2018

A Secure electronic document management system using public key encryption: a case of Strathmore University.

Stephen M. Bichage
Faculty of Information Technology (FIT)
Strathmore University

A Secure electronic document management system using public key encryption: a case of Strathmore University

Momanyi, Stephen Bichage

Submitted in Partial Fulfillment of the Requirements for the Degree of Master of Science in Information Technology at Strathmore University

Faculty of Information Technology
Strathmore University
Nairobi, Kenya

June, 2018

Declaration

I declare that this work has not been previously submitted and approved for the award of a degree by this or any other University. To the best of my knowledge and belief, the thesis contains no material previously published or written by another person except where due reference is made in the thesis itself.

Signature: …….. Date: ……

Momanyi, Stephen Bichage

08322

Approval

The research thesis of Momanyi, Stephen Bichage was reviewed and approved by:

Supervisor,

Signature: …….. Date: ……

Dr. Joseph Orero

Senior Lecturer, Faculty of Information Technology,

Strathmore University

Dr. Joseph Orero

Dean, Faculty of Information Technology

Strathmore University

Professor Ruth Kiraka

Dean, School of Graduate Studies

Strathmore University

Table of Contents

Contents

Abstract

The need to safeguard and ensure the authenticity of academic records is paramount for any reputable educational institution. As institutions are moving into the digital era, more and more documents are being stored in digital form. This comes with its own challenges and is compounded by the fact that computer files can be modified without leaving any trace, and one cannot tell the difference between the copy and the original. Hence, the need to employ security measures, such as cryptographic techniques to ensure the integrity of digital documents.

Paper documents have been used as backups of academic information store in academic management systems, these are difficult to manage and entail a great deal of manpower to maintain. Besides, they also require large storage facilities. The volumes of the paper documents have increased over the years and this has necessitated the search for a better way of managing them.

A lot of effort has gone into research of verification of academic certificates, while little has been done ensuring that accurate grade information maintained in academic management systems. Strict procedures must be put in place to ensure there is not tampering of the grade information in the academic management systems. In this regard, a secure document management system using public key encryption can provide a solution for storage and security of digital documents. In this research, we explored how these systems can be used to protect the integrity of digital documents.

This research proposes the use of a secure document management system using public key encryption that will facilitate the digital signing and storage of the digital documents. To implement the model, we developed a system that manages user rights and enables authorized users to use a public certificate to embed a digital signature on digital documents.

The system was implemented using a document management system with an additional module for digital signing. Private keys and digital certificates were generated for each of the authorized user and uploaded to the system using each user's credentials. To apply their digital signatures, the users' must input a password to complete the process, thereby providing an addition layer of security.

The system was tested for functionality, usability and effectiveness in managing digital documents and from the test results it shows that the system can be relied upon to securely and efficiently manage digital documents.

Keywords: Digital Signatures, Cryptography, Public Key Encryption, Asymmetric Encryption Electronic Document Management, Enterprise Content Management

Acknowledgment

I would like to thank wholeheartedly my supervisor Dr Joseph Orero who guided and provided the encouragement to pursue the thesis. To work colleagues who also provide the moral support throughout the entire process.

List of Figures

List of Tables

Abbreviations/Acronyms

AD	Microsoft Active Directory
AES	Advanced Encryption Standard
AMP	Alfresco Module Package
AMS	Academic Management System
CA	Certificate Authority
CSR	Certificate Signing Request
CUE	Commission for University Education
DSA	Digital Signature Algorithm
ECM	Enterprise Content Management
EDM	Electronic Data Management
FOSS	Free and Open-Source Software
ITU-T	International Telecommunications Union's Standardization Sector
LDAP	Lightweight Directory Access Protocol
PAdES	PDF Advanced Electronic Signatures
PDF	Portable Document Format
PKCS	Public Key Cryptography Standards
PKI	Public Key Infrastructure
MMT	Module Management Tool
RSA	Rivest-Shamair-Adleman
SHA	Secure Hash Algorithms
XAdES	XML Advanced Electronic Signatures

Definition of Terms

Alfresco	An OpenSource Document Management System
Cryptanalysis	It refers to analyzing and breaking the keys used for encryption and decryption (Rao & Nayak, 2014).
Brute-force attack	The trial of every possible key on a piece of ciphertext until an intelligible translation into plaintext is obtained. (Stallings & Brown, 2015).
Consolidated Marksheets	Final approved list of students and the marks they scored in each academic unit obtained from the AMS.
X509	A standard that defines the format of public key certificates

Chapter 1: Introduction

1.1 Background to the Study

Student grade information is an asset whose confidentiality is considered to be highly important by students (Stalling & Brown, 2013). Security of applications storing such information is considered paramount; therefore, the security measures employed must guarantee the privacy of the information and only to be made available to authorized person. The authenticity of the information also needs to be assured, verified and protected against any form of attack or deliberate modification.

In the effort to assure confidentiality and authenticity academic records, many checks and balances have been put in place in the process of managing these records, this include assigning different roles to administrators managing these records. In the case of Strathmore University, periodic reports are generated from the academic management system and archived in paper-based format, for verification of system integrity. This can be used to verify whether any data has changed in the live system. This process aims to meet the requirements of the Kenyan Universities' regulatory body.

The examination process must have high standards of controlled to maintain accuracy and meet the requirements of the Commission for University Education (CUE). The Commission conducts regular audits of universities in Kenya to ensure high standards are maintained in the whole academic process (CUE, 2017).

Traditional paper-based document business process has many weaknesses; as they can be easily lost or forged. Therefore, the need for a secure electronic document system for eliminating weaknesses of the paper document system. An author of a document must undeniably link to the document, and any change in the contents of the document must be immediately detectable. In a secure electronic document system, all documents are stored in a central location and the risk of loss is minimized (Na & Lee, 2008).

Simply implementing an electronic document system does not eliminate all the weaknesses of paper document, in fact digital documents can also be tempered with. Digital documents can be changed and copied without leaving any evidence, and copies cannot be differentiated from the original (Wilson, 1999). Therefore, the need to implement cryptographic security measures to

protect digital documents. A Digital signature is one such security measure that will provide assurance of the integrity of the digitally stored records. On the other hand, an electronic document management system will provide the functionality of controlling who views the documents through authentication and it will eliminate the needs for storing paper documents in cabinets.

1.2 Problem Statement

Institutions have been using paper documents to manage business process for a long time. The inherent challenges of paper have been compounded by the increased volumes of paper documents generated in the day to day operations of organizations.

This situation is not sustainable because it is time-consuming and paper documents are not easy to secure against unauthorized access, it also requires a lot of physical storage space. The indexing and filing of these documents may be neglected, and this makes it difficult to find and retrieve paper documents. The employees who are affected by this exercise can be released to do more constructive tasks, if the documents were to be managed in digital form.

The need to change from paper document management to digital management of documents poses a numbers of security risks. These risks include; authenticity of digital documents and authentication of users. In this research, a model for a secure electronic document management system using public key encryption was proposed.

1.3 Research Objectives

i. To investigate the challenges of paper records management.
ii. To investigate current techniques of securing digital documents.
iii. To propose a digital signing model for digital document management.
iv. To validate the digital signing model.

1.4 Research Questions

i. What are the challenges of paper document management?
ii. How what are the current techniques of securing digital documents?
iii. How will the proposed digital signing model be implemented?
iv. How will the digital signing model be tested?

1.5 Justification

Sensitive documents, such as grade information need to be safeguard against security risks. These risks may be in the form of loss of confidentiality or alteration of the records, thereby losing its authenticity (Wilson, 1999).

The handwritten signatures take a lot of time and effort on the part of the administrators who are tasked to manage or approve paper documents. The paper documents also require a huge space for storage. This research will contribute to making the digital storage of documents more secure and eliminate the need for a large storage facility. The importance of securing academic records cannot be overstated and all possible security measures need to be put in place. Therefore, the use of a secure document management system using public key encryption will assure the authenticity of digital material. This can be further extended to other critical business documents that need approval, making the process more efficient and highly secure.

1.6 Scope and Limitations

This research will focus on security of digital documents in higher learning institutions and in this case, Strathmore University. The system will also have minimum functionality to demonstration the applicability of the proposed model.

Chapter 2: Literature Review

2.1 Introduction

The challenges of managing paper-based documents are many and have repercussions in terms of efficiency, costs and security. The process of changing to an electronic document system comes with new needs, such as security management. This move is occasioned by the challenges faced by paper documents, the need to be more efficient and to save on costs while not compromising on security. This chapter will introduce several concepts that will be used to overcome security challenges posed by the adoption of electronic document storage and to review challenges of paper documents.

2.2 Challenges of Paper-Based Documents

The Commission on Systemic Interoperability (2005) identified some challenges of paper documents. Paper documents are difficult to share, the physical copy has to be transported to the party that needs access to it and this cannot be done remotely via a computer system. It is difficult to maintain confidentiality with paper documents, since they can easily fall to the wrong hands and once viewed, there is no way of tracing who view it. There is no guarantee of backup in case of a disaster.

2.2.1 Missing Marks

Nation Media (2012) relates how difficult it is to retrieve academic records, "the search for missing marks in institutions of higher learning can take several years, and those who are not lucky enough to trace them are forced to drop out of university altogether". In another paragraph the newspaper states, "… thousands of students across the country's public and private universities will be forced to put off their graduation until their marks are found or they re-sit the units". "About 2,000 students are not graduating this year because of missing marks. Some of them have unsuccessfully camped at the university for weeks trying to trace their marks". There is a high possibility of paper documents getting lost, this accounts for the missing marks in these universities.

2.3 Digital Documents

On the other hand, digital documents can be shared instantaneously with several users who are authorized. It is easier to limit unauthorized access and the trace who have viewed digital documents. It is easier and faster to access digital documents in a more secure manner and there is less likelihood of losing information. The use of digital record in institution of higher learning can

spare the students the agony of tracing their marks and minimizes the chance of the marks being lost.

2.4 Digital Certificate Verification Systems

There are number of system that are in the market providing verification services of academic certificates. They rely on the institution of higher learning to provide accurate information of students' academic records. There is need to ensure that the source of the information is accurate from the side of the academic institutions.

2.4.1 iCertify

This is a web-based platform in India that provides: degree certificate verification, marks card verification, year of passing verification, college enrollment verification, university verification and courses verification. It provides this services to universities, corporates, students and recruitment agencies (iCertify, 2017). The system relies on the information provided by the various institution, and their accuracy depends on the procedures established within those entities. The emphasis of the system is verifying the end-product while the model proposed will tackle inherent problems of electronic documents at the source.

Figure 2-1 Certificate Verification (iCertify, 2017)

2.4.2 iHEDD

HEDD is UK Higher Education's official service for candidate verification and university authentication, it provides verification of academic credentials and authentication of universities/colleges through its web application called iHEDD. It provides its clients a platform to verify whether a candidate is a current or past student of a university or college, the award and grade they received, and their dates of attendance. It has securely fulfilled over 164,000 enquiries (HEDD, 2017).

Just like iCertify describe above, the focus of this system it the verification of the end-product and not the process of managing the students' academic records.

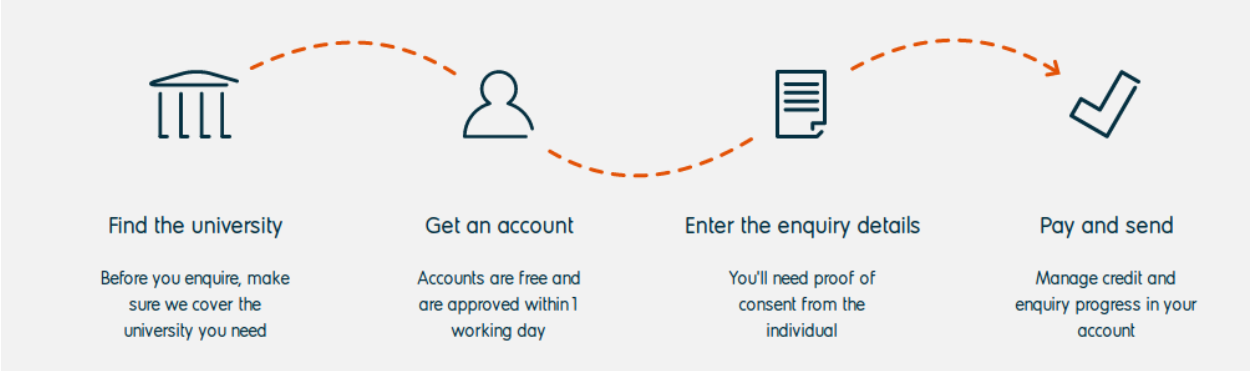

Figure 2-2 iHEDD Verification (HEDD, 2017)

2.4.3 CertVerify

ETX-NG (Electronic Transcripts and Documents Exchange in Nigeria) CertVerify is a web service that allows a requestor to verify the higher education, university education and professional certification qualifications claimed by a graduate or member (EXT-NG, 2017).

2.4.4 Digitary

"Digitary is a secure online service that allow organizations to issue secure digital records online, so that the subjects of those records can access and share them on the web with third parties. This eliminates the hassle associated with having to produce original paper documents". The PDF documents produced through this service contain cryptographic digital signatures to ensure their authenticity and integrity. (Digitary, 2017). This system goes a step forward in using digital signatures to protect the document from tampering.

