University of Nairobi

School of computing and Informatics

MSC information systems

Two Factor Authentication and transaction validation using a mobile phone

By

Caroline W. Maina

Reg. No P56/p/7586 /06

Supervisor

Prof. Wagacha Peter W.

February 2013

Submitted in Partial fulfilment of the requirements of Master of Science Information Systems

DECLARATION

This work is my original work and it has never been presented to any other institution for the award of any certificate whatsoever.

Signature-- Date ---

Caroline W. Maina

Reg.No. P56/P/7586/06

This project has been presented for examination with my approval as the student supervisor.

Signed --- Date -------------------------------

Prof. Wagacha W. Peter

Acknowledgement

I would like to thank Prof. Peter Wagacha for his invaluable guidance while undertaking this program and the research project. To the teaching fraternity school of computing and informatics and especially those in projects for your time and constructive criticism that made me think outside the box and want to go the extra mile. Thank you for making UoN the University of Choice.

I would also like to thank the following whose support in one way or the other has helped me so much: -

To the almighty God, for the gift of life and the opportunity to come this far.

To all my classmates who were there supporting me throughout the course.

My workmates who made provisions for me to be able to attend sessions and encouraged me.

To my husband Paul waiting for me every day for me to finish classes and supporting me in all ways to finish my course. To my children allowing me to be able to go through the program.

ABSTRACT

Online Banking provides speed, flexibility, and efficiency, the Internet has become the means for conducting growing numbers of transactions between suppliers and large international corporations. In this way, the Internet has opened new markets to the world and has accelerated the diffusion of knowledge. Internet markets or online business are widely used in these days (Hamdan et al., 2010).

Currently there are there are 43 licensed commercial banks and 1 mortgage finance company in Kenya. A number of banks offer internet banking. In the same breadth there are companies that have e- commerce sites in Kenya selling goods and services (Kenya Bankers Association (KBA), 2011).

In a study carried out by Phone Factor, they found out that real-time attacks from online banking Trojans (ZeuS, Clampi, etc.), also referred to as *Man-In-The-Middle attacks*, are seen as the greatest threat to online banking today for more than half (51%) of survey respondents.

Insecurity is also in the case of the personal data that may be stolen and also man in the middle attacks. Some of the attacks happen at the time of logging into the website or in the process of doing the transactions. This is usually done during authentication of the user of the website. (P.T.Joseph, 2005) identifies the risks as Data Protection, Data reliability and Taxation.

A Prototype application was designed that used a mobile phone to provide second factor authentication. To do a transaction a user entered their name and password into the website, once the details were authenticated they got a code on their phone that they used to do transaction. Once the transaction is complete a second code was sent to the mobile phone of the registered user or second account holder to log into the system and validate a transaction that has been done. The significance of the study will be to help institutions of different sizes to be able to secure their clients data as well as reduce the impact of Man in the middle attacks. The study Was able to demonstrate that a solution can easily be obtained at a cost that is not prohibitive without the reduction in service.

Table of Contents

List of Figures

Definition of terms

1. **Application Denial of Service** - Numerous types of attacks make use of the possibility of entering rogue information in input fields.
2. **Cross-Site Scripting** – A script is injected to one web site or web log, but it is operated at a different web site.
3. **Cookie tampering** – Information in the cookie is changed to allow an attack.
4. **Deception**

   This is the acceptance of false information. This could be in the form of snooping that is passive but listens to data that is being moved from point A to Point B. Deception could also include the alteration of the information that has been sent.

5. **Disclosures**

   This is an unauthorized access to information

6. **Disruption**

   This is interruption or prevention of correct operation.

7. **Dynamic Passwords**

8. Protection dynamic passwords this are passwords that expire through use of time. The Passwords can also expire based on the certificates dates and/or the number of users.**Key Logging** – Software implanted in the customer's computer that records all the keystrokes of the customer, providing a complete record of user IDs, passwords, pin codes, account numbers and transactions. Sometimes this is integrated with additional rogue software, and usually it sends the information it has collected to the hacker.
9. **Form Tampering (read-only and hidden fields)** – Changes are made in hidden or read-only fields in the HTML form.
10. **eplay attacks**

    When an attacker eavesdrops and records the authentication as it is communicated between a client and the financial Institution systems recording and establishing a connection later.

11. **Man-in-the-browser** (man in the middle) – A "Trojan horse" changes the contents of the form that the customer submits to the bank website. The change is not noticeable in the form itself. It takes place only

    In computer memory. It takes place before SSL encoding.

12. **Man in the Middle** - Rogue software is put in place at some point between the customer computer and the bank web sites and intercepts all the information transmitted between the customer and the bank.
13. **Man-in-the-browser** – A "Trojan horse" changes the contents of the form that the customer submits to the bank website. The change is not noticeable in the form itself. It takes place only in computer memory. It takes place before SSL encoding.
14. **Man in the Middle** - Rogue software is put in place at some point between the customer computer and the bank web sites and intercepts all the information transmitted between the customer and the bank.
15. **Session Hijacking** – The session is hijacked by unauthorized use of the cookies deposited by the banking site.
16. **Pharming** – Pharming is diversion of traffic from a legitimate site to a rogue web site.
17. **Phishing** – Customer identity details are stolen. Typically, this is carried out in a place and context removed from the bank web site, such as a fraudulent e-mail asking for information.
18. **Site Cloaking** – Cloaking fools search engines by disguising one web site as another.
19. **OS command injection** – Injection of operating system commands to be carried out at the web site.
20. **SQL Injection** – Injection of SQL queries to be executed at the web site.
21. **Outbound Data Theft** – Data sent from the web site are intercepted for use in attacks. For example, that may include data about the software installed at the site, version number etc.

List of Abbreviations used

ATM	**Automated Teller Machine**
CDI	**Uncontrolled Data Items**
HTML	**Hypertext Markup Language**
ITU	**International Communication union**
SQL	
PIN	**Personal Identification Number**
TP	**Transformation procedures**
UDI -	**Uncontrolled data items**

1.INTRODCTION

Chapter 1 – Overview of Chapter

This chapter looks at the introduction the background of the problem, and the justification of doing this research. It also looks and objectives and a statement of the problem.

1.1. Background

Many institutions are embracing ecommerce as well as electronic banking. These institutions would wish to have the knowledge that there systems will be safe when transacting online. The Government of Kenya is currently trying to sensitize organizations to move into the use of internet for commerce as well as the provision of services to the people of Kenya both in the country as well outside the country. In to enable many Kenyans get good connections as well as take services closer to the people.

Many Business owners with online websites, ecommerce applications or other services that require secure login for their customers, understand the importance of protecting customer information. These include information such as credit card numbers, personal addresses, and other "personally identifiable information" (PII). This kind of information can be used for wrong purposes if it went to people who are not trusted.

When evaluating Computer security the three core goals are confidentiality, integrity and availability says (Venkatramanayya, 2007) .these goals face two major risks namely identity theft and man-in the-middle- attack.

Identity theft leads to crooks opening accounts under the names they have obtained, using utilities, counterfeit checks making ATM withdrawals, taking out loans or even obtaining government benefits...

Man-in-the-middle attack or bucket-brigade attack (often abbreviated MITM), sometimes Janus attack, is a form of active eavesdropping in which the attacker makes independent connections with the victims and relays messages between them, making them believe that they are talking directly to each other over a private connection when in fact the entire conversation is controlled by the attacker (Venkatramanayya, 2007).

Not only is it important to protect this information, but it's also important that access to sensitive areas of your website is protected against users you don't trust.[1]

Different methods are used to secure the data that is given to other people. To be able to secure the transaction a number of things are possible this include the use of passwords such as pins, pin codes, transaction information, the use of a challenge such as mother's maiden name, the use of certificates or even the use of tokens. Many of these methods have been in existence for a long time but have challenges that need to be overcome.

[1] 1http://www.justice.gov/criminal/fraud/websites/idtheft.html

Many of the Methods have been compromised in one way or another. Some of the recommendations that are given by ITU are the use of two factor authentication (Bauer, 2008). Two factor authentication requires that there are two pieces of information that are of different category that are also store in different locations. So to the earlier methods mentioned they suggest the use of information that one knows such as the password and something that one has. The something that one knows exists in the systems such as user names and passwords. The passwords or pass codes are enhanced by the use of something that one has such as the use of tokens such as smartcards.

Methods that are used to secure transactions are often sophisticated and cost a lot of money in hardware, infrastructure and software costs. The cost of protecting identification costs from US$9 to millions of dollars[2].

Some of these methods that are used come in different forms to secure a website for example using the tokens that are already in the market are very expensive. The Owner of the site in most cases needs to rely on infrastructure belonging to third parties this includes companies like VeriSign. The company implementing this policy needs to have at least three to four software that is required to run the security systems.

Each of the software will have a one off cost as well as recurring costs such as licenses and annual maintenance fees that can be prohibitive to the customers.

Different types of attacks based on stolen identity or diversion of commands do billions of dollars’ worth of damage each year, according to Gartner. Interception services catch hundreds of thousands of "phishing" attempts each month in the United Kingdom alone, but many more go undetected. There are numerous case of fraud that each run into millions of dollars. Enterprising hackers stole identities of online brokerages using "man-in-the-browser malware. A different scheme intercepted utilities payments made through a bank, and increased the sum. The thieves then requested that the banks send refunds to their own bank accounts

Transaction Verification

Once the data has been stolen the data could be used to alter the current transaction or do other transactions at a later time to prevent this we propose to use a confirmation that is sent to the users phone instead of the

[2]. Man-in-the-browser attacks are common with banks and their customers all over the world they find the online industry exposed and take the industry totally in surprise. These sophisticated attacks occurred they alter the customer's original intention in a way that could not be detected. Attacks of this type piggy back on the customer's original transaction such as transferring money from their account to another account; in fact any transaction that touches money is exposed to attacks of this type. The malware is then in a position to change the amount to be transferred and the target account for that transfer without showing any of its alteration on the screen. Actually it does what it does right after the customer hits the Send key and before that transaction is encrypted by the SSL.

[2] www.identitytheft protection.com

For more sophisticated banking system that replies to the customer with a screen on which the requested transaction is detailed and asks for confirmation, the malware again changes the details so they reflect original intention of the customer.

Right after the customer confirms the data that they see, which comply with their original intention, the malware tampers with it again to fit the original tampering, making a mockery of the security confirmation process.

This Project has demonstrated that Small companies that provide financial services do not have to spend so much while doing there transactions. The cost that is incurred in development and the recurrent cost of software licences can be reduced significantly by using the findings in this project.

1.2. Statement of the problem.

The study aimed at studying security challenges that arise from the use of the internet for banking purposes. The study looked at the challenges of offering strong authentication at the time a client logs into the system cost effectively.

The dangers that arise when on is using the internet to do financial transactions falls into two major categories these are identity theft and man-in- the- middle- attack.

