End of variables declaration//GEN-END:variables

    private static String verify(java.lang.String phonenum, java.lang.String pin, byte[] finger) {
        hybrid.NewWebService_Service service = new hybrid.NewWebService_Service();
        hybrid.NewWebService port = service.getNewWebServicePort();
        return port.verify(phonenum, pin, finger);
    }

    private static String errorMsg() {
```

```
        hybrid.NewWebService_Service service = new hybrid.NewWebService_Service();
        hybrid.NewWebService port = service.getNewWebServicePort();
        return port.errorMsg();
    }




}
```

Database Codes

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package Hybrid;


import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.Date;
```

```
/**
 *
 * @author Niyi
 */
public class DB_OPS {

//      private static String dbURL = "jdbc:derby://localhost:1527/emission_db";
//      private static String dbURL = "jdbc:derby:C:\\Users\\OOO\\.netbeans-derby\\emission_db";
//      public  static String dbURL = "jdbc:derby:C:\\Finger PRNT\\DB\\two_way_DB";
        public  static String dbURL =
"jdbc:sqlserver://localhost\\SQLEXPRESS:1433;databaseName=fingerdb;User=twoway;Password=myPas
s#1";
//      public static String tableName = "category_tab";
// jdbc Connection
        private  Connection conn = null;
        private static Statement stmt = null;
        private static Date thedate;

        public String error_msg;
        private  ResultSet Results;
```

```
   public void DB_OPS()
   {}



      public  void createConnection()
      {
         try
         {
    //       Class.forName("org.apache.derby.jdbc.ClientDriver"); //.newInstance();
//           Class.forName("org.apache.derby.jdbc.EmbeddedDriver");
          Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
           //Get a connection
            conn = DriverManager.getConnection(dbURL);
            if (conn == null){
        //     for_error.setText(conn.toString());
               System.out.println("NULL");
        //     System.out.println(" NOT NULL");
               error_msg = "CONNECTION FAILURE";
            }
            else
            {
```

```
            error_msg = "CONNECTION SUCCESS";
          }
          }
        catch (Exception except)
        {
         error_msg = " CREATE CONNECTION " + except.toString();
         System.out.println(except.toString());
         except.printStackTrace();
        }
    }

public  void in_data(String tableName, String values)
{
    try
    {
     //       for_error.setText("Yea Here b4 create stmt");

        stmt = conn.createStatement();
        stmt.execute("insert into " + tableName + " values(" +
              values + ")");

        stmt.close();
```

```
      }
      catch (SQLException sqlExcept)
      {
         //sqlExcept.printStackTrace();
         error_msg = "IN DATA " + sqlExcept.toString();
         System.out.println("ERROR : " + sqlExcept.toString());
      }
   }

   public ResultSet sel_data (String stmt_str)
   {
       ResultSet results;
       results=null;
        try
      {
       //      for_error.setText("Yea Here b4 create stmt");
         stmt = conn.createStatement();
         results = stmt.executeQuery(stmt_str);
//                + " where cat_code = " + upd_combo.getSelectedItem());

         return results;
```

```
//          stmt.close();
      }
      catch (SQLException sqlExcept)
      {
         error_msg = "SEL CATE " + sqlExcept.toString();
         sqlExcept.printStackTrace();
      }

        return results;
  }

  public void upd_data(String stmt_str)
  {
        try
     {
       //       for_error.setText("Yea Here b4 create stmt");
         stmt = conn.createStatement();
         stmt.execute(stmt_str);

        stmt.close();
     }
     catch (SQLException sqlExcept)
```

```
      {
//          JSGD for_error = new JSGD(0);

          error_msg = sqlExcept.getMessage();
//          for_error.jLabelStatus.setText(error_msg);
//          sqlExcept.printStackTrace();
      }
   }

   public void del_data(String table, String where)
   {
      try
      {
       //       for_error.setText("Yea Here b4 create stmt");

          stmt = conn.createStatement();
          stmt.execute("delete  from "  + table
                + where);

          stmt.close();

      }
```

```
        catch (SQLException sqlExcept)
        {
           //sqlExcept.printStackTrace();
        }
    }

}
```