2.5 Security of Information Systems

According to Stallings and Brown (2015), “Computer security means the protection afforded to an automated information system in order to attain the applicable objectives of preserving the integrity, availability, and confidentiality of information system resources (includes hardware, software, firmware, information/data, and telecommunications)”. Confidentiality means preserving authorized restrictions on information access and disclosure. Integrity refers to guarding against improper information modification or destruction, including ensuring information nonrepudiation and authenticity. A loss of integrity is the unauthorized modification or destruction of information. Authenticity is the property of being genuine and being able to be verified and trusted; confidence in the validity of a transmission, a message, or message originator.

2.6 Electronic Document Management (EDM)

“An automated system used to support the creation, use and maintenance of electronically created documents for the purposes of improving an organization’s workflow. These systems do not necessarily incorporate recordkeeping functionality and the documents may be of informational rather than evidential value” (Johnston & Bowen, 2005).

EDMs are not be suitable for the purposed model of tracking changes in digitals documents, an Enterprise Content Management system, described below will provided a more sophisticated platform for securely managing digital documents.

2.7 Enterprise Content Management (ECM)

Enterprise Content Management is the systematic collection and organization of information, it combines strategies, methods, and tools use to capture, manage, store, preserve, and deliver information supporting key organizational processes through its entire lifecycle. (AIIM, 2017).

ECM goes beyond the idea of a file as an object to be managed by addressing other technical and organizational challenges related to managing content in the context of its organizational production and use (Haug, 2012). ECMs include modules that facilitate security and audit trails of what happens in the system. The is crucial the detect and address any breaches of security within the system.

2.8 Cryptography

Rao and Nayak (2014) define cryptography as the process of converting simple plain text into secret text called ciphertext and converting ciphertext back to its original simple text.

Cryptography is mainly used to protect confidentiality of the data. It is also used for checking integrity and authentication processes as well. For example, in many governance processes, a signature is an essential part of the process for authentication and maintaining integrity. In computerized systems, where approval, and other governance is done via network or Internet, then we need a mechanism to authenticate the user's signature digitally (digital signatures) and provide a digital timestamp.

2.9 Cryptographic Algorithms

The mathematical function used to encrypt and decrypt data consists of keys: a word, number, or phrase. The cryptographic algorithm makes use of one or more of these keys to encrypt the data. The same plaintext can be encrypted using different keys to get different ciphertext. The strength of the encryption depends on the keys and cryptographic algorithm which makes use of these keys to encrypt (Rao & Nayak, 2014).

There are three types of crypto-algorithms (based on key)

i. Symmetric Key (Secret Key Cryptography): Uses a single key to encrypt and decrypt the messages.

ii. Asymmetric Key (Public Key Cryptography): Uses one key to encrypt and another key to decrypt the messages.

iii. Hash Functions: Uses a mathematical transformation that transforms the message into a fixed length data that is unique to the corresponding source. These transformations are carried out using hashing functions/algorithms and are not normally reversible or are one-way hashes, as shown in Figure 2-3.

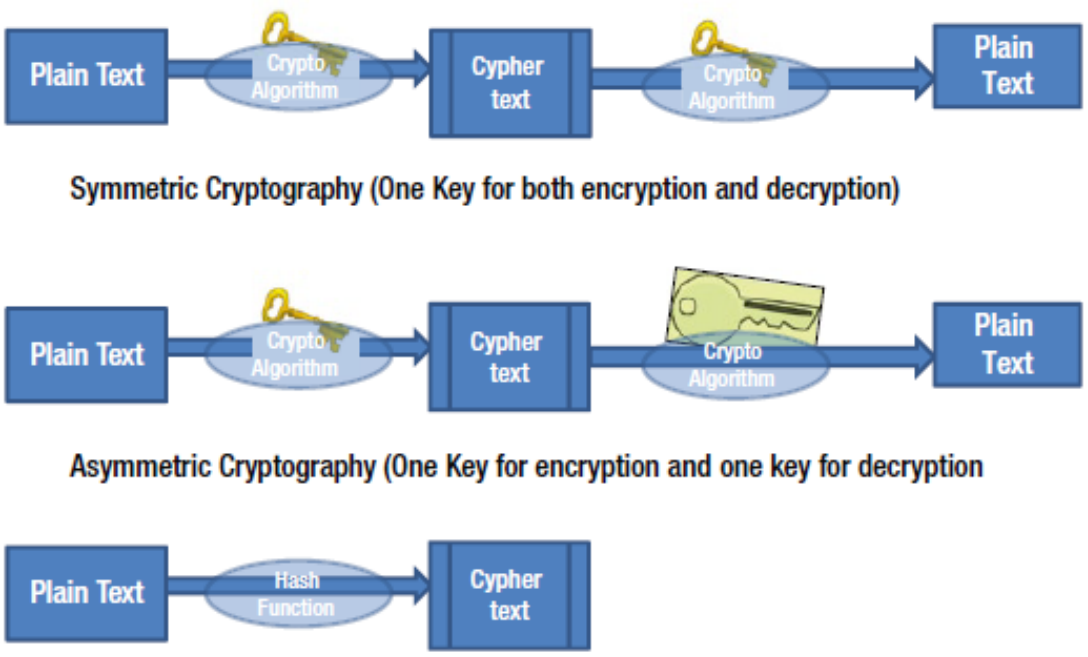

Figure 2-3 Cryptographic Algorithms (Rao & Nayak, 2014)

2.9.1 Symmetric Encryption

A single key is used to encrypt and decrypt the data. Both, the sender and receiver must share the same key to share confidential information. The two most important symmetric encryption algorithms are DES and AES. Stallings and Brown (2015) identified the two most common attacks on symmetric encryption as cryptanalysis and brute-force attack.

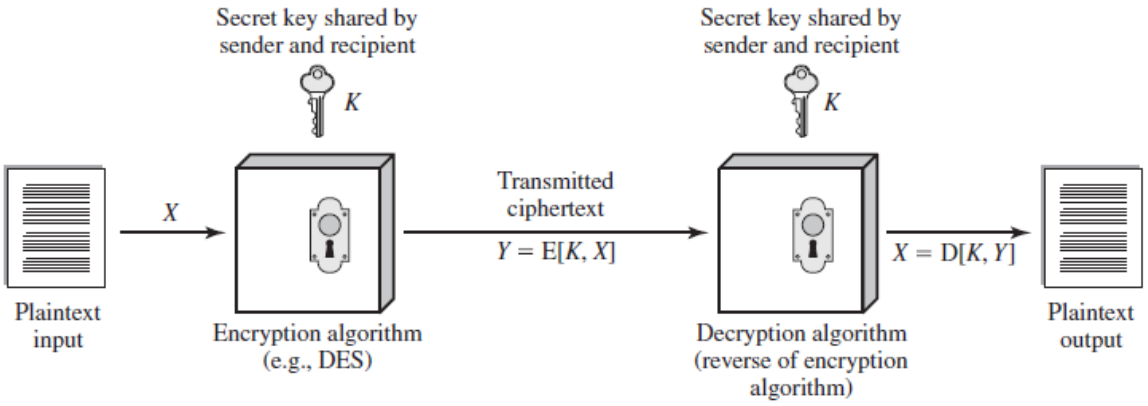

Figure 2-4 Symmetric Encryption (Stallings & Brown 2015)

2.9.2 Data Encryption Standard

DES takes a plaintext block of 64 bits and a key of 56 bits, to produce a ciphertext block of 64 bits. The shortfalls of DES are: the concerns about the algorithm itself and concerns about the use of a 56-bit key. Figure 2-5 shows the block and key sizes of the different encryption methods.

	DES	Triple DES	AES
Plaintext block size (bits)	64	64	128
Ciphertext block size (bits)	64	64	128
Key size (bits)	56	112 or 168	128, 192, or 256

DES = Data Encryption Standard
AES = Advanced Encryption Standard

Figure 2-5 Symmetric Encryption Algorithms (Stallings & Brown, 2015)

Figure 2-6 below shows how much time is required for a brute-force attack for various key sizes. Key with greater key sizes are effectively unbreakable using simply a brute-force approach.

Key size (bits)	Cipher	Number of Alternative Keys	Time Required at 10^9 decryptions/μs	Time Required at 10^{13} decryptions/μs
56	DES	$2^{56} \approx 7.2 \times 10^{16}$	$2^{55}\,\mu s = 1.125$ years	1 hour
128	AES	$2^{128} \approx 3.4 \times 10^{38}$	$2^{127}\,\mu s = 5.3 \times 10^{21}$ years	5.3×10^{17} years
168	Triple DES	$2^{168} \approx 3.7 \times 10^{50}$	$2^{167}\,\mu s = 5.8 \times 10^{33}$ years	5.8×10^{29} years
192	AES	$2^{192} \approx 6.3 \times 10^{57}$	$2^{191}\,\mu s = 9.8 \times 10^{40}$ years	9.8×10^{36} years
256	AES	$2^{256} \approx 1.2 \times 10^{77}$	$2^{255}\,\mu s = 1.8 \times 10^{60}$ years	1.8×10^{56} years

Figure 2-6 Average Time Required for Exhaustive Key Search (Stallings & Brown 2015)

2.9.3 Triple DES (3DES)

3DES is a variant of DES which involves repeating the basic DES algorithm three times, using either two or three unique keys, for a key size of 112 or 168 bits.

i. The 168-bit key length, which overcomes the vulnerability to brute-force attack of DES.
ii. Same encryption algorithm in 3DES as DES, which has stood the test of time with no effective cryptanalytic attack based on the algorithm has emerged. (Stallings & Brown, 2015).

This encryption technique will be used in the process of generating the digital signature because of its robustness against attack.

2.10 Public/Asymmetric key Cryptography

In public key cryptography two keys are used, one public and another private, whereby the public key is made public. The public key is used to encrypt messages intended for the owner of the key pair, because only he knows the corresponding private key necessary to decrypt the message (Hassler & Biely, 1999). The confidentiality of the message can be ensured by using the public key cryptography, and it is possible to establish authentication of the sender using digital signatures (Rao & Nayak, 2014).

2.10.1 Public key Algorithms

Public key algorithms are based on mathematical functions rather than on simple operations on bit patterns, such as are used in symmetric encryption algorithms. More important, public key cryptography is asymmetric, involving the use of two separate keys, in contrast to symmetric encryption, which uses only one key (Stallings & Brown, 2015).

The following are the algorithms used in public key encryption:

i. **RSA** One of the first public-key schemes was developed in 1977 by Ron Rivest, Adi Shamir, and Len Adleman.

ii. **Diffie-Hellman Key Agreement** The first published public-key algorithm appeared in the seminal paper by Diffie and Hellman that defined public key cryptography. The algorithm itself is limited to the exchange of the keys.

iii. **Digital Signature Standard** The DSS, published by the National Institute of Standards and Technology (NIST) makes use of SHA-1 and presents a new digital signature technique, the Digital Signature Algorithm (DSA). The DSS uses an algorithm that is designed to provide only the digital signature function. Unlike RSA, it cannot be used for encryption or key exchange.

iv. **Elliptic Curve Cryptography (ECC)** The principal attraction of ECC compared to RSA is that it appears to offer equal security for a far smaller bit size, thereby reducing processing overhead.

2.10.2 RSA Public-Key Encryption

The RSA scheme is the most widely accepted and implemented approach to public-key encryption. RSA is a block cipher in which the plaintext and ciphertext are integers between 0 and n - 1 for some n. Encryption and decryption are of the following form, for some plaintext block M and ciphertext block C:

$$C = M^e \bmod n$$

$$M = C^d \bmod n = (M^e)^d \bmod n = M^{ed} \bmod n$$

Both sender and receiver must know the values of n and e, and only the receiver knows the value of d. This is a public-key encryption algorithm with a public key of PU = {e, n} and a private key of PR = {d, n}. For this algorithm to be satisfactory for public-key encryption, the following requirements must be met:

a. It is possible to find values of e, d, n such that $M^{ed} \bmod n = M$ for all $M < n$.
b. It is relatively easy to calculate M^e and C^d for all values of $M < n$.
c. It is infeasible to determine d given e and n.

The first two requirements are easily met. The third requirement can be met for large values of e and n.

More should be said about the first requirement. We need to find a relationship of the form

$$M^{ed} \bmod n = M$$

The preceding relationship holds if e and d are multiplicative inverses modulo $\phi(n)$, where $\phi(n)$ is the Euler totient function. For p, q prime, $\phi(pq) = (p - 1)(q - 1)$. $\phi(n)$, referred to as the Euler totient of n, is the number of positive integers less than n and relatively prime to n. The relationship between e and d can be expressed as

$$ed \bmod \phi(n) = 1$$

This is equivalent to saying

$$ed \bmod \phi(n) = 1$$

$$d \bmod \phi(n) = e^{-1}$$

That is, e and d are multiplicative inverses mod ϕ (n). According to the rules of Modular arithmetic, this is true only if d (and therefore e) is relatively prime to ϕ (n). Equivalently, gcd(ϕ (n),d) = 1; that is, the greatest common divisor of ϕ (n) and d is 1. The summary of the RSA encryption is shown in Figure 2-7 below.