Identity theft leads to crooks opening accounts under ones names, using utilities, counterfeit checks, making ATM withdrawals, taking out loans or even obtaining government benefits. Man-in-the-middle attacks leads to transactions that are not authorized being done on behalf of the customer without their knowledge.

1.3. Objectives

To Develop a system that can be able to provide two factor authentication using a device that most of the clients of a bank already have at a minimal cost or no cost at all.

1.4. Specific Objectives

1. To create a solution that works with any type of phone available in the market thus eliminate the addition hardware and software that a consumer of a site would require to log into different sites to securely log in.

2. To create a solution that does not require hardware installations before a client can use the application thus reducing the cost of implementation and maintenance that come with proprietary based systems.

3. To create a solution that will be used to strongly authenticate the users of a website.

4. To reduce the impact of the man in the middle attacks. Reduce the vulnerabilities caused by man in the middle attacks.

5. To create an application that has significantly lower cost implications to the clients accessing it by eliminate product proliferation that is required by many security products this could range from between 4-5 different software to allow them to use tokens such as cards, USB keys.

1.5. Research questions

1. Is it possible to reduce the impact of man-in- the- middle attack by alerting the owners of accounts early?
2. Is it possible to reduce significantly the impact of the man-in-the-middle attack to engage in business in reduced costs?
3. Is it possible to generate software that will not require other support software's to be able to work in most web environments.

1.6. Justification

Methods that are used to secure transactions are often sophisticated and cost a lot of money in regards to hardware, infrastructure and software costs. The cost of purchasing solutions for protecting identification is enormous both at the time of purchasing and maintenance. Solutions provided by companies like VeriSign, now Symantec involve the use of tokens such as security cards flash drives, as well as biometrics. Clients usually end up carrying lot gadgets for authentications when it comes to the tokens. The Cost of obtaining the devices is prohibitive to many companies. Some of the solutions that are available on mobile phones are limited to the types of phone that one can use. The solutions are usually hosted at a site away from the users increasing the cost of doing business.

The study looked at an alternative method of securing transactions especially for the small and the medium enterprises. The solution design reduced on the number of applications were required to secure a transaction and hence made the application affordable. The application eliminated the need to carry tokens were only required for authentication. The solution will reduced the cost significantly of obtaining a secure session with online systems. The research presented tried to solve the problem of identity theft at the time of log in as well as tried to alert a user of the transactions that are going on their accounts before confirming transactions. The applications will eliminate the cost of third parties during transactions reducing the cost of doing business.

1.7. Limitations of the study

Due to limitations of time the study was limited to a banking application even though there are many institutions that offer financial transactions online due to time.

The application was limited to transactions that relate to the transfer of cash from one account to another.

2. Literature review

2.1 Overview of Chapter This chapter will look at the work that has been done previously in this area and highlight a gap between what has been and the current study.

2.1. Previous studies

A number of studies have been carried out in the field of security. Chepken (2006) looked at the challenges that are involved in cellular communication system. In his work he developed a system that gathered information related to a transaction (balance Enquiry) that was done. He gathered information that was related to the phone that was in use, the interface as well as the number of logins. In his study he aimed at securing a transaction that was done between a merchant and a mobile user and securing this transaction.

Walter (2008) says that some of online banking attacks include Phishing which involves motivating a user to enter confidential information on fake websites. The confidential information is then used for a number of reasons such as to open accounts, selling the accounts information to the underworld (VeriSign 2008)

Bauer (2008) looks at Malware as one of the major problems. The goal of the malware is to steal financial information and other personal information. To do this Key loggers and Trojans are used. Botnets are used to trick users into revealing their personal information.

2.2. Goals of security

The goals or strategies for security that may be used together or separately have been defined as

- Prevention this is to try and prevent the attack these strategies have been defined as cumbersome to implement by Venkatramanayya (2007).
- Detection it is useful when an attack cannot be prevented. The mechanism work on the principle that an attack will occur, the goal is to determine that the attack is under way ,or has occurred and report it.
- Recovery it can take two forms it could at first stop an attack and assess the damage caused by the attack. The second form of recovery is to identify the weak sport that were used by the attacker to enter the system and effectively handle them.

2.3. Authentication

Venkatramanayya (2007), asserts that computer security rests on confidentiality, integrity, and availability. These are the components of security. He defines confidentiality as the concealment of information or resources he goes ahead to explain that the need for keeping the information secret arises from the use of computers in sensitive fields such as the government.

In the discussion Venkatramanayya (2007), defines integrity as the trustworthiness of the data or resources, and is usually phrased in terms of preventing unauthorized change. Integrity includes the content of the information and origin integrity. The source of the data is called authentication. The source of data then determines the accuracy of the data as well as the credibility.

Authentication affects the basic component of integrity, where the source of the transaction needs to be authentic. The federal authentication institute defines authentication as “the verification of identity of by a system based on the presentation of unique credentials to that system. It can be in the form of something that the user knows or something that they have and or the user is. There could be shared secrets, tokens or biometrics.

2.4. Multifactor Authentication

This is authentication that relies on more than one form. It can use shared secrets as in passwords or keys as well as something that one has such as finger prints. Security challenges that would be in an internet banking application

Online computing does not benefit from the physical security and controls that computing and communication devices that are available. It is used by people whose actions cannot be controlled by the institution.

The systems that are used for online transactions are faced with threats. A threat is defined as Potential violation of the threat. The activities that could occur are referred to as attacks Venkatramanayya (2007),

Venkatramanayya (2007), divides the threats that affect computer security components - confidentiality, integrity and availability into four broad classes these are:

1. Disclosures

   This is unauthorized access to information it can take different forms such as snooping or the unauthorized access to information. Man- in- the middle- attacks affects disclosures

2. Deception

   This can also take different forms such as modification or alteration of information. The modification of the data could have the following goals it may affect the action that should be taken. Modification is active. Masquerading or spoofing is also form of deception. A user thinks that they are using one system but in reality they are using a different system.

   Other forms of deception that are outside the scope of this study are denial of receipt this affect integrity and availability.

3. Disruption Prevention and interruption of correct operation.

4. Usurpation (unauthorized control of some part of the system).

   Delay or the temporary inhibition of service is form of usurpation as well as deception. A delay for example in the authentication of a user can force a user not to use a system at the time that they required to.

Wuust (2010), outlines the threats into the following categories: local attacks where the attack resides in the local machine of the user. Usually between the computer and the financial institution there will be an SSL. The SSL secures the channel where the data will go through but not the data that is on the computer itself. Trojans are able to intercept any information that is entered on a webpage before encryption. The Trojans can be injected directly into a browser memory space and also be used to bypass the local machines firewall. The virtual keyboard is another method of collecting data that is on system as way of protecting against key loggers fake we popups and malicious code can still be able to get to the data. To mitigate against this some financial institutions have introduced USB keys or smartcards that can be used for securing the data. This threat described by Wuust falls under the disclosures.

Wuust (2010), classifies the attacks that occur in online financial transactions into three, namely Remote attacks, Local attacks and joint forces attack.

A local attack involves the use of Trojans and malwares that affect the computer that the user is using. Most users believe that if they see a padlock sign their computers are protected. Most of the banks secure the connection between bank and the computer at the point of transmitting the data through the use of SSL. The Trojans generate POP up window which overlays the current browser. Kharouni (2012) explains the way automatic funds transfers are affected by this.

This involves the user of a copy of the webpage to impersonate a webserver. The attacker sends emails to the user to authenticate the data or require the user to enter the website at the time the location of the real server is masked the authentication is made to look like that of the impersonated domain. Once the user is able to enter the spoofed website the data is picked as the data is entered. Trojans like PWSteal.Bankash exploit this fact.

Joint forces, the attacker combines both the local and the remote attacks. The attacker then act as a man in the middle. This is usually more harmful Wuust (2010)

2.5. Authentication methods in use

Online systems mostly use shared secret systems such as passwords. They identify the user of the system by the knowledge that they have. These includes strategies such as passwords, pass phrases, account balances or even some transaction event such as location or time. The strength of this method is on non- disclosure of and about the secret.

Different methods are used to secure the data that is given to other people. To be able to secure the transaction a number of things are possible this include the use of passwords such as PINS, PIN codes, transaction information, the use of a challenge such as mother’s maiden name, the use of certificates or even the use of tokens. Many of these methods have been in existence for a long time but have challenges that need to be overcome.

Many of the Methods have been compromised in one way or another. Two factor authentication requires that there are two pieces of information that are of different category that are also store in different locations. So to the earlier methods mentioned they suggest the use of information that one knows such as the password and something that one has. The something that one knows exists in the systems such as user names and passwords. The passwords or pass codes are enhanced by the use of something that one has such as the use of tokens such as smartcards. Methods that are used to secure transactions are often sophisticated and cost a lot of money in hardware, infrastructure and software costs. The cost of protecting identification costs from US $9 to millions of dollars3

Some of these methods that are used come in different forms to secure a website for example using the tokens that are already in the market are very expensive. The owner of the site in most cases needs to rely on infrastructure belonging to third parties this includes companies like VeriSign. The company implementing these polies needs to have at least three to four software that are required to run the security systems.

Each of the software will have a one off cost as well as recurring costs such as licenses and annual maintenance fees that can be prohibitive to the customers.

2.6. Two Factor Authentication

Two factor authentications is a method that has been advocated by a number of bodies as a way of authenticating the users of a system such as the Central bank of Kenya. These methods include the use of two channels to authenticate a system. The channels used could be different depending on the website. Smartcards, tokens or even biometrics could be used as methods of authenticating transactions. These methods many of them include the use of many other software as well as hardware items. This will increase the cost of securing the transactions the federal financial institutions examinations body defines authentication as the verification and identity by a system based on presentation of unique credentials to that system. The validation can be something that the user knows, or they have or something that the user.

Paget (2012), writes about the strong password authentication and highlights the methods of authentication to be the following when one uses more than one method of authentication then this is referred to as multi factor authentication. The authentications that are currently in use one of the following:

What the user knows: Password, PIN, secret question that the user has: cards, authenticators, Certificates[3] What the user is : Biometric

For strong authentication to be used at least two of this factors should be available. These factors are used to create one time passwords.

[3] www.identitytheft protection.com

Two factor authentication involves the uses of two forms of authentication this is the use of something that one knows or has such as a password and something that someone has like a card or a token. Using some of the items mentioned one is able to generate one time passwords (OTP) that are flexible. The passwords that are generated are one time. The password uses something that someone has in this project this would be the Mobile phone. The second one is something that someone knows this is usually something like a password.

Authentication can also be done using Knowledge based authentication. This involves the use of question such as what is your mother's maiden name , or what city were you born . The current practice is to use dynamic questions that are not stored. These would be questions that are related to transactions for example that one did in the recent past (Paget, 2012)

In a research done by (Vasquez, Martha;, 2010), she found the drivers for one time password market to be as shown in Table 1.

RANK	**Drivers**	**Year1-2**	**Year 2-4**	**Year 5-7**
1	Weakness of Passwords	High	High	High
2	Compliance with Legislations	High	High	High
3	Regulations and Standards	High	High	High
4	Need to Authenticate and Secure Remote Access Users	**High**	High	High
5	Identity Theft and Phishing Attacks	Medium	Medium	Medium
6	Cost of Supporting Passwords	Medium	Medium	High
7	Efficiency of Digital Signature Laws and PKI Medium	Medium	Medium	High
8	Ease of Use of Tokens	High	Medium	Medium

Table 11: **Drivers for one time passwords (Vasquez, Martha;, 2010),**

2.7. Challenges with the methods that are currently in use

Authentication methods that are currently in use such as the use of Smartcards and the USB keys require that the organization that is securing transactions use millions of shilling to be able to do this. For example is using smart cards one has to access the Smart Card readers or get a machine that can be able to read the smart card. For the terminal work software's have to be installed either in the server machine or in the local machine in some cases this has to be done on both systems.

VeriSign and now Symantec that bought the validation business one has to down load the application to the Mobile phone in case one is using a mobile phone. The challenge for example in the case of VeriSign is that the phone that is to be used needs to be able to take JAVA apps which cannot apply to all the users.

In the findings by (Vasquez, Martha;, 2010) in her findings she found that the use of password was weak and therefore there was a market for authentication to be used.

Other drivers according to her were interoperability of the software , Cost as well as brand names. The brand names involve the companies that are making the one time password as well as the tokens

2.8. Attacks in online banking

In online banking threats can be divided into two major categories according to how they are delivered that is either push or pull. Push threats use fraudulent methods to lure the users to malicious website or inject malware this includes threats such as Session Hijacking, Malwares affect the systems in different ways. There are malwares that are used to access the data that is contained in messages that are sent and use the data to do different things. These class of malware in include;- Man-in-the-browser, Man- in- the- Middle, Session Hijacking, Pharming, Form Tampering, Cross-Site Scripting, SQL Injection

Others get the data as it input into a computer or a device or trick a client into giving information this kind of attacks include Key Logging, Phishing

Some of the attacks may not take information but may affect the transactions that are taking place, for instance they may prevent the access to the website such as application Denial of Service attacks.

2.9. Integrity Policies used when designing systems

To be able to preserve the integrity of data (Lipner , 1999) defines five requirements for commercial application where online banking applications fall. These requirements are:-

1) users will not write their own programs, but will use existing production programs and databases

2) Programmers will develop and test programs on a nonproduction system; if they need actual data they be given production data via a special process

3) A special process must be followed to install a program from the development system to production

4) The special process must be controlled and audited

5) Managers and auditors must have access to the system.

From the above requirements several principles of operation have been identified. These include separation of duties, separation of functions and auditing. To be able to build secure systems several models have been suggested they include:

- **Biba integrity model**

In the model a system consists of subjects(S) and a set of objects (O) and a set of objects (I). The levels are ordered. This model was used by POZZO and Gray with the goal of limiting the execution domains of each program to prevent untrusted software from altering data or any other software.

- **Bella- LaPadula model**

This model is good with military type of systems where information is categorized and accessed. This becomes difficult for commercial software where limited amount of information is public and a large amount is sensitive. For example a credit cards number is public. These small items of information can be added up to give some sensitive information.

- **Clark- Wilson Integrity Model**

This model was developed in 1987. It uses transactions as the basic operation. The model simulates the commercial applications more realistically.

Commercial applications are more concerned with the integrity of data in the system and the actions that are performed on the data. The data is said to be consistent if it satisfies some given properties.

If D is the amount of money deposited today, W is the amount of withdrawn today. YB the amount of money at the close of business yesterday and TB is the amount of money in all the accounts today.

The consistency formula then is:-

$$\mathbf{D + YB - W = TB}$$

This consistency condition should hold at all times. Once a customer does a transaction in internet banking a well

Formed transactions all the series of operations should leave the system in a consistent state.

The integrity of the transactions is also catered for by this model by requiring the separation of duties. In the online transactions this will be done by requesting that the information that will be sent as a request is validated using data from a mobile phone. The request for the validation of the transaction will be done by the person who initiated the transaction should be done by a second person who did not initiate the transaction.

In this model Clark- Wilson (2007) defines data subject to be the integrity controls as constrained data items or CDI's. Data that is excluded from the controls is called unconstrained data items. (UDI). The balance of a bank account is a CDI integrity of the balance is crucial to the operations of the account.

The model further defines two sets of procedures: -

- **Integrity verification procedures(IVPs)** (checking that the account is balanced is IVP)
- **Transformation procedures (TPs)** (the process of moving money from one account to another account in this context will be the TP)

  They change the state of the data into from one valid state to another TPs are used to implement well-formed procedures.

Certification rules that are derived relevant to this project:

Certification rule 1: when any IVP is run it must ensure that all CDI's are in a valid state.

Certification rule2: A TP must transform the CDI's to a valid state.

Certification rule 3:- the allowed relations must meet the requirements imposed by the principle of association of users.

Enforcements

Rule 1 for this to ensure conformity only TPS's certified may run on a CDI. For this system the enforcement will be on the person that is doing the transaction and authorizing the transaction

Rule 2 a user must be associated with a TP- transaction

The system must authenticate a user attempting to execute a TP.

Rule 3

Any TP that has a UDI may perform only valid transactions or no transformations for all the possible values of the UDI. This may lead to acceptance or rejection of a transaction.

2.10. Security Measures that are taken by Financial institutions

(Vasquez, Martha;, 2010) highlights that some of the measures that are taken to protect data are included in the table below.

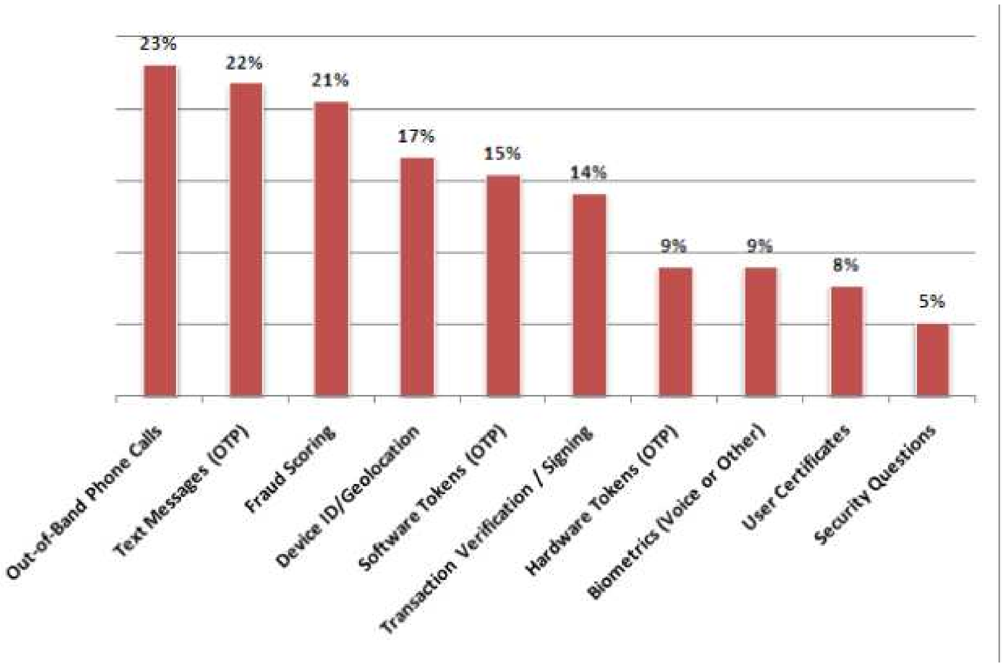


Figure 2: Methods that are in use currently (Vasquez, Martha;, 2010)

2.11. Conceptual Framework

The application will use the Onion ring for its security purposes and will have the following components interacting**.**

The application will have the following interfaces

1. A web interface that will be used by clients to do transactions and the back office staff to create the users of the system.
2. An SMS gateway that will be used to send SMS to the customer with the Log in code and also send a Validation request to the customer

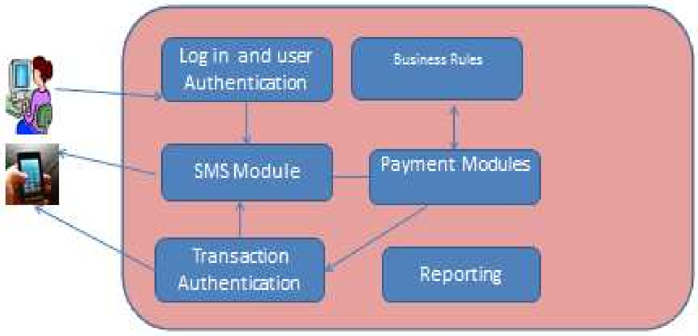


Figure 3: Conceptual Model

A user will log into the application and enter a user name and password. The system will validate the user name and password. If the user name and ID are verified to be correct the user will receive a code that he will need to enter to be able to enter the data into the system.

Once the user enters the details of the payment and completes a transaction the user will get a text on their mobile phone that will need to be entered to be able to validate the transaction that was done, the code will be sent to the person who performed the transaction for an account that is held by one user or to another account holder.

The system will require the details of the transaction to be validated before they are committed to the database.

3. METHODOLOGY

Comment [p1]: Chapter 3 is methodology

The application will have a web interface that the user will access through the interface will allow the customer to send all the information to a central location. The application will generate pass key and send it to a customer through a mobile phone. The customer is supposed to use the passcode sent to their phone to log in.

After the customer has entered the data there will be an engine that will check if the business rules are followed according to the customer if there are not a notification will be sent to the customer declining the transaction.

Below is a diagram showing the relationship of the various interfaces.