Key Generation	
Select p, q	p and q both prime, $p \neq q$
Calculate $n = p \times q$	
Calculate $\phi(n) = (p-1)(q-1)$	
Select integer e	$\gcd(\phi(n), e) = 1;\ 1 < e < \phi(n)$
Calculate d	$de \bmod \phi(n) = 1$
Public key	$KU = \{e, n\}$
Private key	$KR = \{d, n\}$

Encryption	
Plaintext:	$M < n$
Ciphertext:	$C = M^e \pmod{n}$

Decryption	
Ciphertext:	C
Plaintext:	$M = C^d \pmod{n}$

Figure 2-7 Summary of the RSA Algorithm (Stallings & Brown, 2015)

2.11 Public Key Infrastructure (PKI)

A Public Key Infrastructure (PKI) enables users to securely transact using public key cryptography. Key pairs are obtained from a third-party trusted authority called Certificate Authority (CA). The PKI provides an infrastructure to issue a "digital certificate" that identifies an individual or organization (Rao & Nayak, 2014).

A public key infrastructure consists of:

i. A Certificate Authority (CA) that issues and verifies digital certificates. A certificate includes the public key or information about public key
ii. A registration Authority (RA) which verifies the user's authenticity for CA before CA issues a digital certificate
iii. A secured storage place to hold the certificates and public keys
iv. A certificate management system
v. Hardware, software, policies, procedures, and people used to create, manage, and revoke digital certificates along with the distribution and storage of the digital certificates.

A certificate contains information referring to a public key, issued by a Certification Authority (CA). The information in the certificate should conform to the ITU (IETF) standard X.509 v3. Certificates conforming to that standard include information about the published identity of the owner of the corresponding public key, the key length, the algorithm used, associated hashing algorithm, dates of validity of the certificate, and the actions the key can be used for.

2.12 Certificate Authority (CA)

A CA is responsible for issuing certificates. CA issues the digital certificate based on the recommendation of RA. This digital certificate is signed by the CA using its own private key. The CA issues the certificate which contains the public key of the party who owns the certificate. Certificates have to be purchased from the CA. CA can issue a certificate only after it confirms all the credentials to prove your identity. Once identity is proved, it stamps the certificate to prevent modifications of the details contained in the certificate.

Registration Authority (RA) is a third-party verification agency for a Certificate Authority (CA), to perform the verification of the organization or individuals who have applied for the certificate. Final component of the PKI is the Certificate Management System (CMS) through which certificates are published, renewed, or revoked.

2.13 Digital Certificate

A digital certificate provides an electronic identity to conduct secure transactions by providing your identity (authentication). With a digital certificate, an organization or an individual can provide authentication for all the transactions with friends, business partners, and other online services. Digital certificate assures identity among all the parties involved in the transactions. The

most widely used format of a digital certificate is as defined by the ITU-T X.509 standards. Digital certificate uses public key cryptography to verify the integrity of the certificate itself.

A self-signed certificate is signed by the private key corresponding to the public key contained within it. For the purposes of demonstrating the working of the digital signing model a self-signed digital certificate will be used. This is because it is cheap and fast to generate unlike certificates obtained from a CA (Rao & Nayak, 2014).

2.14 Hash Function Cryptography

Hash functions, also called message digests, use a fixed length hash value to transform the data that makes it difficult for someone to decrypt or change the data without affecting the hash value, thus securing the data from intruders.

Hashing functions are one-way mathematical functions that are easy to compute but hard to reverse. A hash function H( ), applied on input (x), and returns a fixed string, h_s. Mathematically it is written as $h_s = H(x)$. A cryptographic hash function in general should have the following properties:

i. Flexible input length (x)
ii. H(x) should be relatively easy to compute
iii. H(x) is one-way function and cannot be reversible
iv. The output is of fixed length and does not depend on input length

Hashing is generally used in the following situations:

i. This method of cryptography is normally used in operating systems to protect passwords.
ii. Digital signatures and file integrity checkers to check the integrity of data.

Hashing functions are used to vouch for the integrity of the message by appending the message with the hash value. If the message is changed, the hash value when recomputed will not match the precomputed hash value. In order to avoid man-in-the middle attacks, it is ideal to send the hash value in a secure way to the intended party. Such secure transfer is possible using public key cryptography. Hashing is also used in some of the implementation of digital signatures which vouches for the integrity of the message sent (Rao & Nayak, 2014).

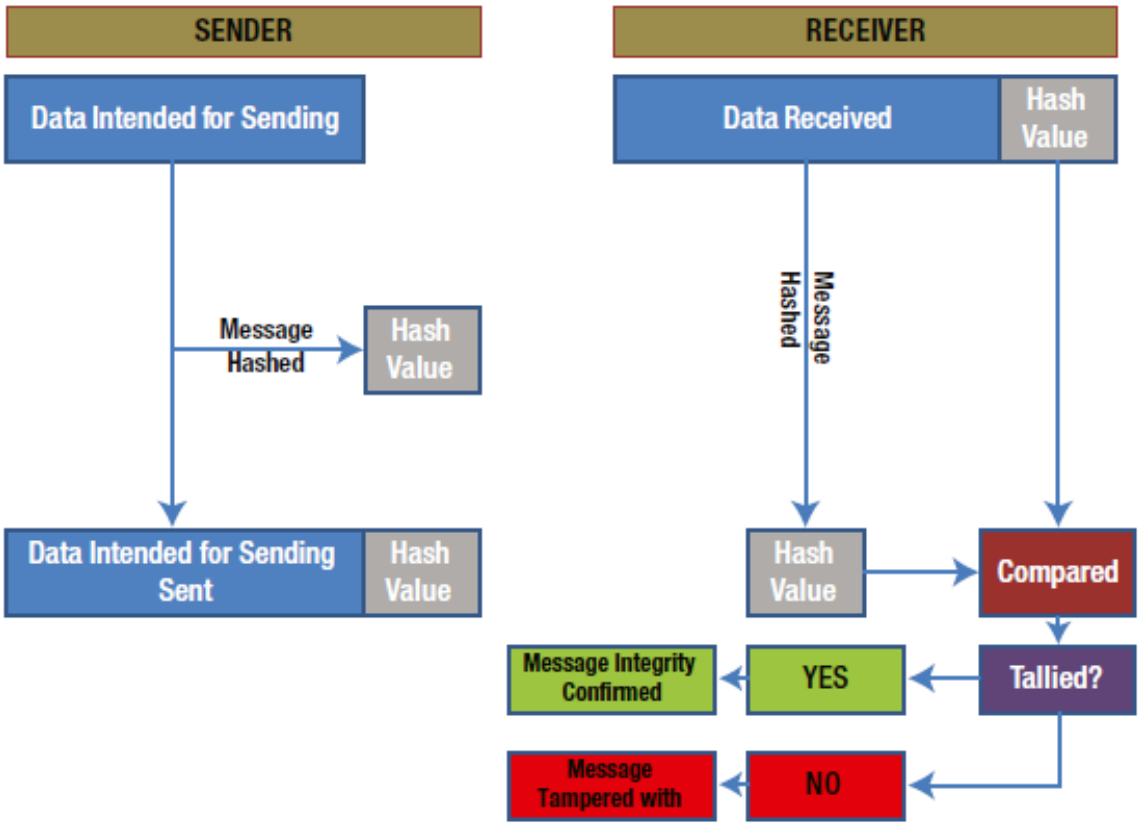

Figure 2-8 Message Integrity Check through Hashing (Rao & Nayak, 2014)

2.14.1 Popular Hashes

MD5 (Message Digest Function 5), SHA1 (Secure Hash Algorithm 1), SHA2 (Secure Hash Algorithm 2), and SHA3 (Secure Hash Algorithm 3) are the popular hashing functions /algorithms. MD5 outputs are of 128 bits and are popularly used for storing of the passwords as well as to ensure file integrity. MD5 is prone for collision. SHA algorithms again provide for one-way hash. SHA1 provides for 160-bit output. SHA-224, SHA-256, SHA-384, and SHA-512 are known as SHA-2. SHA3 is the most advanced hashing function which was announced by NIST in 2012. MAC (Message Authentication Code) is another popular hash function which is also known as a Keyed Hash Function.

2.15 OpenSSL

It is an open source project that provides a robust, commercial-grade, and full-featured toolkit for the Transport Layer Security (TLS) and Secure Sockets Layer (SSL) protocols. It is also

a general-purpose cryptography library. This application will be used to generate the digital signatures.

2.16 Document Management in Strathmore University

In the current system, the consolidate marksheets are exported from the Academic Management Systems (AMS) of Strathmore University for archival and referencing purposes. Once downloaded in PDF, the marksheets are printed and manually signed by the Deans of Faculties and Schools before being filed in cabinets. The exported documents are imprinted with a timestamp to indicate the precise date and time it was exported. This has generated a lot of paperwork which is tedious, time-consuming and requires a large storage facility.

After the marking and moderation of examinations by external examiners, the examination coordinators of each Faculty/School input the marks in the AMS. The marks are then approved by the Faculty Board before being released to the students, afterwards the students verify their marks. It is at this point, that queries are verified and necessary adjustments and made. The Academic Council then approves the final marks and the examination period is closed and no further change is possible. The consolidated marksheets of each examination period can be printed from this point onwards, as illustrated in figure 2-9 below:

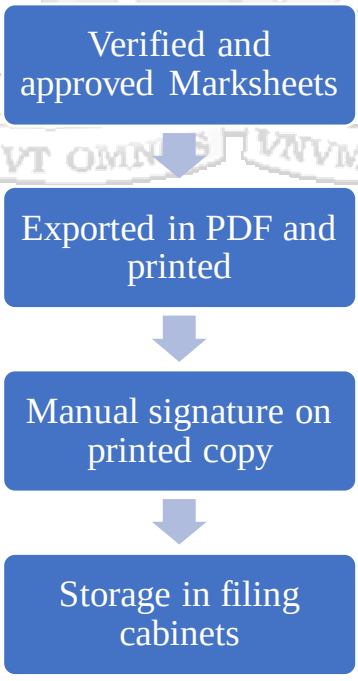

Figure 2-9 Export of Consolidated Marksheet from AMS

2.17 Digital Signatures

A digital signature is like a handwritten signature, but it is in the digital form for an electronic document. The document containing the digital signature is verified by the recipient using a hash function to check whether the message has been altered either intentionally or accidentally during the transmission. If the message is altered, the hash function returns a different result. Digital signature ensures authenticity and non-repudiation.

Here, usually the hash value is encrypted with the sender's private key. This provides for the authenticity. When the receiver decrypts the private key using the sender's public key, he gets the hash value. He can check this hash value with the hash value generated using the hash algorithm from the message received. (Rao & Nayak, 2014)

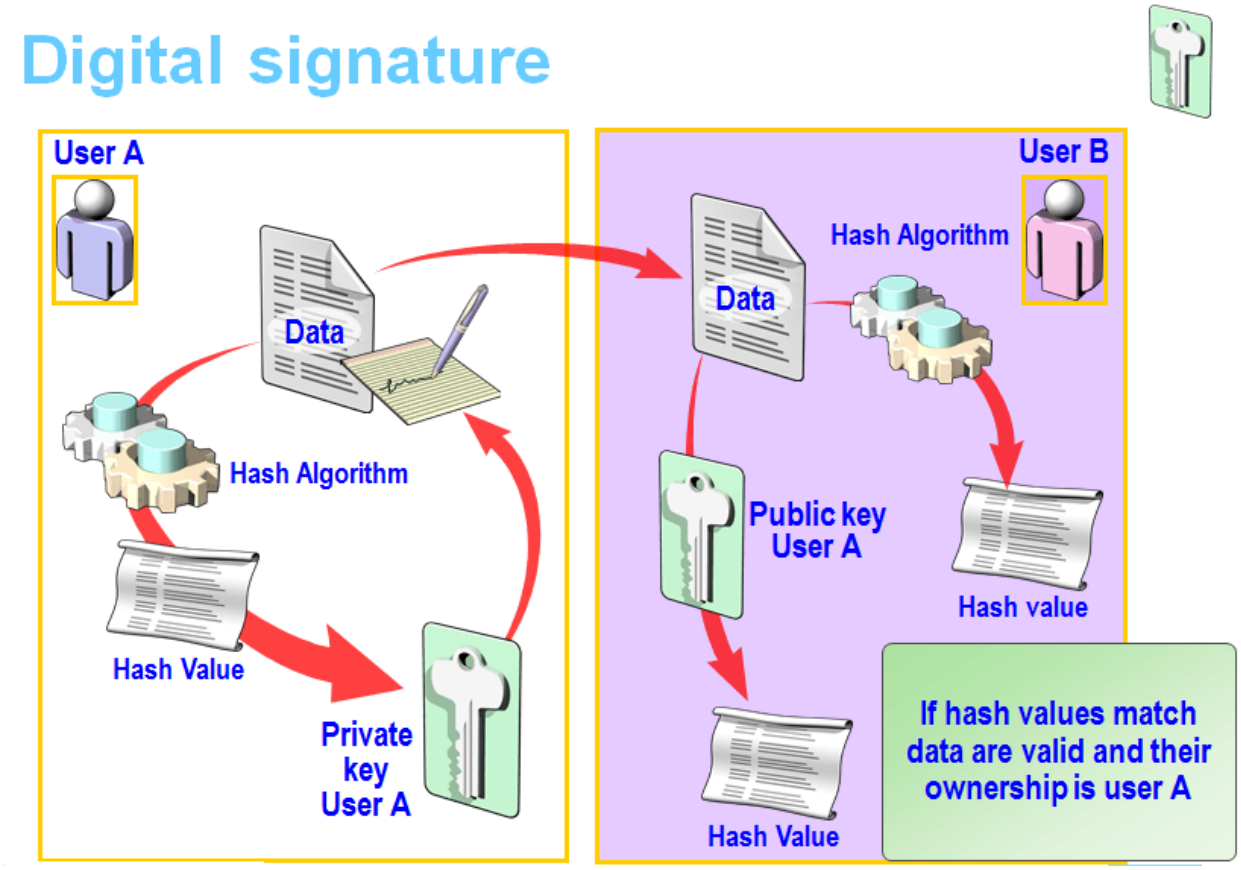

Figure 2-10 Digital Signature (Paganini, 2012).