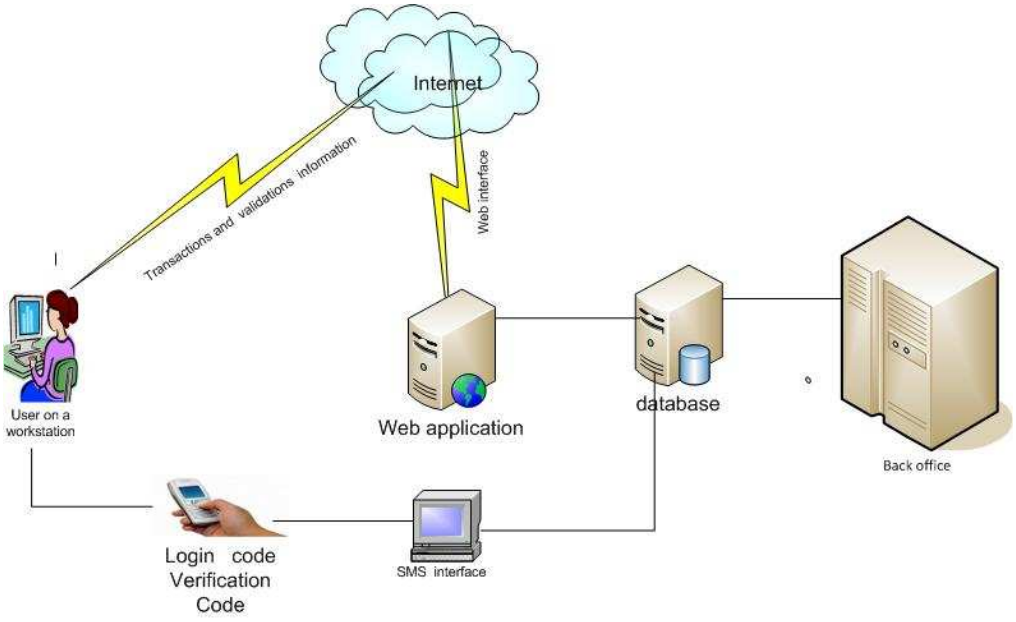


Figure 4: Architecture

Methodology

The methodology to be used for this project will be Rapid application development. James Martin, in his book first coining the term, wrote, “Rapid Application Development (RAD) is a development lifecycle designed to give much faster development and higher-quality results than those achieved with the traditional lifecycle. It is designed to take the maximum advantage of powerful development software that has evolved recently.” Whitten, 2007.

RAD places a lot of emphasis on the user involvement. The method is sometimes called the Spiral approach because one has to repeatedly spiral through the phases.

The basic idea of the RAD are:-

1) To involve the user actively through the various stages of the system development
2) Organize the system into series of focused intense workshops
3) Accelerate the requirements and design phases
4) To reduce the amount of time that one takes to work on the system.

3.1. Steps in RAD

1. Requirements Planning (the steps explained in RAD

Also known as the Concept Definition Stage, this stage defines the business functions and data subject areas that the system will support and determines the system's scope. In this project this will be represented by this the proposal presentation that will include the scope of the project and the time lines that are involved.

2. User Design

Also known as the Functional Design Stage, this stage uses workshops to model, the system's data and processes and to build a working prototype of critical system components. In this project the workshops will be represented by the time taken with the supervisor as well as the Projects evaluation panel. It comprises of the problem analysis, and decision analysis

3. Construction

Also known as the Development Stage, this stage completes the construction of the physical application system, builds the conversion system, and develops user aids and implementation work plans. A prototype will be built at this stage and it will include iterations until the requirements are achieved.

Design will involve the use of models and this will be analysed to ensure that the system meets the required standards.

4. Implementation

Also known as the Deployment Stage, this stage includes final user testing and training, data conversion, and the implementation of the application system.

The Proposed system is the Rapid application Development. James Martin, in his book first coining the term, wrote, "Rapid Application Development (RAD) is a development lifecycle designed to give much faster development and higher-quality results than those achieved with the traditional lifecycle. It is designed to take the maximum advantage of powerful development software that has evolved recently."

Also known as the Deployment Stage, this stage includes final user testing and training, data conversion, and the implementation of the application system.

3.2. Justification for the methodology

1 Reduction in Development time due to the use of the prototypes

2 A user will be able to visualize the system and give the feedback,

3 Prototyping allows for change in case the requirements are not clearly understood

4 Allow for quick feedback as errors and omissions tend to be detected earlier in prototypes.

3.2.1. Disadvantages of using the method

Whitten, 2007 pin pointed some of the drawbacks of the method as the following:

1) The code is generated using the "code, implement and repair mentality this increases the cost of development of a period of time
2) The method may discourage the analyst from looking at other solutions
3) RAD emphasis on speed can impact on the quality.

3.3. Requirements gathering

A case study of an already existing system was used to collect the requirements that are required for an internet banking application. The sample was picked off a bank that had put a sample of their application online.

The requirements were improved using the information that had been gathered during the literature review some of the requirements included standards and best practices that had been offered to be able to achieve the security systems thresholds.

The application developed was a web based application that contained the modules that are found in a normal banking application.

The modules that will be included fell into two categories, non – financial and financial modules. The

An Interview was also used to gather data from users who already have Internet Banking.

The questionnaire was used to get information on areas of usability and security that is place.

3.4. Scope analysis

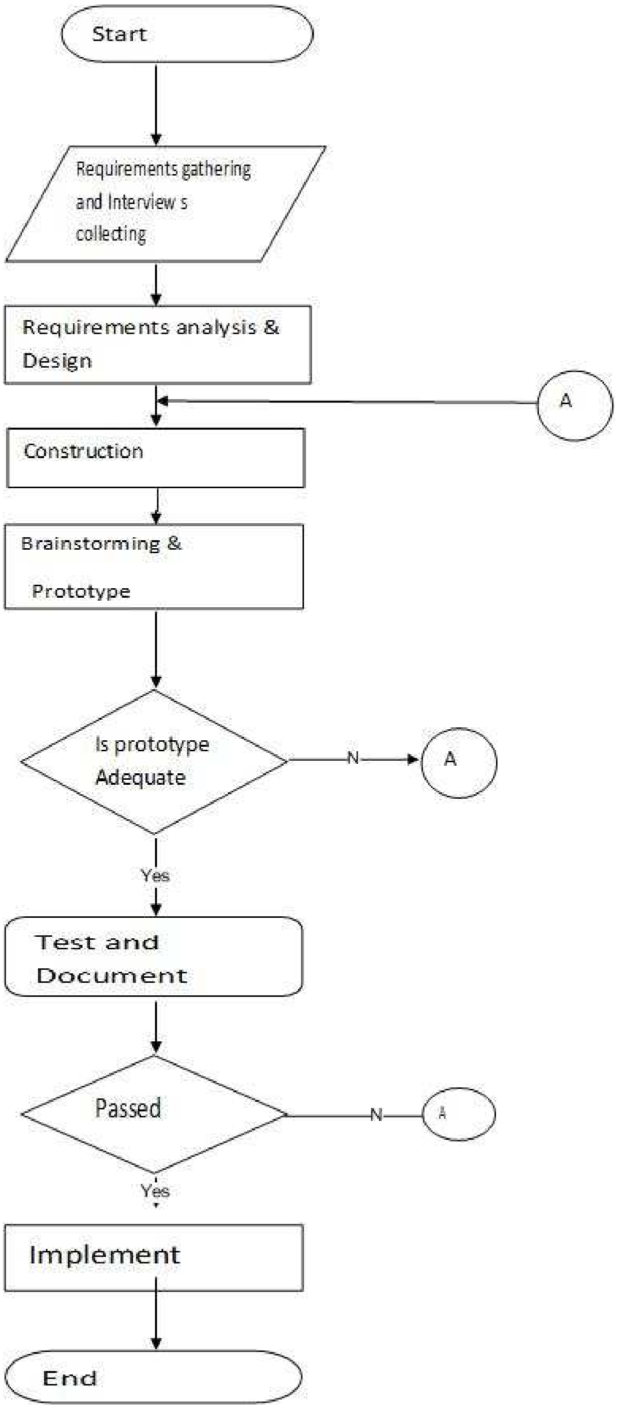
- Develop a prototype Internet banking solution
- Develop a prototype application for validation purposes that sends a message to a mobile phone
- Deployment of the application

Modules in the application

- Financial Transaction
    - Transfer own accounts

    - Transfer other accounts in the same account
- Non-financial
    - Stop Cheque
    - Statement request

Figure 5: RAD STEPS USED IN THE APPLICATION


Start
Requirements gathering and Interview s collecting
Requirements analysis & Design
A
Construction
Brainstorming & Prototype
Is prototype Adequate
N
A
Yes
Test and Document
Passed
N
A
Yes
Implement
End

4. Chapter 4 System Design

4.1. system design

The application has three major interfaces this are:-

Web application

The web application provided the user interface that had modules that contained all the transaction that will be provided in the prototype. It will contain the log in screen and the confirmation of the module. The modules contained will be back office operations that ordinarily would be done by the bank staff and front office modules which would contains the modules that are used by the client.

SMS module

This will be a module that will send an SMS to a phone that is registered in the system. The module will also be used before the transaction is committed in the database to validate the transaction. Authentication and This module will be used to authenticate a user and send a confirmation to the user.

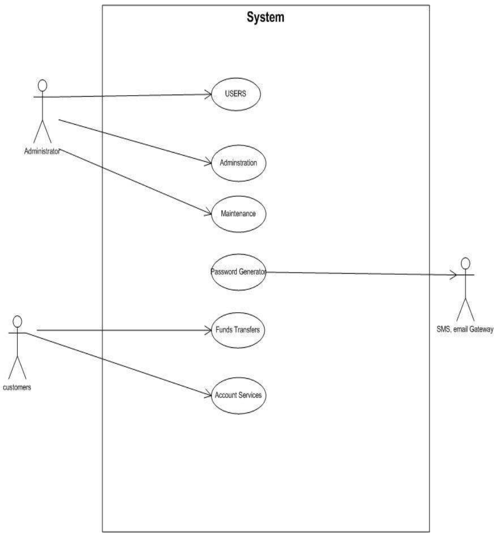
Validation module

This module will be used to do validation for the transactions that have been done. A code is sent to the owner of the transactions so that the user can be able to confirm that the details that have been entered are correct before the transaction can be completed. The details that need confirmation include the amount and the destination account that is usually modified during the Man- in-in-the middle- attacks.

Figure 6: Software Architecture

4.2. Web application

The web application will have the following Actors and components



Transaction Data

Figure 7: Application Use Case

The application has an administrator who will be a bank official who does background work such as the creation of accounts that are to be used as well as add the customer to the system. The administrator creates all the users of the system and manages them as well. For instance if the user does not use their system for a period of time that user is blocked to avoid some else using their system.

The Administrator also is also responsible for creating the Parameters that are suitable to the system for example the location of the database, the SMS gateway location among other parameters

Once a user logs in the system the system generates a Password that is send an SMS to the user. The user proceeds after entering the right code sent to their Mobile phone.

The customer will do both Financial and non-financial transactions. The financial transactions include funds transfer.