2.18 Conceptual Framework

A conceptual framework brings together the different concepts and ideas discussed in this chapter and provides an overview of the entire process. Figure 2-11 below, explains the process of integrated digitally signed documents into a secure electronic document management system.

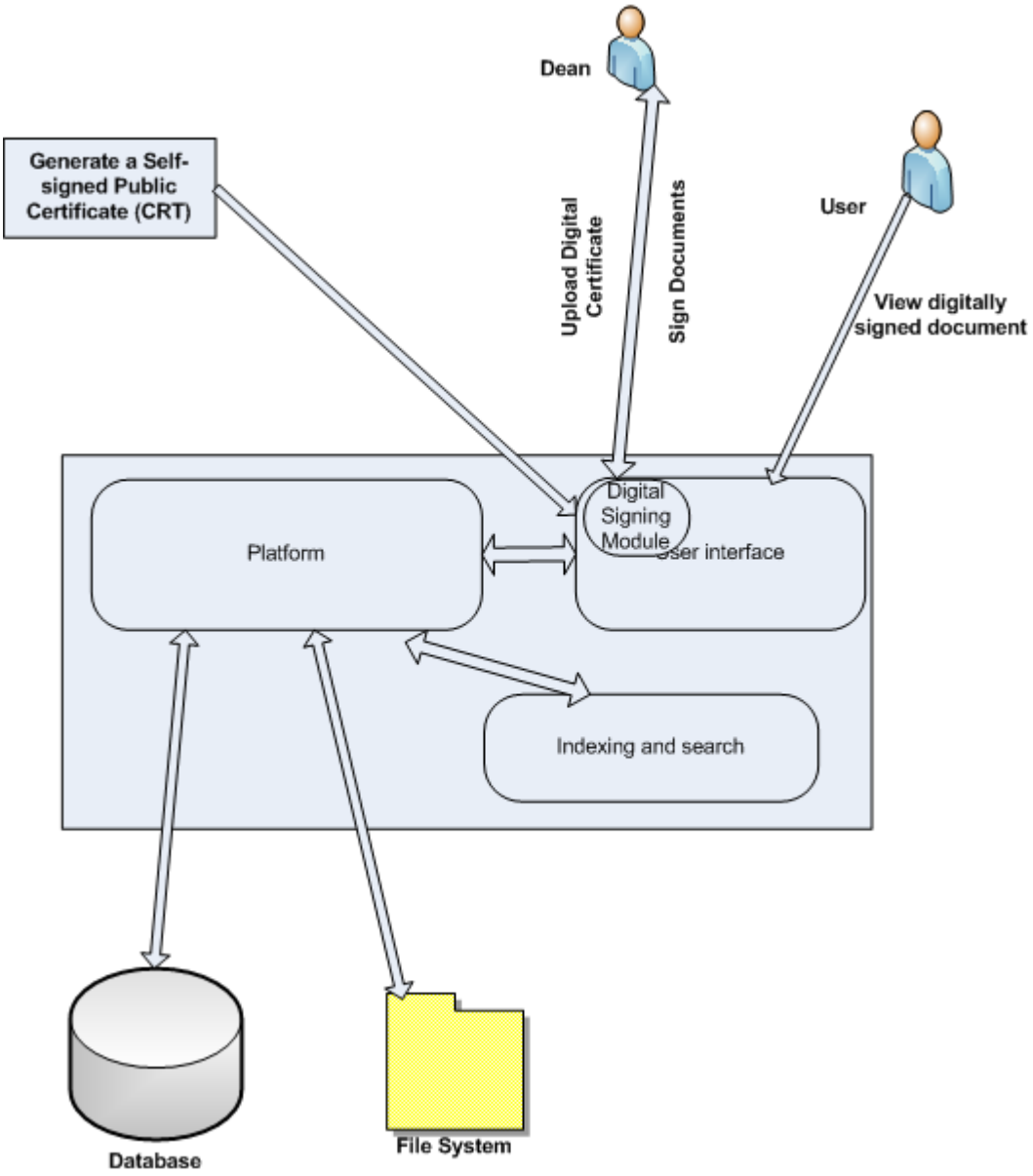

Figure 2-11 Conceptual Framework

Chapter 3: Research Methodology

3.1 Introduction

This chapter covers the research methodology used to study the challenges of paper documents and the procedure used to design and develop the proposed system. The methods used for the study and why they were selected are detailed in the sections below.

3.2 Research Design

Kothari (2004) defines research design as the conceptual structure within which research is conducted, it outlines the plan for collection, measurement and analysis of data. Since the research involved the study of the challenges face by administrator in managing paper documents and developing a solution to the problem applied research was used. The term "action research" was introduced by Kurt Lewin (1946) who characterizes it as research oriented towards bringing about change with the researchers actively involved in the situation or phenomenon being studied (as cited in Haug, 2012). The result of the process is to provide practical solutions for the existing problems in the organization. Applied research aims at finding a solution for an immediate problem facing society or an industry/business organization, while quantitative research is based on the measurement of quantity or amount (Kotari, 2004).

Questionnaires were used to evaluate the effectiveness and usability of the system and the results summarized in form of charts.

3.3 Rapid Application Development

According to WaveMaker (2017), "Rapid Application Development model relies on prototyping and rapid cycles of iterative development to speed up development and elicit early feedback from business users. After each iteration, developers can refine and validate the features with stakeholders". It also involves the re-use of software components instead of developing them from scratch. It also involves a rigidly paced schedule that defers design improvements to the next product version, and less formality in reviews and other team communication (Kikama, 2010).

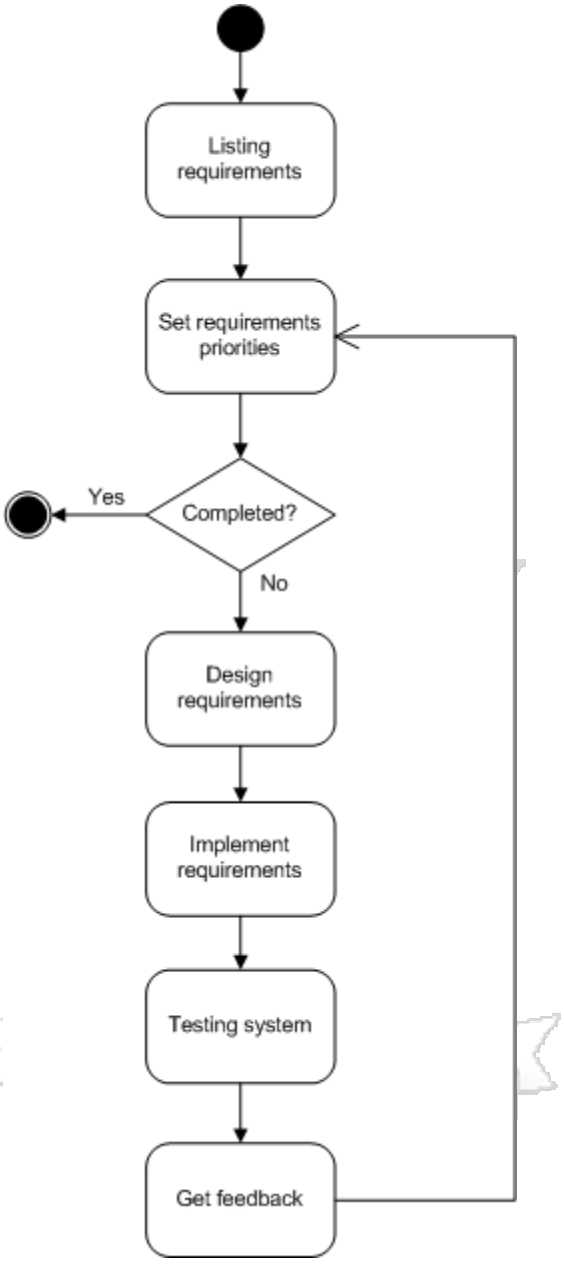

Figure 3-1 Rapid Application Development Methodology (Daud, Bakar & Rusli, 2010).

3.3.1 Requirement Planning Phase

This is the stage where the objectives, functionality, and scope are established. The project scope as well as a high-level list of initial requirements were developed.

The research used interviews to elicit the challenges of the users in managing the paper documents and their aspirations of the new system, the results were categorized and user requirements for the proposed solution were obtained. Interviews provide in-depth data which is not possible to get using a questionnaire and the interviewer can clarify questions thereby helping

the respondent give relevant responses (Mugenda & Mugenda, 1999). Five users were interviewed and their responses documented, appendix A, shows the interview guide used. Qualitative research includes designs, techniques and measures that do not produce discrete numerical data. (Mugenda & Mugenda, 1999).

3.3.1.1 Target Population

The population is 500 full-time staff of Strathmore University. Due to the limitation of time, purposive sampling was used. According to Kothari (2004) "the organizers of the inquiry purposively choose the particular units of the universe for constituting a sample on the basis that the small mass that they so select out of a huge one will be typically or representative of the whole"

3.3.1.2 Sample Size

The number of staff interviewed was 5, while 20 staff members were selected to participate in an online survey selected staff to get feedback on the functionality and usability of the system.

3.3.1.3 Data Analysis and Functional Model

The data collected by using an online questionnaire were analyzed and presented in graphical format. The data collected from the interviews was categorized and analyzed to come with the user requirements. Object oriented modelling using Use Cases was applied to identify the objects in the system and their relationships.

3.3.2 User Design Phase

The development team which is comprised of the main stakeholders meets to plan how the essential parts of the system should work. Several meetings were held to discuss the progress of the system and feedback channeled to the development process.

3.3.3 Construction Phase

In this step, the prototype is converted into a functional application. At this stage actual coding occurs, the application developers add the functionalities to the prototype. This is done in iterative cycles of development, testing, requirements refining, and development again, until the application is complete. Observation was also employed to check the usability of the prototype; 5 selected users were observed while undertaking different task within the system.

3.3.4 Cutover Phase:

The final user testing and training is done and decisions are made on the publication of the application system. This step involves a review of the constructed system by the stakeholders to determine whether it meets their expectations.

Questionnaires were used to get the feedback of the users on the functionality of the system. The questions are accompanied by a list of all possible alternatives from which the respondents select the answer that best describes their situation (Mugenda & Mugenda, 1999).

3.4 Ethical Considerations

The will be data collected with the consent of the participants and the information gathered will only be used for the purposes of the study.

Chapter 4: System Analysis and Design

4.1 Introduction

In this chapter, the process of getting user requirements and the design of the model of the model are covered. The requirements of the system were acquired through interviews with the users and the research carried out about a suitable solution.

The requirements analysis phase defines the business requirements for a new system, it answers the question, "What do the users need and want from a new system" (Whitten & Bentley, 2007). A successful system requires the system analyst to collect the requirements of the system by engaging the clients, also making use of the information available in the market to select the most suitable solution based on the budget and functionality of the system. This process requires constant communication with the users to determine their expectations of the system and resolve ambiguities.

4.2 Requirements Analysis

Interviews were conducted with the users to determine the currents operations in the process of handling documents. The challenges of the current systems were established, and this assisted in coming up with the requirements. The stakeholders also provided their input on what they would like the new system to operated.

4.3 Requirements Analysis

4.3.1 Challenges of Current System

The stakeholders interview identified the following challenges of the current system as shown in Table 4-1.

Challenge	Description
Time consuming/ Not efficient	i. The paper documents must be signed on each page ii. The documents have to be physically circulated to the signatories iii. It takes time to index and file the documents
Slow retrieval	It takes time to trace a document
Storage	The documents have to be filed in cabinets
Loss	Paper documents can be misplaced or damaged
Security	It is difficult to know who has the documents, if it was not signed out

Table 4-1 Challenges of Current System

4.4 Requirements of the Proposed System

The stakeholders interviewed provided feedback on the feature they wanted to see in the new system, these were captured and summarized in table format.

Requirements	Description
Efficient	i. Electronic signing ii. Automated indexing of documents
Fast	The search of documents should be fast
Digital storage	The documents to be stored in a server
Backups	The documents management system should be backed up daily
Security	User to login Audit trail of user activity on the system

Table 4-2 Requirements of the Proposed System

4.4.1 User Requirements

i. Manage user roles and rights in the system
ii. Upload documents into the system
iii. Authorized users to digitally sign documents
iv. A functionality for searching for documents
v. Authorized users should be able to view documents

4.4.2 Functional requirements

This describes activities and services a system must provide.

i. User registration – the system should maintain a register of users and their roles
ii. Login/logout – the user should login with their credentials to access the system and logout after completing their tasks.
iii. Document upload – the system should provide the option of uploading documents into the systems
iv. Generate digital certificates – the system should provide an interface for generating the certificate needed to sign digital documents.
v. Digital signature – the system should allow a user to upload a digital certificate with a scanned image of the users handwritten signature.
vi. Digitally signing – the system should provide the option of either signing a single or multiple documents.