The software Architecture will be as follows:

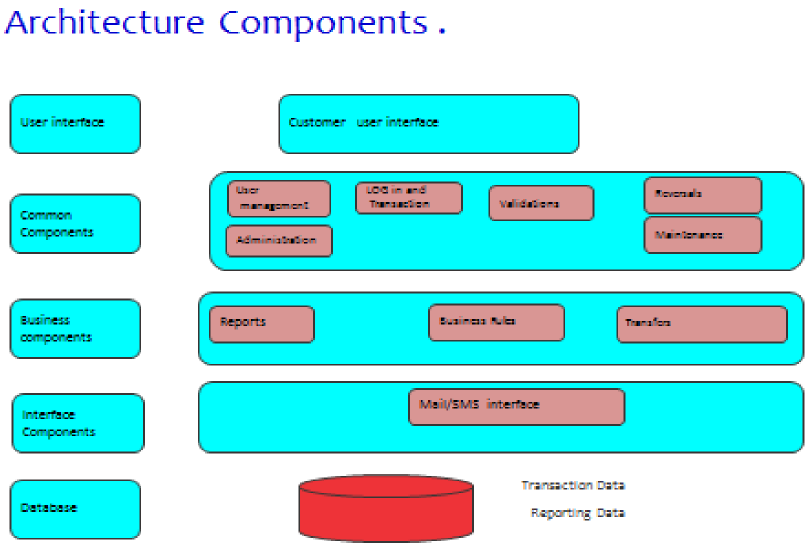


Figure 8: Architecture Components

Architecture components Explained & Technologies used

The user components were developed using JSPs. These were all the input screens and t displays that were in the system. The system was designed to have the same look and feel so that it is easy to learn once a user has done one functionality.

Common components include user management, system Maintenance and transaction validation

These contain the functions of the application. The functions that are used include:

The business components contain the rules that are used when performing the transactions. Some of the business rules that were included were on the password , instance if a user logged into the system and entered a wrong password three times the password is locked, if a user also entered the wrong PIN code that PIN code is flagged and a another one is sent The application is coded using Java and specifically using Spring MVC that provided the framework for coding

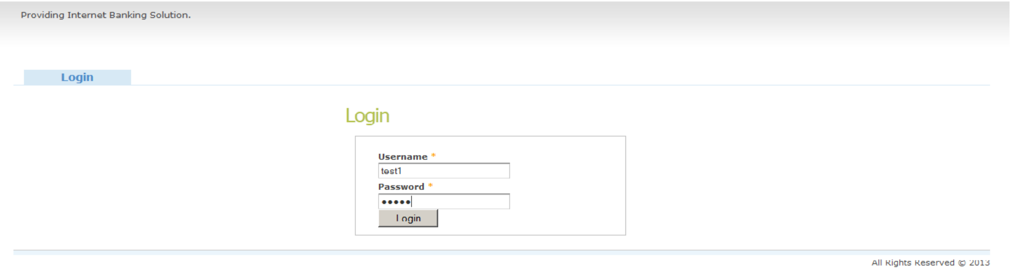
The Application interfaces with an SMS module that is used to send Login credentials to customers.

Connection to the database wad through Hibernate and the data base used is MySQL.

4.3. Interface Design

The following are some of the input screens that are found in the application.

This screen is used to enter the user name and Password.



Providing Internet Banking Solution.

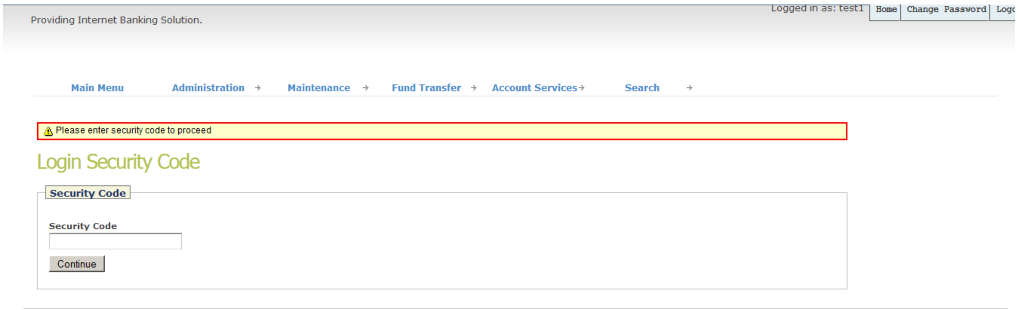
Login

Username * test1

Password * •••••

Figure 9: Log in screen

After successful login the application will request for the user to enter the code sent to the mobile phone that is registered in the application



Providing Internet Banking Solution.

Logged in as: test1

⚠ Please enter security code to proceed

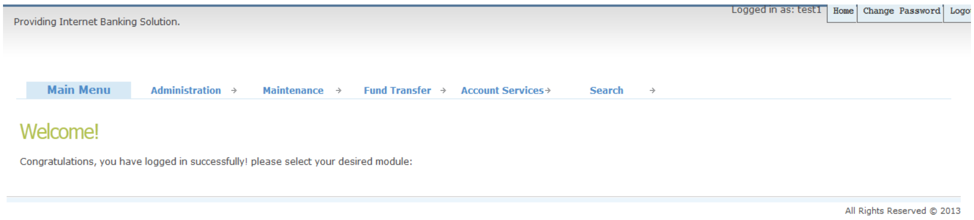
Login Security Code

Security Code

Figure 10: Security Code

Once the system confirms that the Security is the same as the one sent the application will allow the user to see the modules that are assigned to them.

The following screens relate to the activities of a customer.



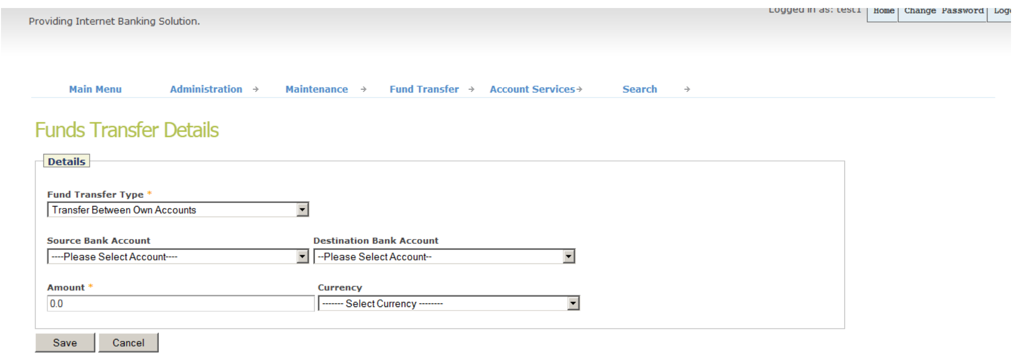
Providing Internet Banking Solution.

Welcome!

Congratulations, you have logged in successfully! please select your desired module:

All Rights Reserved © 2013

Figure 11: Transfer Internal Funds Transfer



Providing Internet Banking Solution.

Funds Transfer Details

Details

Fund Transfer Type *
Transfer Between Own Accounts

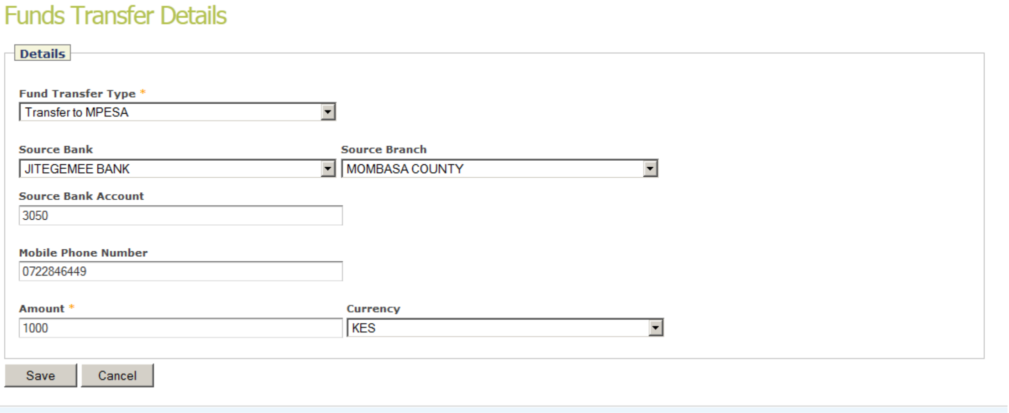
Source Bank Account
----Please Select Account----

Destination Bank Account
--Please Select Account--

Amount *
0.0

Currency
------- Select Currency --------

Figure 12: Funds Transfer to other parties



Funds Transfer Details

Details

Fund Transfer Type *
Transfer to MPESA

Source Bank
JITEGEMEE BANK

Source Branch
MOMBASA COUNTY

Source Bank Account
3050

Mobile Phone Number
0722846449

Amount *
1000

Currency
KES

Figure 13: Funds Transfer details

Once this transaction is done it saved but it is not committed to the main system. for accounts that have more than one person operating a second person will receive a message to confirm the transaction . Before it is completed

Authentication Process flow

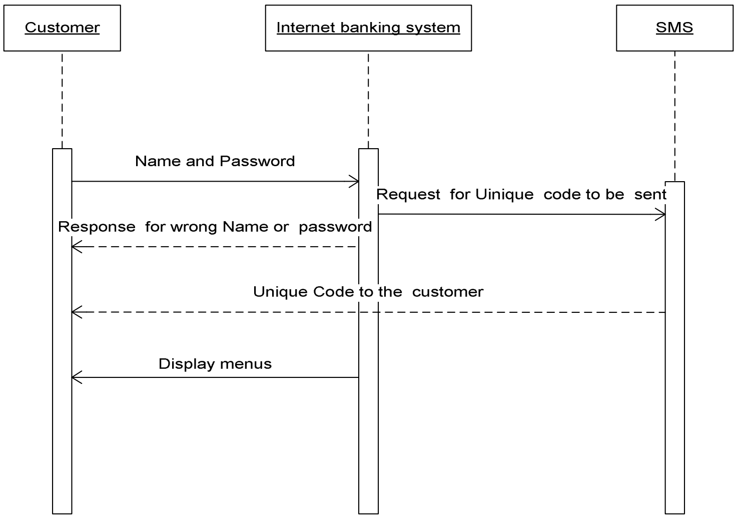

Customer
Internet banking system
SMS
Name and Password
Request for Uinique code to be sent
Response for wrong Name or password
Unique Code to the customer
Display menus

Figure 14: Authentication Process Flow

FUNDS TRANSFER - MAKER (Person Originating the transaction)

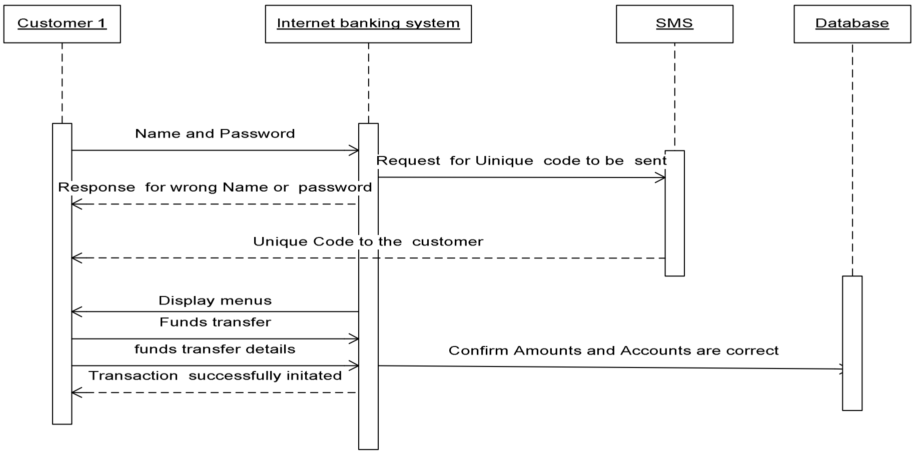

Customer 1
Internet banking system
SMS
Database
Name and Password
Request for Uinique code to be sent
Response for wrong Name or password
Unique Code to the customer
Display menus
Funds transfer
funds transfer details
Transaction successfully initated
Confirm Amounts and Accounts are correct