4.4.3 Non-Functional Requirements

i. Ease of use – the system should be easy to use, since that users are busy and do not have time to learn complicated systems.

ii. Security – Controlled access to the system

iii. Availability – the system should be stable and available at all times.

iv. Performance – the system should fast response time

v. Reliability – The system should be robust and regular backup of the system done.

4.5 System Architecture

The architecture of the system comprises of four major components, the web client, network, application server and storage. The web client provides the users with an interface to interact with the system through the network. The system uses Apache Tomcat server to process the various request of the users, such as authentication, indexing and searching, document storage and retrieval and digital signing. The various configurations and documents paths are stored in a relational database such as MySQL. The file storage provides the space storing the documents.

The Java Virtual Machine is the cornerstone of the Java platform. It is the component of the technology responsible for its hardware - and operating system - independence, the small size of its compiled code, and its ability to protect users from malicious programs. The Java Virtual Machine is an abstract computing machine. Like a real computing machine, it has an instruction set and manipulates various memory areas at run time (Oracle, 2013).

Tomcat is an open-source product maintained by the Jakarta Project of the Apache Software Foundation. It contains the class libraries, documentation, and run-time support that you will need to create and test servlets (Schildt, 2002). The secure electronic document system uses the Tomcat server to provide the engine to run the core modules and provide a web interface for the users.

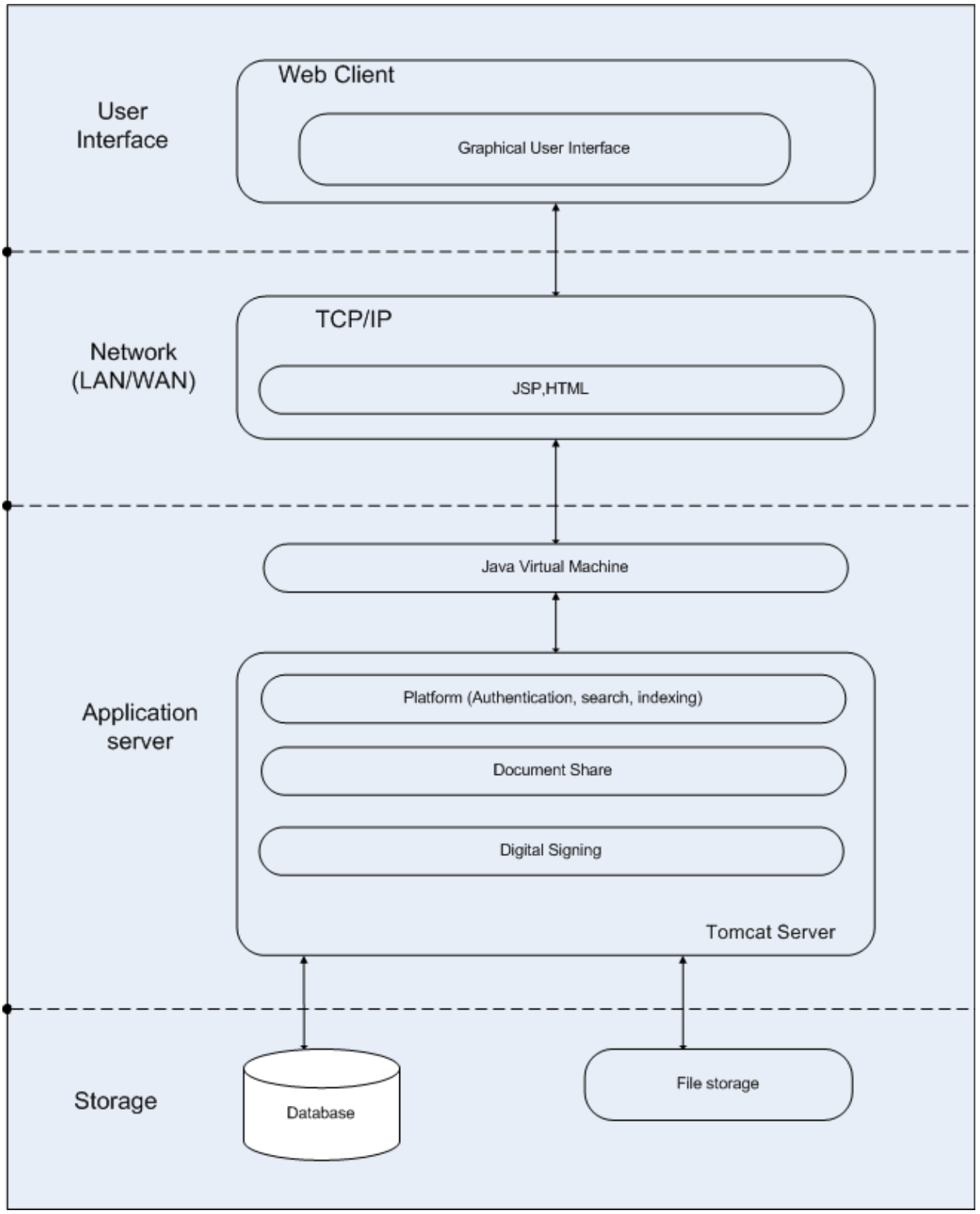

Figure 4-1 System Architecture

4.5.1 Alfresco Document Management

Alfresco is a Free and Open-Source Software (FOSS) Java-based enterprise application. The main component with all the core Enterprise Content Management functionality provides the repository where content is stored plus all the associated content services. Alfresco Share provides a web client interface for the repository. The search functionality is implemented on top of Apache Solr 4 and provides the indexing of all content, which enables powerful search functionality (Alfresco, 2017).

4.5.2 Validate System Architecture

Table 4-3 details the aspects of the system which have to conform to the requirements of Alfresco.

Validate that your environment	Operating Systems, Databases, Application Server
Validate and optimize the hardware	(I/O subsystems and CPU) settings
Validate the database	MySQL 5.6
Validate the operating system	64-bit OS recommended
Validate and tune the JVM	Fine tune JVM

Table 4-3 Validation of Alfresco Architecture

4.5.3 Selection of Digital Signature Module

There are several modules that were review, with the aim of selecting one that will fulfil the requirements of the model. The table below show the different features of the modules:

Addons/Classification Criteria	**X509 Certificate (Client)**	**X509 Certificate (Server)**	**Handwritten**	**External service**	**License**
SKYTIZENS Advanced Electronic Signature			X	X	Proprietary
Sinadura PDF Digital Signature Client for Alfresco ECM	X				GPL
Queres Digital Signature Alfresco Share	X				Proprietary
GroupDocs Signature			X		GPL
Digital Signature via Applet for Alfresco Share	X				Proprietary
Alfresco Share Electronic Signature (using Applet)	X				GPL
Digital signing		X			CC
Online PDF Signer	X		X		Proprietary
Add Digital Signatures to pdf files		X			Public domain
.be eID Sign (RedTree)	X		X	X	AGPL

DocuSign Connector for Alfresco		X	X	X	Proprietary
CounterSign for Alfresco		X	X		AGPL

Table 4-4 Digital Signature Comparisons (Borroy, 2015)

Table 4-3 shows the Alfresco Digital Signing Addons Classification Criteria.

i. **X509 Certificate (Client):** Electronic Signature operation is performed on client machine, by using a certificate physically owned by the user. It produces an attached (e. g. PAdES format) or detached (e. g. XAdES format using .xsig extension) electronic signature file.

ii. **X509 Certificate (Server):** Electronic Signature operation is performed on server machine (Alfresco), by using a certificate physically stored on this server. User allows the operation by identification mechanisms (e. g. user / password). It produces an attached or detached electronic signature file.

iii. **Handwritten:** Signature operation is performed by the user as an image of their physical signature. It can include biometric verification. It produces an image that can be attached to the document.

iv. **External Service:** Signature operation is performed by a third-party service outside of Alfresco.

v. **License:** Addon license declaration.

4.5.4 Reasons for the Selection of Digital Signing Module

This module was selected it is well documented and compatible with the version of Alfresco installed. The license type, that is, Common Creatives (CC) allows for anyone to modify the code to add new functionality and can be distributed it freely. The digital signature operation is carried out in the server, which provides more security unlike most which are store in a local machine.

4.6 Use Case Modelling

Use case analysis is the process of identifying and modelling business events, who initiated them, and how the system responds to them. Use cases identify and describe necessary system processes from the perspective of users. Each use case is initiated by users or external systems

called actors. An actor is anything that needs to interact with the system to exchange information (Whitten & Bradley, 2007). Figure 4-2 shows the users' interactions with the system.

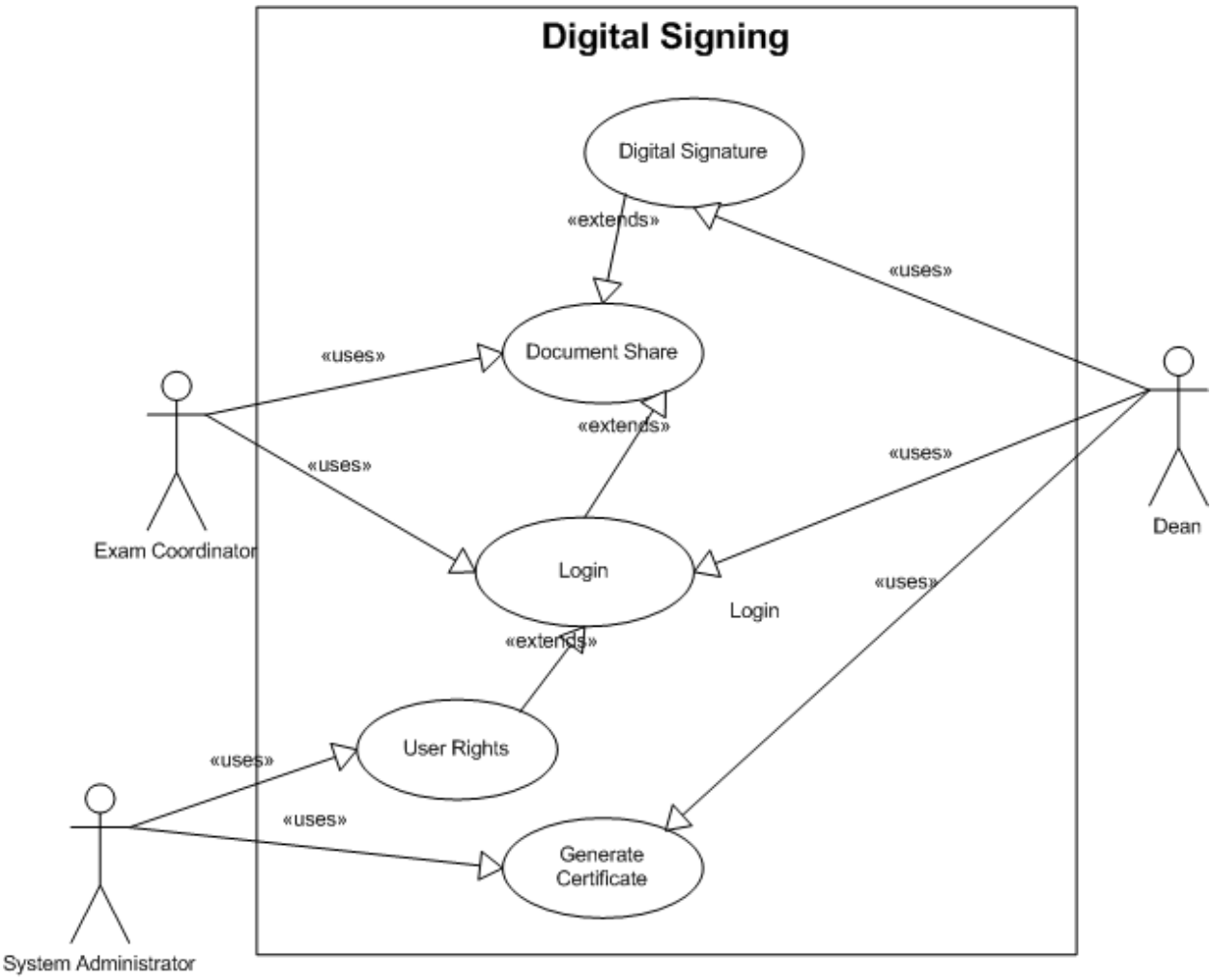

Figure 4-2 Use Case of the Model

Actor	Use Case
Exams Coordinator	Assigned user rights Uploads documents Views documents
Dean	Assigned user rights Upload signature View documents Digitally signs documents
System Administrator	Manages user rights Generate server certificates

Table 4-5 Main Use Cases

The examination coordinators are the actors tasked to upload the marksheets for the various Faculties and schools. They must be registered in the systems and be given access to the folders that corresponds to their Schools. The same applies to the Deans, who have the responsibility to digitally sign the marksheets once they are uploaded to system.

The system administrator assigns the different actors the rights to the system and assists the Deans in generating server certificates and uploading the digital signatures to the system. The main use cases are shown in Table 4-5.