Figure 15: Funds transfer Completion

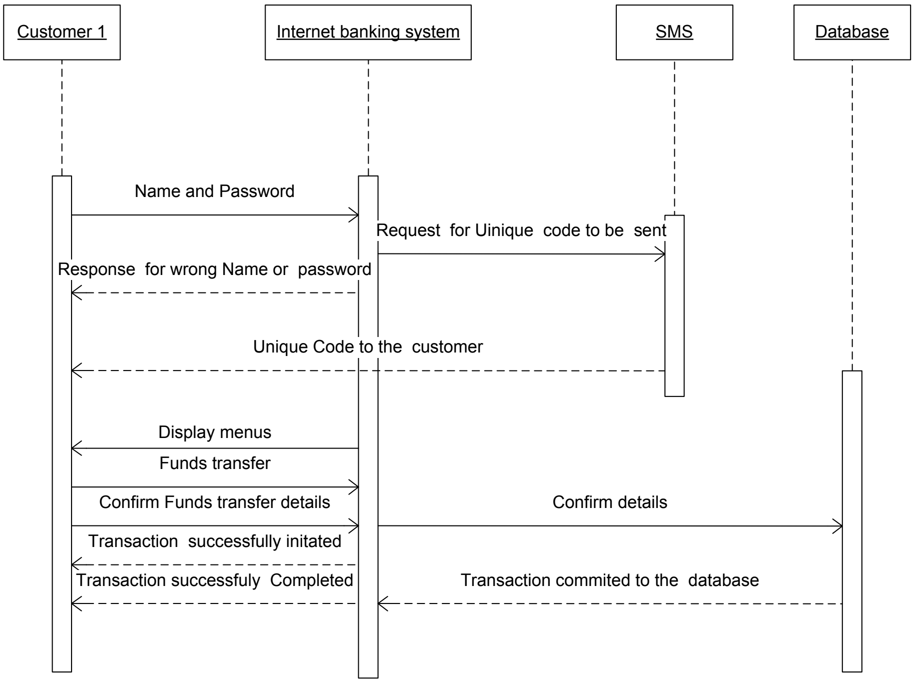

Customer 1
Internet banking system
SMS
Database
Name and Password
Request for Uinique code to be sent
Response for wrong Name or password
Unique Code to the customer
Display menus
Funds transfer
Confirm Funds transfer details
Confirm details
Transaction successfully initated
Transaction successfuly Completed
Transaction commited to the database

Figure 16: Funds transfer - Validation

4.4. Testing Methodology

Software testing Software testing is the process of analysing a software item to detect the differences between existing and required conditions (that is, bugs) and to evaluate the features of the software item.

This application was tested using the following methodologies:

4.4.1. Black box Testing

Black box testing (also called functional testing) is testing that ignores the internal mechanism of a system or component and focuses solely on the outputs generated in response to selected inputs and execution conditions.

Using black box testing techniques, the application was examined for the high-level design and the r requirements specification to ensure the code did what it was intended to do. Functional testing involved ensuring that the functionality specified in the requirement specification works. System testing involved putting the new program in many different environments to ensure the program worked in a typical customer environment with various versions and types of operating systems and/or applications. System testing is testing conducted on a complete, integrated system to evaluate the system compliance with its specified requirements. The following were the requirements that were used for the black box testing

- Requirements
- Functional specifications
- High-level design documents
- Application block source code

4.4.2. White Box testing

White box testing (also called structural testing and glass box testing) is testing that takes into account the internal mechanism of a system or component. White box testing assumes that the tester can take a look at the code for the application block and create test cases that look for any potential failure scenarios. During white box testing, you analyse the code of the application block and prepare test cases for testing the functionality to ensure that the class is behaving in accordance with the specifications and testing for robustness. Microsoft MSDN outlined the requirements for white box testing as:

The following input is required for white box testing:

- Requirements
- Functional specifications
- High-level design documents
- Detailed design documents
- Application block source code

4.4.3. White Box Testing Steps

The white box testing process for an application block is shown in Figure17.

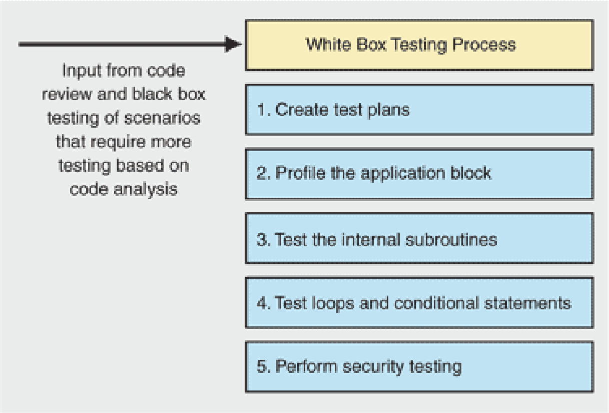

Input from code review and black box testing of scenarios that require more testing based on code analysis
White Box Testing Process
1. Create test plans
2. Profile the application block
3. Test the internal subroutines
4. Test loops and conditional statements
5. Perform security testing

Figure 17: Steps in white Box Testing

a) People Experts

People Experts were used to test the application and give their views from the perspective of similar applications as well as feedback on the look and feel.

4.5. User feedback on design

The following were the findings that were gathered from the interviews:

1. Majority of the banks had both retail and corporate customers using Internet Banking. The banks surveyed except for one had both retail and corporate clients. Internet banking was provided to both Market Segments.

2. For the retail accounts most of them had one user while for the corporate customer they had more than one person who used the account that was used for internet banking.

3. For most of the banks there was a monthly cost for the internet banking that was loaded at the end of the month. Some of the banks had a onetime fee some of up to Kshs 4,000. The banks that had no cost to the server had a ledger balance that was fixed for different services that were provided.

4. For the banks that had internet banking there were some requirements that needed to be done. Most installed some software that was required to work with application as well as certificate software. One of the banks gave MacAfee antivirus at the time of the download as well as a certificate. In one of the banks they installed some applications in the machine that was to be used and gave dongles to be used.

5. For the situations where the bank was using external devices for security it was revealed that for some the process of replacing the toggles takes some time and has cost implications. One of the banks was charging 4,000 to replace the dongle. One of the banks that was using PIN codes that a client scratched the process of disabling them was quite lengthy and also had a cost implication.

6. Some of the modules were by their nature candidates for fraud such as the transfers. This according to them worked with the balance Enquiry where an enquiry was done first then a transfer

7. All the banks applied the same rules for the different types of accounts. The security strategy that was used applied to all the account. Care was also taken to educate the customers on the care that they were to take when using the internet as way of doing transactions .

8. Fraud detection took a long time for all the respondents who said they noticed when they received statements which are sent monthly. Though some of the websites had a facility for the statements it was not used frequently by the users.

5. Implementation and testing

This chapter will cover the implementation of the project. It will contain details on the platform that is was used the software , the programming tools as well as the testing that took place.

5.1. Platform

The following are the platforms that were used for the development of this project.

The application took the format of server client relationship architecture. The hardware discussed caters for both the server side and the client side.

Server hardware

Servers 1 data base server and 1 application server	
Processor	2.4Ghz (GHz) or faster 32-bit (x86) **or 64-bit (x64)** processor Intel® Core i3,
Main memory	4(GB) RAM (32-bit) or 4 GB RAM (64-bit)
Hard disk	250 GB available hard disk space (32-bit) or 20 GB (64-bit)
Monitor	DirectX 9 graphics device with WDDM 1.0 or higher driver
Networking	Ethernet card and cable
Modem	Safaricom was used

5.2. Client work Stations

Minimal requirement for the client workstation is:

Servers 1 data base server and 1 application server	
Processor	1.7 (GHz) or faster 32-bit (x86) **or 64-bit (x64)** processor
Main memory	1(GB) RAM (32-bit) or 1GB RAM (64-bit)
Hard disk	40 GB available hard disk space (32-bit) or 20 GB (64-bit)

Monitor	DirectX 9 graphics device with WDDM 1.0 or higher driver
Networking	Ethernet card and cable
Modem	Safaricom was used

Software

Software	
Database server	MySQL 2008
Web application server	Sun sever
Operating System	Windows 7 and above
Browser	Internet Explorer , Mozilla, chrome ,

Development Environment

Software	
Programming language	JAVA(JDK 1.6.0)
Development framework	Appfuse Framework Spring MVC
Database	MY SQL 2008 Hibernate
Deployment	Sun server

Justification for the programming tools.

Java and spring MVC framework were used for programming as they provided a framework and libraries that were required to be able to meet the requirements of the application.

The Framework is open source and hence getting the requirements was easier such as the SMS gateway.

5.3. Testing

The application went through the following test Phases. The following are the results of the testing that was done.

Test case ID	Action	Expected Result	Actual Results
1	Administration		
1.1	Create user	The details of the user such as the name , the address, the mobile number are successfully added into the system	The user details were successfully added to the system
1.2	Confirm the created user	A second user should log into the system and confirm the details entered by the first user	A second user was able to enter into the system and confirm the details entered by the first user
1.3	Edit user	Modify the details of the user such as the email address, the user name should not be modified	Details of an already added user were modified
1.4	Edit user	Modify the user name - the system should not allow the system to modify the user name	. It was not possible to modify the user name
1.5	Edit User confirmation	A second user was able to enter the system and confirm the details entered by a first user or reject	The details were successfully confirmed.
2	Maintenance		
2.1	Maintenance	The module was used to maintain parameters used in the system such as the data base location	The parameters were maintained and could be modified
3	Customer and Account creation		
3.1	Create customer (first user)	• Add a customer to the system by adding the customer details such as the name, the address,	The customer was added successfully

		the mobile number and the email - Enter the name of the account - Enter the Account Id	
3.2	Confirmation of the user created	A different user to log into the system and confirm that the details that are entered are correct	A second user was able to log into the system and log confirm or reject the details that were entered.
3.2	Confirmation of account created	A second user logs into the system views the details and confirms or reject the details of the account created	A second user views the details of the account created and was able to select approval
4	Authentication of the user		
4.1	Login	A user that is created is able to login into the system and enter their user name and password as the first level of authentication	The user was successfully logged in.
4.2	Security code	The user who logged in successfully receives a code on their mobile phone so that they can be able to get to their modules	The user enters the code received on their phone and was able to log into the system and view all the modules they were entitled to do
4.3	Wrong security code	When the user enters the wrong security code another message is sent to their mobile phone. The first security code should be denied access	When the security code was entered twice the user was not able to log into the system
5	Transactions		
5.1	Funds Transfer	- The user enters the	The system captures the

		account the details of the account that they want to send money from . - The customer enters the destination account - The customer enter the bank and the branch if a different bank - The customer enters the an amount that they intend to transfer and then selects save	details of the transaction that is the source account, destination account , the bank and branch and the amount successfully
5.2	Transaction validation	The user or a second user gets a message on their phone to enter the PIN code.	A message was received from the phone with PIN and the transaction details to be confirmed.
6	Validation		
6.1	Validate a transaction done	A second user is able to log into the system and confirm that a transaction was done and the details match what was sent to the phone	A second user got a text message with a pin for them to validate a transaction that had been done.
6.2	Confirm account status	The account status could be confirmed by checking the balance before the validation to ensure that the transaction still was not completed	The account did not reflect the details of the transaction before the validation.

5.4. Results and Findings

Summary From the results the following items were achieved:

1. A customer who was maintained in the system was able to log into the system.
2. They customer got a Pin that when entered into the system allowed them to be able to access other modules.
3. Most of the activities needed a second person to validate the activities done in the system. This allowed for the validation of the activities that had been done in the system
4. A second user was able to login into the system and confirm the transactions that have taken Place.
5. Transactions that had not been validated did not appear in the database until the confirmation was done. The transactions were held in a queue.

5.5. Training

With successful testing the users of the application will be done. This will be achieved by raining trainers of Trainers who will be able to train the other People.

The trainings were divided into 2 sessions. One training was for the administrators of the system and the second was the training for the end users who would train the customers.

5.6. Changeover

This being a new application the best method of changeover of choice was the Pilot changeover. This will allow the users of a certain branch to be able to start with the training as they complete the training and start using the lessons learnt will be used to train the other Branches

6. Conclusions

6.1. Achievements

The application was successfully developed and provided the following benefits

1) The application used the mobile as a token. Any mobile phone can be used in the solution.
2) No software was required on the phone or additional software except for the service program that was used for authentication
3) No hardware was required for the solution Except for the SIM card for communication Purposes.
4) The application was designed to prevent fraud by giving information before a transaction was completed.
5) The application did not require the use of third party solutions to be able to do authentication.
6) The application relied on items that were easily available and the users would be able to use therefore resistance was at a minimum.
7) The only cost that a client could incur in the transaction was the cost of the SMS that was sent to their phone. Therefore costs that were incurred like the 4,000 that was used for the maintenance of the dongle were avoided.

6.2. Challenges and Limitations

The work was limited to only a small sent of transactions that are prone to attacks. The work was only limited to integrity and authentication.

Information that was required for the study was not so easily available as many banks do not publish issues on attacks.

The use of internet banking is not widely embraced and therefore getting information on this area was also a challenge. Many of the banks who had internet banking provided a very small set of transactions. Most of the transactions that are done are generally on account information but not transactional.

6.3. Conclusion

The project was able to demonstrate that it was possible to have an inherent banking applications that can provide authentication using the items that one already have and protect the owners of systems the high costs that are usually incurred on security. The information that the client got was in real time and the customers were able to get information on the activity of their accounts. In case an account is compromised the owner of the account gets information in real time that lets them what is happening to their accounts instead of waiting to confirm the transactions at the point at which they receive their statements. There could be a lapse of time that could cause other fraudulent transactions to have taken place.

6.4. Recommendations

The solution can be used by other organizations that have sensitive data. The solution did not tackle Fraud that would be from within the bank neither did it tackle the problem in case someone was able to access ones Phone.

This would suggest having another Module that would study the behavior of people so that in case something unusual happened like transferring huge amount of money. The application can provide an Alert that is sent until the person confirms in person that they are the ones that are trying to do the transaction.

REFERENCES

1. Bauer, J.M., 2008. *ITU study on the financial aspects of network security Malware and Spam working Paper 2008*. [Online] Available at: HYPERLINK "Http://www.ITU.Cybersecurity/docs" Http://www.ITU.Cybersecurity/docs [Accessed 15 october 2012].

2. Chepken.K.C (2006), *Cellular Communication Security system on mobile communication*. University of Nairobi Masters thesis

3. Hamdan, A.O., Rami, A., Zaidan, M.A. & Yahya Al-Nabhani, 2010. *Journal of computing Volume 2 issue 5*,. [Online] Available at: HYPERLINK "www.journal of computing.org" www.journal of computing.org [Accessed 29 october 2012].

4. Hoffner , J.A., 1999. *Mordern systems Anlaysis and Design*. Newyork: Addison Wessly.

5. Kenya Bankers Association (KBA), 2011. *KBA_Bank_Branch_Listing_14th_October_2011.pdf*. [Online] Available at: HYPERLINK "http://www.centralbank.go.ke/index.php/bank-supervision/commercial-banks-mortgage-finance-institutions" http://www.centralbank.go.ke/index.php/bank-supervision/commercial-banks-mortgage-finance-institutions [Accessed 29 october 2012].

6. Lipner , s., 1999. Twenty years of Evaluation criteria and commercial Technology Proceedings of the 1999 IEEE symposium on security and privacy. *IEEE symposium on security and privacy*, pp.111-12.

7. online Banking survey, onlinebankingsecuritysurvey201_4778.pdf. *Phone factor Market Research*. [Online] Available at: HYPERLINK "http://media.scmagazineus.com/documents/20/onlinebankingsecuritysurvey201_4778.pdf" http://media.scmagazineus.com/documents/20/onlinebankingsecuritysurvey201_4778.pdf [Accessed 29 october 20122].

8. Paget, F., 2012. *Finanacial Fraud and internet Banking : Threats and counter measures*. Mcfee.

9. Vasquez, Martha;, 2010. *An overview and competitive Market Analysis of the One time Password (OTP)*. [Online] Available at: HYPERLINK "www.frost.com" www.frost.com [Accessed 29 November 2011].

10. Venkatramanayya, A.S., 2007. *Introduction to Computer Security*. 2nd ed. Kindersley: Dorling.

11. James w Kivuva Emergency Information Dissemination system p56/p/7614/06/ using sms for disaster management

12. R. C. Martin, *Agile Software Development: Principles, Patterns, and Practices*. Upper Saddle River: Prentice Hall, 2003.

13. Microsoft developer Network Black Box and White Box Testing for Application Block, http://msdn.microsoft.com/en-us/library/ff649503.aspx [accessed on 05/05/2013]

14. Wüest Candid, Phishing In The Middle Of The Stream" - Today's Threats To Online Banking AVAR 2005 White Paper: Symantec Security Response [Accessed 29 November 2011].

Appendix

Appendix 1

Interview guide

Below is the interview sheet that was used in the study

1) What type of customer is served by internet banking?
   - i. Retail
     - a) Mostly Retail
     - b) More than half are retail
     - C) Very few are retail
     - D) There are no retail customers
   - ii. Corporate
     - a) Mostly Corporate
     - b) More than half are Corporate
     - C) Very few are Corporate
     - D) There are no corporate customers

a) Retail
   - a. For the retail accounts how ,many people operate an account
     - i. One
     - ii. Two
     - iii. More than three
     - iv. There is no standard number

----

b) Corporate Accounts
   - a. Are there a minimum number of people who can operate an account?
     - i. Yes
     - ii. No
   - b. Are there any prerequisites that are required for one to operate internet banking in terms of software and hardware?
     - i) Yes II) NO

c) Corporate Accounts
   - a. Are there a minimum number of people who can operate an account?

b. Are there any prerequisites that are required for one to operate internet banking in terms of software and hardware?

ii) Yes II) NO

2) a) If yes state the prerequisites

----

b) What is the estimated cost of the that a customer incurs when using internet banking

a) No cost to the customer

b) Monthly cost of Below 1000

c) Monthly cost of above 1000 per year

d) One off cost of below 5000

e) One of cost of Above 5000

……………………………………………………………………………………………………
………………………

c) What security precautions do you take to log in to the application

……
………………………………

d) Do you find the method of logging in use friendly?

Strongly Agree	**Agree**	**Neutral**	**Disagree**	**Strongly Disagree**

e) Are the security precautions the same for the retail and cooperate customers

a) YES b) NO

----

f) Are there modules that are impacted by fraud more than others

✓ Yes b) NO

g) IF the answer is a please state the modules that are impacted.

----

h) How long does it take for the bank to detect that fraud has taken place in a certain account.

----

i) Describe the nature of frauds that take place

----

j) For question six suggestion some methods that can be used to militate against fraud.

a) By the customer

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………………………………………

……………………………………………………………………

b) By the bank

----

k) How would you wish the bank to assist you in detecting Fraud

Appendix 2

Findings from the Interview

The following are responses are from five banks that participated in the interview

1) What type of customer is served by internet banking?

i. Retail

a) Mostly Retail

b) More than half are retail

C) Very few are retail

D) There are no retail customers

Question 1	Bank A	Bank B	Bank C	Bank D	Bank E
i	C	B	D	A	C

Table 2: types of Accounts

2) Retail

a. For the retail accounts how ,many people operate an account

i. Mostly One

ii. Mostly Two

iii. More than three

iv. There is no standard number

Question	Bank A	Bank B	Bank C	Bank D	Bank E
A	Mostly One	Mostly One	Not applicable there are no retail customers	Mostly One	Mostly One

Table 3: No of people who hold the same retail account

3) Corporate Accounts

a. Are there a minimum number of people who can operate an account?

i. Yes ii) No

Question	Bank A	Bank B	Bank C	Bank D	Bank E

A	yes	yes	yes	yes	yes

Table 4: Number of people who hold the same corporate account

4) Are there any prerequisites that are required for one to operate internet banking in terms of software and hardware?

iii) Yes II) NO

Question	Bank A	Bank B	Bank C	Bank D	Bank E
B	yes	No	No	yes	yes

Table 5: hardware requirements

5) Corporate Accounts

a. Are there a minimum number of people who can operate an account?