ID	UC1
Title	User rights
Description	Registration of users
Actor(s)	System Administrator
Pre-conditions	System Administrator has logged in
Post-Conditions	Successfully assigned rights
Main Scenarios	i. Administrator searches for user and assigns roles to a site ii. Administrator removes user role from a site

Table 4-6 Manage User Rights Uses Case

The administrator logs into the system and assign users roles to folders within the system, the users can only access the folders that they have been assigned rights to. The administrator can also remove rights of users who are no longer entitled to view documents in the system. Table 4-6 shows that process of allocating rights.

ID	UC2
Title	Generate certificate
Description	Generate user certificate
Actor(s)	System Administrator
Pre-conditions	System Administrator and user have logged in
Post-Conditions	Successfully generated certificate
Main Scenarios	System administrator generates certificate Users upload digital certificates

Table 4-7 Certificate Generation Use Case

That administrator generates that public certificate to be used for the digital signing of the digital document for each of the authorized users and assists them in loading it into the system. The process is shown in table 4-7.

ID	UC3
Title	Login
Description	Login to the system
Actor(s)	All actors
Pre-conditions	System is online
Post-Conditions	Login successfully
Main Scenarios	View documents in assigned sites

Table 4-8 Login Use Case

The users login into the system with their credentials and can view the documents which they have rights to. The process is summarized in table 4-8.

ID	UC4
Title	Document Share
Description	Interface for managing documents
Actor(s)	Exams coordinator
Pre-conditions	Exams coordinate has logged in
Post-Conditions	Successfully upload documents
Main Scenarios	User creates folder User uploads documents

Table 4-9 Document Upload Use Case

The user logins in and uploads document to the folder in which he has rights to, he can create sub-folders within the system to organize the various document into categories.

ID	UC5
Title	Digital Signing
Description	Digital signing of documents
Actor(s)	Dean
Pre-conditions	Dean has logged in
Post-Conditions	Successfully signed documents
Main Scenarios	Dean selects document(s) for signing Dean signs selected documents

Table 4-10 Digital Signing Use Case

The dean logins into the system and selects that documents that needs their approval, once selected he proceeds to sign the document using his digital certificate, which was uploaded to his profile in an earlier process as shown in table 4-10.

4.7 Entity Relationship Diagram (ERD)

An ERD is a data model utilizing several notations to depict data in terms of the entities and relationships described by that data (Whitten & Bentley, 2007). Figure 4-4 shows the different entities and their relationships.

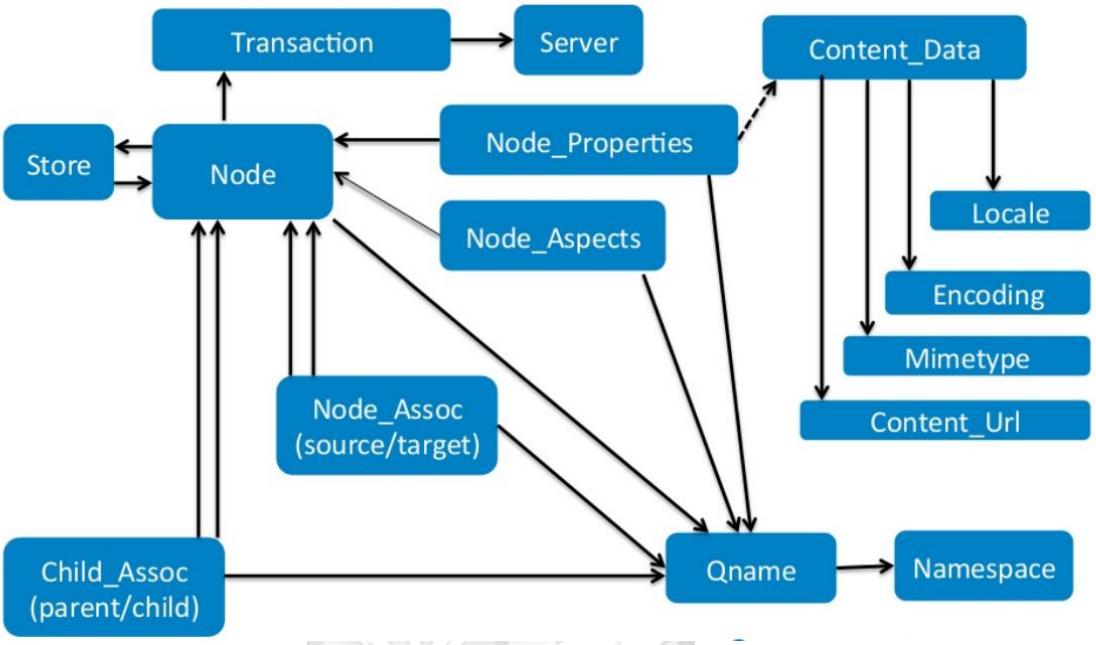

Figure 4-3 Partial Database Schema (Vonka, 2014)

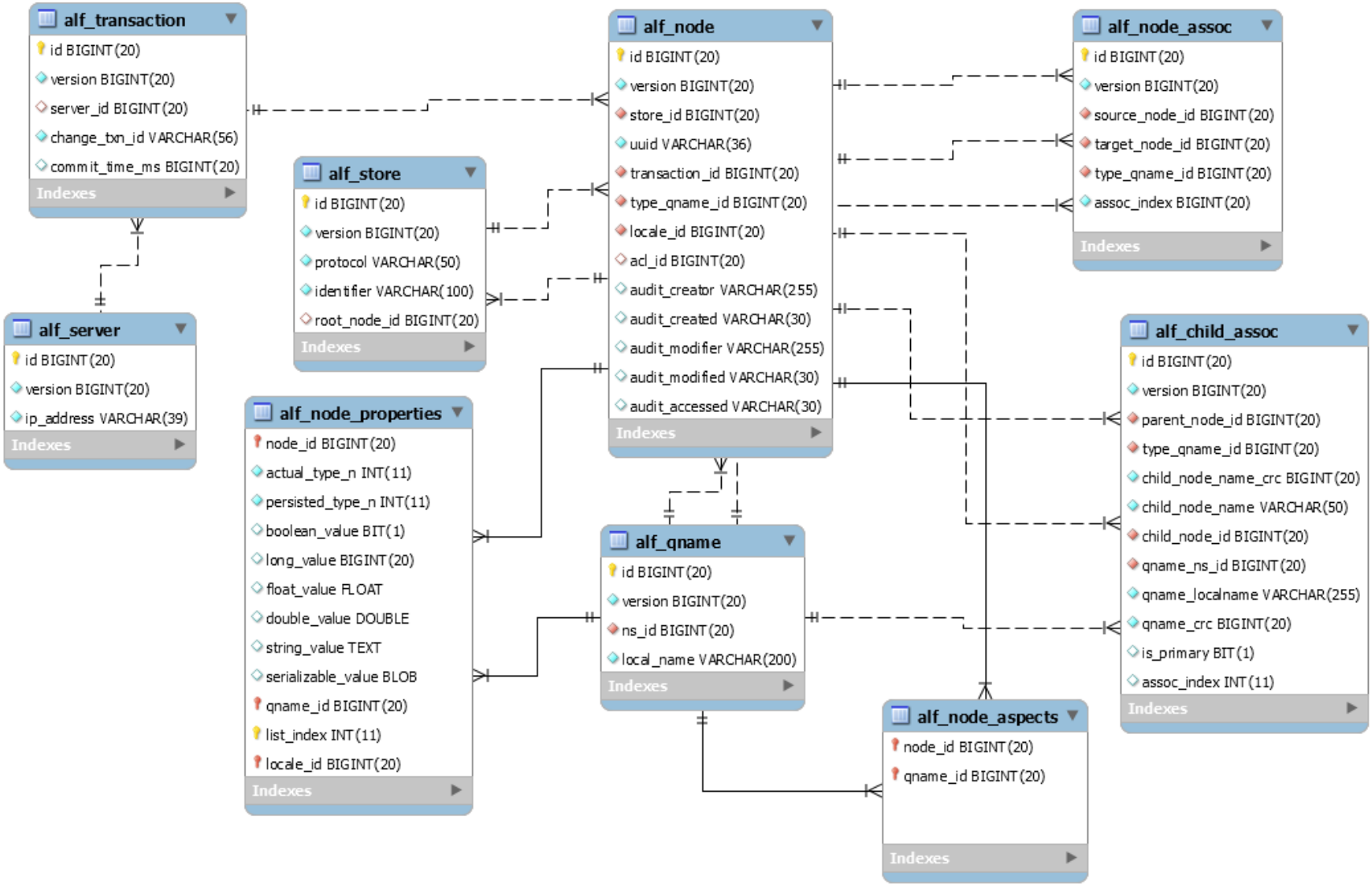

Figure 4-4 Partial Entity Relationship Diagram

Chapter 5: System Implementation and Testing

5.1 Introduction

The system was implemented using an OpenSource Enterprise Content Management (ECM) – Alfresco with an added module for digital signing. The digital certificates were generated using an OpenSource toolkit, OpenSSL.

5.1.1 Server Requirements

The requirements of the server are shown in Table 5-1, the high-end processor and 64-bit operating system are needed to ensure that the system runs fast. The hardisk will accommodate the operating system and the digital documents that will be stored in the system. The system requires an application server to run, and in this case, Apache 2.0 and above. Table 6-2 shows the requirements of Alfresco.

Component	**Recommendation**
Processor	Intel Quad Core 2MHz
Operation system	Linux 64-bit
Web Server	Apache 2.0+
Hardisk Capacity	80 GB

Table 5-1 Server Requirements

5.1.2 System Software Requirements

Component	**Recommendation**
Java Runtime Environment (JRE)	Alfresco supports both Java 7 and Java 8. The JAVA_HOME environment variable must be set to the location of the JRE.
Application server	Alfresco runs within an application server. Alfresco One 5.0 runs within Tomcat.
Database	MySQL 5.6.26
LibreOffice	Alfresco uses LibreOffice 4.2 for transforming documents from one format to another, for example, a text file to a PDF file.
ImageMagick	Alfresco uses ImageMagick to manipulate images for previewing.
GhostScript	Alfresco uses GhostScript in conjunction with ImageMagick to manipulate images for previewing.

Table 5-2 Alfresco Software Requirements

5.2 User Roles and Access

A user's role determines what they can and cannot do in in the system: Managers have full rights to all site content - what they have created themselves and what other site members have created. Collaborators have full rights to the site content that they own; they have rights to edit but

not delete content created by other site members. Contributors have full rights to the site content that they own; they cannot edit or delete content created by other site members. Consumers have view-only rights in a site: they cannot create their own content.

5.3 Generation of Digital Signature

The tool used to generate digital signature is OpenSSL which was installed in a Linux computer. The steps on generating the signature are detailed below:

5.3.1 Generate an RSA private key

```
openssl genrsa -out steve.key 2048
```

The command above generates a private key using RSA algorithm of the length 2048, as has been seen in the research the length of the key determines it strength, the longer the key the more difficult it is to crack it. Figure 5-1 shows the command to generate the private key and Figure 5-2 shows the actual private key generated.

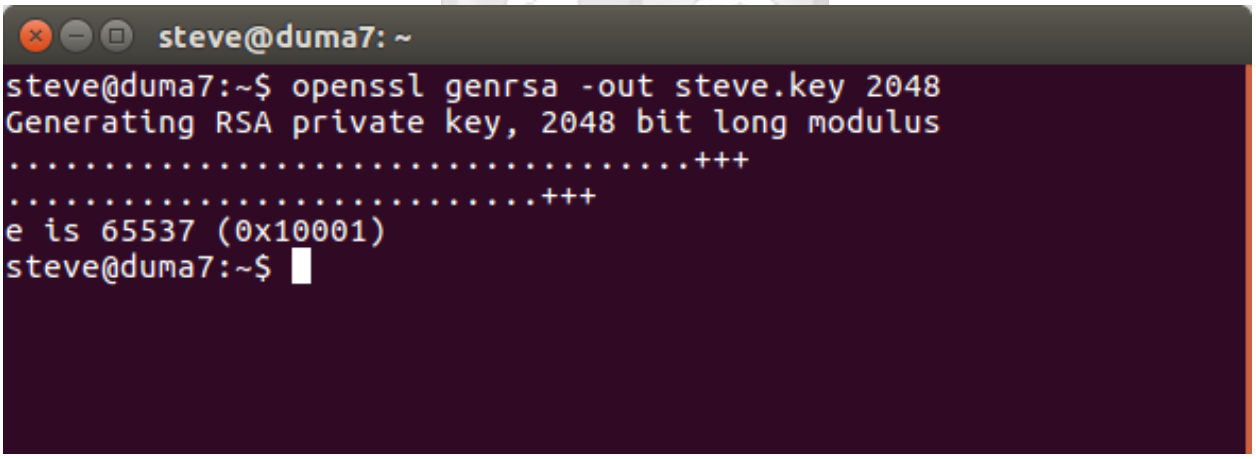

Figure 5-1 Generation of RSA Private Key

```
-----BEGIN RSA PRIVATE KEY-----
MIIEpAIBAAKCAQEAqO43TjfN0AoRin1KxzptpptU4SBu3VYYmiz0VY6cLkjQknFY
Wr2mURkJUvlXdHZzcPXn12VfyTVILfhtDuyxNr5F6VDZucUMXeq5oo3e5xSDtHVt
g+moHOskmHGRIx72tF4Ma6FAR4qRjETeiC1Hbad6qrMbgThv8QY2NTJWzMYBuPdu
NUAY9vqDQb3Fpn3XF/tiePiZu1gSpr0wFsxarpOucrUaJunGgAwZaS5L7E8rTSBH
rIaSHFXetoVtNZuoQA7Qhta9lKs5OyJANmw9pC9Mqd2+gG05B+9edU2d5z//Zin7
AAE8BOcknGTg+mVHXIgEaGsJUM5VQ6sX9uFBZQIDAQABAoIBAH0H41M3JlOGQDla
RpByhFm9ufLuSF/1qzvhbe4tIRd4uTbKQXTXzVNUznvlJH+EdCj1OGVKGL0ZfJTZ
vA5BzWARMDlhp+RBHcgdRMw1aPdrTZexGh96Pv04ZNuXoq516m7Q9SxsKuFIhDVn
e7UanjVgsf/JlN/tX5zq1hBFc80WxLEWFXw2kueG+6pXkr22giYQx/Bl9Is8Hs+W
RopiuIPohbDLbLlgYyUkqyY/WflWZH58k7KyRywDYhk1x2okT7fpTyevvtiTLwnu
l2ieEx89dqi5qYoPzV9zMAvR/SOnJY13haIxv7//7pNDlfIhk7FQXTzB3mMYD0it
Sckjr4ECgYEA0i918XWK8g1kXkXGRfwZuZZwvhfuBqoxsaTqJfbUgy4ngTWSmDcC
7Z8plHprqi88i9s9kBLYD26Ip8XIF51DA8WdUZwqEYu8L6nHEBObZkGF8xClp4MA
kBfnn26ZV36eQM/32D1+DU95fKYjpqJvcPT8g41KydPAQmSHu5dnhzUCgYEAzcCy
JQguP4bvthUPnK+m0MacdaeYGKOlDdJUdWVXkxFVBTH7f1NhZzMHJRsfOPj+aP7Y
5lKEb/Zw4BvLsmtvO1W3OBWxi7kfziqs3oE3JUb7cJkEUG8RO2iBr03Lav7gNHdT
gZHjEWXPiILXXIi1J60i6DKqpPEep3zalDVvp3ECgYEArB0VNa6NIvz4bYIxhGmW
HzUkCWXLEkYX/2IOIxxmlS0cSGA4NNEHYtHs2JsOwwN/Tv8vAGWN68UVvlzlJwEu
OU5/Lq70e/2zEkyyp2WG4UtShuHg5LUPpu8MSq/kfZM/v88NI1PnfrrPvtTFaNSm
8RaMGKWA0pjFXRi5VVH8AukCgYEArs9WS9JMgNrjVT9FmWk/gMgN7IRs3BpFRJ37
9hYZNrGWG6jgtcggzG51VphZia07cfd8Twuyy/GE61OSWE+hHLdxyGveAglSNXba
kt0T/k71mC/Zk4EaEfBzpn6hIpHT6Z+sSjypwIFQtvn7gfoAd+pI6yEJDMRzt/eD
OkLAT3ECgYAcCkD2Qix+ymIYZ6axxGPOmFMNPtZN0sEAcwSluN42p57lotuvZepV
ggOL03QNS1l+VwHPNeHA1rcJI1TxRS0jmEvrCkpbqpuLddHZcmaQa3lMYJRrnIpi
wGP1SxJ50Q4QOOWqxpD3rVaNQWu2S/kTicJ+qMvj22MRU3ZbOjBsyg==
-----END RSA PRIVATE KEY-----
```


Figure 5-2 RSA Private key

5.3.2 Generate a Certificate Signing Request

A Certificate Signing Request is a message sent from an applicant to a Certificate Authority in order to apply for a digital identity certificate. It usually contains the public key for which the certificate should be issued, identifying information (such as a domain name) and integrity protection (e.g., a digital signature).

```
openssl req -new -key steve.key -out steve.csr
```

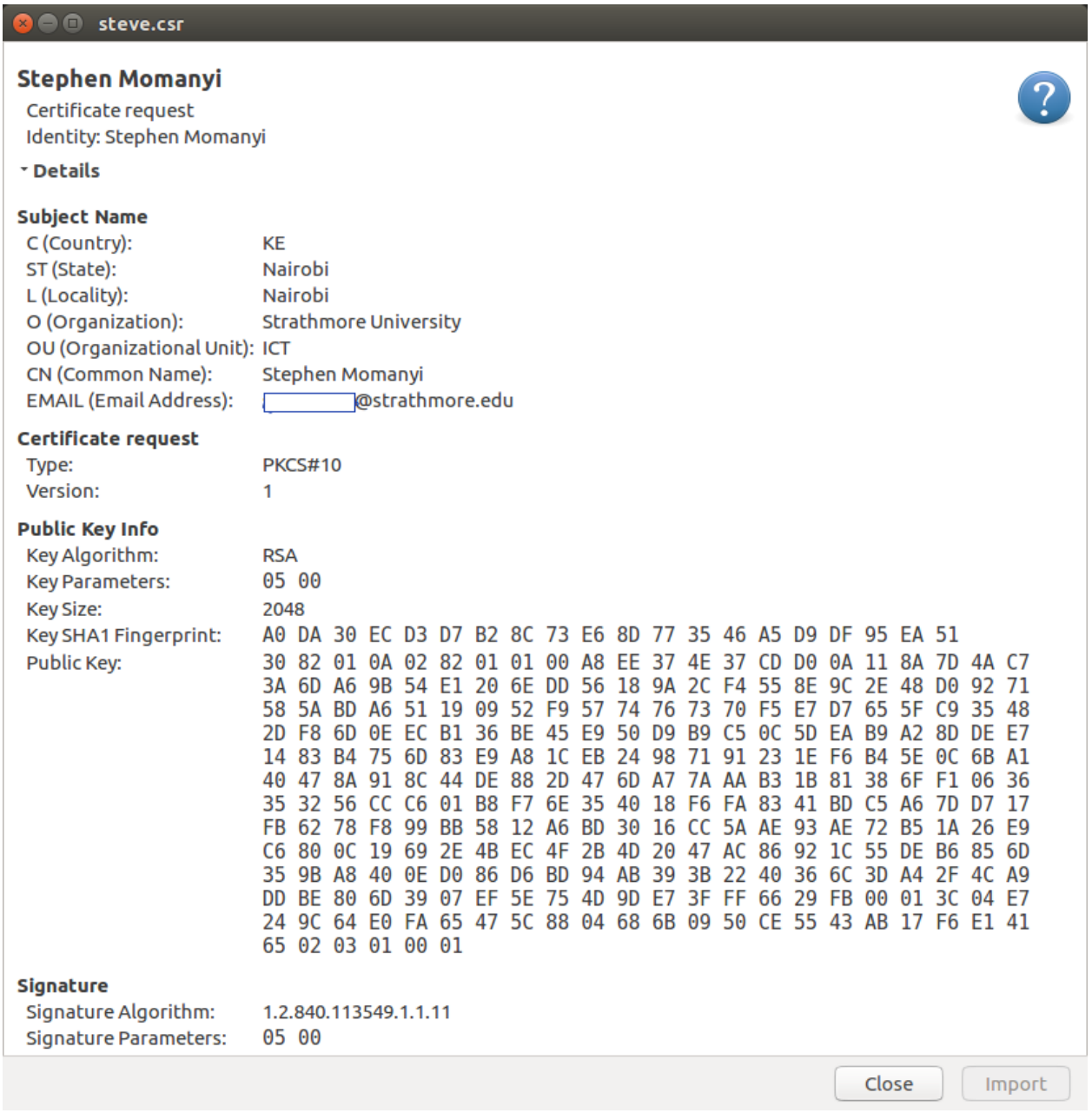

Figure 5-3 Certificate Signing Request

5.3.3 Generate a Self-Signed Public Certificate based on the CSR

openssl x509 -req -days 1095 -in steve.csr -signkey steve.key -out steve.crt

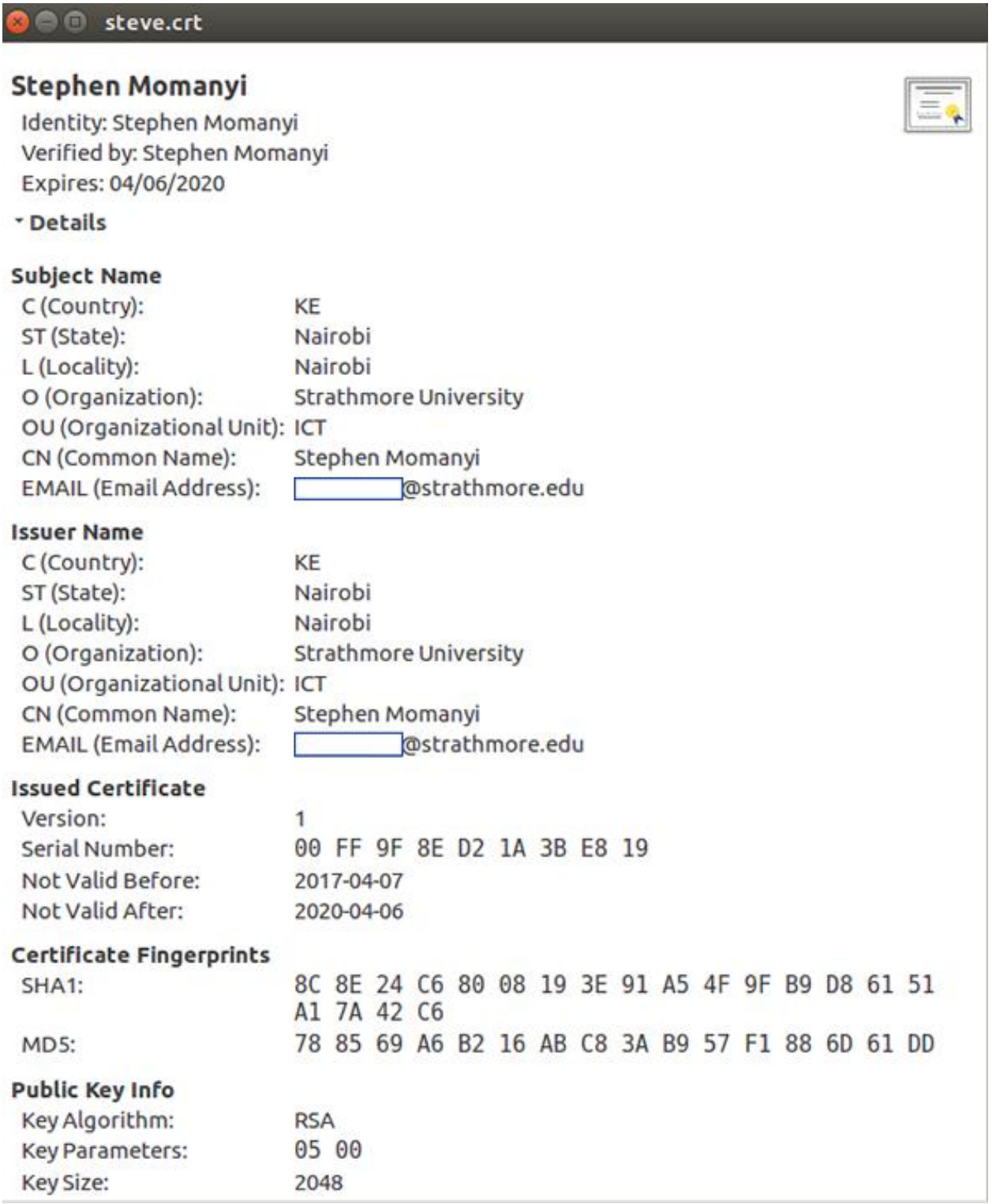

Figure 5-4 Self-signed Certificate

5.3.4 Generate a PKCS#12 file

The PKCS#12 or PFX format is a binary format for storing the server certificate, any intermediate certificates, and the private key into a single encryptable file. PFX files are usually found with the extensions ".pfx" and ".p12". PFX files are typically used on Windows machines to import and export certificates and private keys (SSH, 2015), table 5-3 gives the explanation of the command, while figure 5-5 show the actual file generated.

```
openssl pkcs12 -keypbe PBE-SHA1-3DES -certpbe PBE-SHA1-3DES -export -in steve.crt -inkey steve.key -out steve_pkcs12.pfx -name steve
```

openssl	the command for executing OpenSSL
pkcs12	the file utility for PKCS#12 files in OpenSSL
-keypbe PBE-SHA1-3DES	encrypt the generated key with DES in ede cbc mode (168 bit key)
-certpbe PBE-SHA1-3DES	encrypt the generated certificate with DES in ede cbc mode (168 bit key)
-export -in steve.crt	use steve.crt as the certificate the private key will be combined with.
-inkey steve.key	use the private key file steve.key as the private key to combine with the certificate
-out steve_pkcs12.pfx	export and save the PFX file as steve_pkcs12.pfx