Question	Bank A	Bank B	Bank C	Bank D	Bank E
B	yes	No	No	yes	yes

Table 6: minimum number of people who hold and account

b. Are there any prerequisites that are required for one to operate internet banking in terms of software and hardware?

iv) Yes II) NO

6) a) If yes state the prerequisites

- Take the laptop to be used to the bank for configurations
- Install Certificate that are sent on Email
- Provide Tokes or cards as per the banks policies
- Force a client to change the password at first login

7) What is the estimated cost of incurs when using internet banking

a) No cost to the customer
b) Monthly cost of Below 1000
c) Monthly cost of above 1000 per year
d) One off cost of below 5000
e) One of cost of Above 5000

Bank A	**Bank B**	**Bank C**	**Bank D**	**Bank E**

Security Certificate	Use a toggle	A security card containing codes that are used as cha		Password
Adobe PDF provided a reader for free.	Client has to take a machine for installations to be done (information on the particular was not available	Bank C	Any browser	No additional to the customer side. Hardware security installed at the server side
MacAfee Security Scan (different from what is at the client site		No	User name and Password	yes
Cost - Free to customers who have accounts where the ledger fee is above 684 per month - Retail	Initial Cost 4,000			
b) Monthly cost of Below 1000	One off cost of below 5000	1000 per month	1000 per month	The cost is not passed onto the customers
	If card is lost they don't cancel even when requested (was not able to find out the		If password is comprised fill a form the password is secured using VeriSign at	If password is comprised fill a form the password is secured using VeriSign at the log in

	reason why		the log in	

Table 7: Summary of the cost and the required software from the respondents

……………………………………………………………………………………………………
………………………

8) What security precautions do you take for the log in to the application

9) The security precautions that were taken were those that were associated with the activities that are done by the bank. Client did not try to include any more security as they used the system

10) Do you find the method of logging user friendly?

Table 8 shows the user friendliness at the time of login

	Strongly Agree it is friendly	**Agree**	**Neutral**	**Disagree**	**Strongly Disagree**
A					✓
B			✓		
C					✓
D	✓				
E	✓				

Table 8: User friendliness responses

11) Those who used the password and User name were comfortable with the method

12) The passwords were shared in some cases with the secretary who would be assigned some duties.

13) Are the security precautions the same for the retail and cooperate customers

b) YES   b) NO

The answer was yes for all the interviewees

----

14) Are there modules that are impacted by fraud more than others

a.) Yes b) NO

15) IF the answer is a please state the modules that are impacted.

Modules where funds are moved from one account to another

16) How long does it take for the bank to detect that fraud has taken place in a certain account.

At the time they read there are statements mostly

----

17) Describe the nature of frauds that take place

✓ Funds are withdrawn from the account without knowledge of the customer

18) For question six suggestion some methods that can be used to militate against fraud.

c) By the customer

- Provision of security measures to protect the customers
- Provision of a user friendly site that has information

d) By the bank

- Customers to be aware of simple security rules
- To check Transactions often a good number of customers raise the issue months after it happened.

19) How would you wish the bank to assist you in detecting Fraud

- Provision of early information in case of suspicious transactions

……………………………………………………………………………………………………
………………………

20) What security precautions do you take for the log in to the application

The security precautions that were taken were those that were associated with the activities that are done by the bank. Client did not try to include any more security as they used the system

21) Do you find the method of logging in that user friendly?

Those who used the password and User name were comfortable with the method

The passwords were shared in some cases with the secretary who would be assigned some duties.

The following table shows the user friendliness at the time of login

	Strongly Agree it is friendly	**Agree**	**Neutral**	**Disagree**	**Strongly Disagree**
A					✓

B			✓		
C					✓
D	✓				
E	✓				

22) Are the security precautions the same for the retail and cooperate customers

c) YES b) NO

The answer was yes for all the interviewees

----

23) Are there modules that are impacted by fraud more than others

a.) Yes b) NO

24) IF the answer is a please state the modules that are impacted.

Modules where funds are moved from one account to another

25) How long does it take for the bank to detect that fraud has taken place in a certain account.

At the time they read there are statements mostly

----

26) Describe the nature of frauds that take place

✓ Funds are withdrawn from the account without knowledge of the customer

27) For question six suggestion some methods that can be used to militate against fraud.

e) By the customer

- Provision of security measures to protect the customers
- Provision of a user friendly site that has information

f) By the bank

- Customers to be aware of simple security rules
- To check Transactions often a good number of customers raise the issue months after it happened.

28) How would you wish the bank to assist you in detecting Fraud

- Provision of early information in case of suspicious transactions

Appendix 1: Some of the Source code used

```
package com.nx.ebank;

import org.apache.commons.logging.Log;

import org.apache.commons.logging.LogFactory;
import org.smslib.OutboundMessage;
import org.smslib.Service;
import org.smslib.modem.SerialModemGateway;

import com.nx.ebank.webapp.listener.LoginListener;

public class SendScript {
private static final Log log = LogFactory.getLog(LoginListener.class);
  public static String sendSMS(String phoneNo,String genSequence,String mport){

          if(phoneNo.startsWith("0")){
                    phoneNo="+254"+phoneNo.substring(1);
          }else{
           return "pn";
          }
        try{

                         Service srv;
                                      OutboundMessage msg;
                                      srv = new Service();
                                      SerialModemGateway gateway = new SerialModemGateway("NetEx", mport,
19200, "Motorola", "L9");
                                      gateway.setInbound(true);
                                      gateway.setOutbound(true);
                                      //gateway.setSimPin("6063");
                                      srv.addGateway(gateway);
                                      srv.startService();
                                      String log="Modem information\n"+"manuf"+gateway.getManufacturer()+"\n
gateway Model"+gateway.getModel();
                                      msg = new OutboundMessage(phoneNo, "Login code: "+genSequence);
                                      srv.sendMessage(msg);
                                      SendScript.log.debug("dfdfdf"+phoneNo);
```

```
                                        Xlog.LogError1(log);

                                        System.out.println(msg);

                                        srv.stopService();

            }

            catch(Exception e){

               e.printStackTrace();

               Xlog.LogError(e);

            }

            return null;

    }


}


import java.util.Calendar;
import java.util.Locale;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import org.apache.commons.lang.StringUtils;
import org.springframework.validation.BindException;
import org.springframework.web.servlet.ModelAndView;

import com.nx.ebank.Constants;
import com.nx.ebank.Xlog;
import com.nx.ebank.model.EbPayInstruction;
import com.nx.ebank.model.EbPayInstructionType;
import com.nx.ebank.model.User;
import com.nx.ebank.model.XCurrency;
import com.nx.ebank.service.AuditTrailManager;
import com.nx.ebank.service.EbankingManager;
import com.nx.ebank.service.UserManager;
import com.nx.ebank.service.ViewManager;

public class EBPaymentsInstructionFormController extends BaseFormController {
	private ViewManager viewManager=null;
	private EbankingManager ebankingManager=null;
	private UserManager userManager =null;
	private AuditTrailManager auditTrailManager=null;

	public EBPaymentsInstructionFormController(){
		setCommandClass(EbPayInstruction.class);
		setCommandName("instruction");
	}
	protected Object formBackingObject(HttpServletRequest request)
	throws Exception {
		EbPayInstruction instruction = new EbPayInstruction();
		String id = request.getParameter("id");
```

```
                User ruser = getUserManager().getUserByUsername(request.getRemoteUser());
                try{
                        request.setAttribute("zanguList", viewManager.getAccounts(ruser.getId().toString()));
                        request.setAttribute("instList", ebankingManager.getEbPayInstructionTypeAll());
                        request.setAttribute("currencyList", ebankingManager.getXCurrencyAll());
                        if(!StringUtils.isBlank(id)){
                                instruction = ebankingManager.getEbPayInstructionById(id);
                                if(!(instruction.getAccountFrom()==null)){
                                        instruction.setAccfrom(instruction.getAccountFrom());
                                }
                                if(!(instruction.getAccountTo()==null)){
                                        instruction.setAccto(instruction.getAccountTo());
                                }
                                if(!(instruction.getCurrency()==null)){
                                        instruction.setCurr(instruction.getCurrency().toString());
                                }
                                if(!(instruction.getPayinstid()==null)){
                                        instruction.setOmbi(instruction.getPayinstid().toString());
                                }
                        }
                }catch (Exception e) {
                        Xlog.LogError(e);
                        e.printStackTrace();
                }
                return instruction;
        }
        public ModelAndView onSubmit(HttpServletRequest request,HttpServletResponse response, Object
command, BindException errors)throws Exception{
                User ruser = getUserManager().getUserByUsername(request.getRemoteUser());
                EbPayInstruction instruction = (EbPayInstruction) command;
                boolean isNew = (instruction.getPayid()==null);
                String success = getSuccessView();
                Locale locale = request.getLocale();
                String action ="";
                String msg = "";
                try{
                        XCurrency currency = ebankingManager.getXCurrencyById(instruction.getCurr());
                        EbPayInstructionType type =
ebankingManager.getEbPayInstructionTypeByCode(instruction.getOmbi());
                        if(isNew){
                                action ="ADD";
                                msg = "Added Payment Instruction for "+type.getInstruction();
                                instruction.setApproved(false);
                                instruction.setDeleted(false);
                                instruction.setUserid(ruser);
                                instruction.setOsysdate(Calendar.getInstance());
                        }else{
                                action ="EDIT";
                                msg = "Edited Payment Instruction for "+type.getInstruction();
                                EbPayInstruction payInstruction =
ebankingManager.getEbPayInstructionById(instruction.getPayid().toString());
                                instruction.setApproved(false);
                                instruction.setDeleted(false);
                                instruction.setUserid(payInstruction.getUserid());
                                instruction.setOsysdate(payInstruction.getOsysdate());
                                instruction.setEditdate(Calendar.getInstance());
                                instruction.setEditedby(ruser.getId());
                        }
                                instruction.setAccountFrom(instruction.getAccfrom());
                                instruction.setAccountTo(instruction.getAccto());
                                instruction.setAmount(instruction.getAmount());
                                instruction.setBillNumber(instruction.getBillNumber().trim());
                                instruction.setCurrency(currency);
                                instruction.setDescription(instruction.getDescription().trim());
                                instruction.setHashfield("hashed");
                                instruction.setPaydate(instruction.getPaydate());
                                instruction.setPayinstid(type);
                                instruction.setRefNumber(instruction.getRefNumber().trim());
                                instruction.setSessionid(request.getSession().getId());
                                ebankingManager.saveEbPayInstruction(instruction);
                                String key = (isNew) ? "rec.added" : "rec.updated";
                                saveMessage(request, getText(key, msg,locale));
                                getAuditTrailManager().saveAuditTrail(action, msg, request.getRemoteUser(),
Constants.FUND_ROLE, request.getRemoteAddr());
                                if (!isNew) {
                                        success = "redirect:/fund.html?action=instruction";
                                }
                }catch (Exception e) {
                        Xlog.LogError(e);
                        e.printStackTrace();
                }
                return new ModelAndView(success);
        }
        public ViewManager getViewManager() {
                return viewManager;
        }
        public void setViewManager(ViewManager viewManager) {
                this.viewManager = viewManager;
        }
        public EbankingManager getEbankingManager() {
                return ebankingManager;
        }
        public void setEbankingManager(EbankingManager ebankingManager) {
                this.ebankingManager = ebankingManager;
        }
        public UserManager getUserManager() {
                return userManager;
        }
        public void setUserManager(UserManager userManager) {
                this.userManager = userManager;
        }
        public AuditTrailManager getAuditTrailManager() {
                return auditTrailManager;
        }
        public void setAuditTrailManager(AuditTrailManager auditTrailManager) {
                this.auditTrailManager = auditTrailManager;
        }
}
```