Table 5-3 Summary of Command generate PFX file

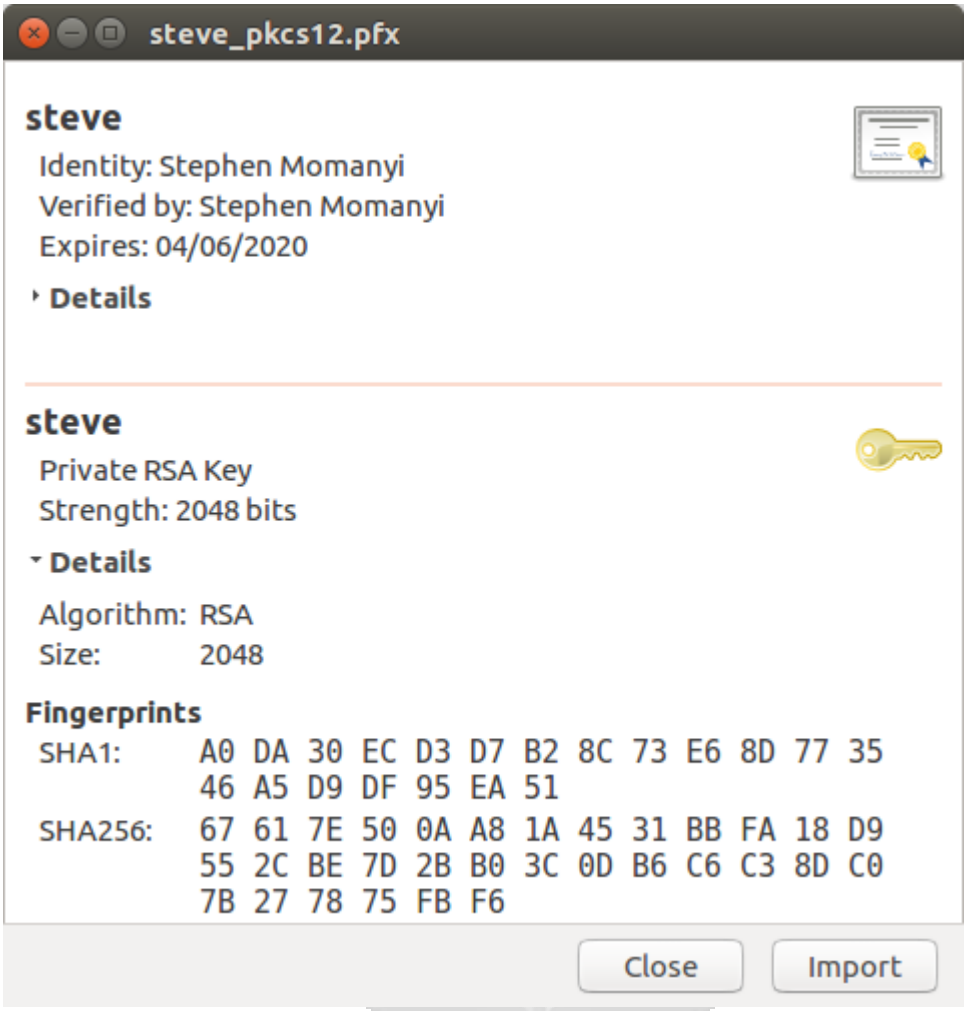

Figure 5-5 .pfx Contains Certificate and Private Key

5.3.5 Test Case for Digital Certificate Generation

"In software engineering, a test case is a set of conditions or variables under which a tester will determine if a requirement upon an application is partially or fully satisfied. It is given as an input to test whether functionality is working fine or not" (Mishra, 2014).

Test step	**Test step description**	**Expected rest**	**Actual result**	**Status**
Generate Private key	Process of generating private key using RSA encryption	Private key	Private key	Pass

Generate Certificate Signing Request:	Process of generating CSR	Certificate signing request file with extension .csr	Certificate signing request file	Pass
Generate a self-signed public certificate based on the request	Process of generating X509 certificate	X509 certificate with extension .crt	X509 certificate with extension .crt	Pass
Generate a PKCS#12 file:	Process of generating .pfx file	.pfx file	.pfx file	Pass

Table 5-4 Test Case for Digital Certificate Generation

5.4 Installation of Alfresco ECM

The chosen setup environment is Debian Linux. The setup wizard for Linux installs all the software and components that you require for running Alfresco. This setup wizard installs Alfresco and additional software, including: Tomcat application server, Solr4. Figure 5-6 shows the process of installing Alfresco.

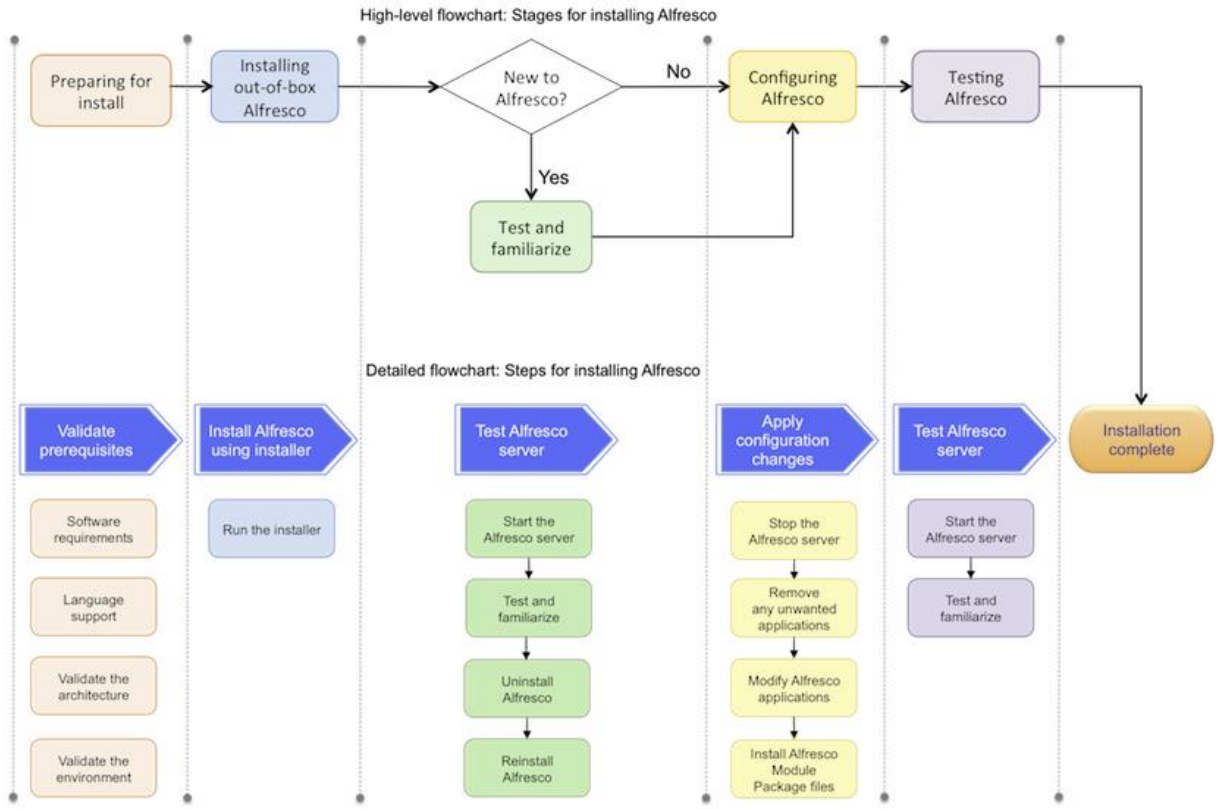

Figure 5-6 Steps for Installing Alfresco (Alfresco, 2017)

5.5 Alfresco Modules (AMPs)

A Module is a collection of code, configuration, scripts and media resources, contained in an AMP file. AMP files can be used to package custom templates, custom models, web scripts, UI customizations, and can be used to implement extensive additions to the Alfresco functionality.

Modules can be thought of as installable extensions to Alfresco. Once packaged into an AMP file format they can be applied to Alfresco using the Module Management Tool (MMT). When a module is installed using the MMT it is applied to the relevant WAR file. The Alfresco application typically consists of at least two WAR files: alfresco.war for the content server and share.war for the web client.

Larger, more complex modules can be distributed as two AMP files, one to be applied to the Alfresco WAR file and another to be applied to the Share WAR file (Alfresco, 2017). The digital signing module will be installed from an AMP file and applied to both Alfresco and Share WAR files.

5.5.1 Advantages of using AMP files

Afresco (2017) lists some of the advantages of packaging your extension in an AMP file:

i. AMPs provide a namespace feature that helps prevent file name clashes with other extensions.
ii. The minimum and maximum version of Alfresco required for the extension to operate correctly can be specified in the AMP file.
iii. Dependencies on other modules can be declared in the AMP file.
iv. AMP files can contain a module version number.
v. Data required by the extension when the module is loaded can be included in the AMP.
vi. It is possible to run initialization code when the AMP is installed.

5.5.2 Using the Module Management Tool (MMT)

The Module Management Tool (MMT) helps install and manage modules packaged as AMP (Alfresco Module Package) files. These AMP files are applied to a target WAR file. The MMT supports the following: installation of AMP files including upgrades to later versions, uninstallation of installed modules, and listing of currently installed modules. The MMT program, alfresco-mmt.jar, is available in the bin directory of the Alfresco installation and it will be used in the installation of the digital signing module.

5.6 Installation of Digital Signing Module

The digital signing modules will be downloaded from the host site and packaged to make an AMP file. The advantage of this method is that the code can be modified before being converted to the AMP format.

5.6.1 Building the Digital Signing Module

Table 5-6 shows the steps to build the AMP of the digital signing module.

Download files from the GitHub repository	Download files from: https://github.com/rouxemmanuel/DigitalSigning.git
Create Alfresco AMP	cd "DigitalSigningAlfresco" mvn install Output: digitalSigningAlfresco-1.1.0.amp
Create Share AMP	cd "DigitalSigningShare" mvn install Output: digitalSigningShare-1.1.0.amp

Table 5-5 Build Digital Signing Module

5.6.2 Install Digital Signing Module using Module Management Tool (MMT)

Table 6-6 shows the commands used to install the digital signing module using Alfresco's MMT.

Change directory to Alfresco bin directory	cd /opt/alfresco/app/bin/
Install module in Alfresco Core (alfresco.war)	java -jar alfresco-mmt-2.1.0.jar install /opt/digitalSigningAlfresco-1.1.0.amp /var/lib/tomcat7/webapps/alfresco.war
Install module in Alfresco Share (share.war)	java -jar alfresco-mmt-2.1.0.jar install /opt/digitalSigningAlfresco-1.1.0.amp /var/lib/tomcat7/webapps/share.war
List installed modules	java -jar alfresco-mmt.jar list /var/lib/tomcat7/webapps/alfresco.war

Table 5-6 Installation of Digital Signing Module

```
smomanyi@nyamindi: ~
root@nyamindi:/opt# cd /opt/alfresco/app/bin/
root@nyamindi:/opt/alfresco/app/bin# java -jar alfresco-mmt.jar list /var/lib/tomcat7/webapps/alfresco.w
ar
Module 'digitalSigning' installed in '/var/lib/tomcat7/webapps/alfresco.war'
   -    Title:        Alfresco Digital Signing Plugin
   -    Version:      1.1.0
   -    Install Date: Tue Jan 31 17:29:57 EAT 2017
   -    Description:   Alfresco Digital Signing Plugin
Module 'share-site-creators-repo' installed in '/var/lib/tomcat7/webapps/alfresco.war'
   -    Title:        share-site-creators-repo AMP project
   -    Version:      0.0.3
   -    Install Date: Thu Sep 15 08:12:37 EAT 2016
   -    Description:   Changes the permissions so that only members of a specific group can create sites
.
Module 'alfresco-aos-module' installed in '/var/lib/tomcat7/webapps/alfresco.war'
   -    Title:        Alfresco Office Services Module
   -    Version:      1.1
   -    Install Date: Thu Sep 15 08:09:28 EAT 2016
   -    Description:   Allows applications that can talk to a SharePoint server to talk to your Alfresco
 installation
Module 'alfresco-share-services' installed in '/var/lib/tomcat7/webapps/alfresco.war'
   -    Title:        Alfresco Share Services AMP
   -    Version:      5.2.0
   -    Install Date: Thu Sep 15 08:09:27 EAT 2016
   -    Description:   Module to be applied to alfresco.war, containing APIs for Alfresco Share
```

Figure 5-7 List of Modules installed in Alfresco

5.7 Digital Signing Process

The signing process is carried out within the system, therefore one must login to the system to carry out any operation. The documents to be signed have to first be uploaded into an Alfresco repository, this can be accessed through the "Document Library" in an Alfresco site.

A site is the user interface to interact with documents, assign user rights in an Alfresco system. Users can only see sites which they have rights to, and not any other, the access control is implemented at the application and site level.

5.8 Validation of the Model

Five of the Deans selected were observed while using the digital signing module, the process was easy for them, since web-based interface of which they were quiet familiar with.

A questionnaire was also administered to some users to get their impressions of the model, the summary of the findings is detailed in chapter 6 of this work.

5.8.1 Login using Active Directory credentials

The system provides integration with the Microsoft Active Director (AD) through Lightweight Directory Access Protocol (LDAP), this ensures that a person is authenticated with AD before he can proceed to use the system. AD provides several security measures, such as strong password enforcement and password expiry policy (a password expires after a set numbers of days).

Figure 5-8 System Login Window

5.8.2 Upload Digital Certificate

The self-signed digital certificate in X509 format, containing the public key and an identity of the individual is uploaded through the digital signing module. The user needs to provide a password which was set during the creation process for it to upload successfully. The user can also upload a scan of his signature which will be appended to the digital signature.

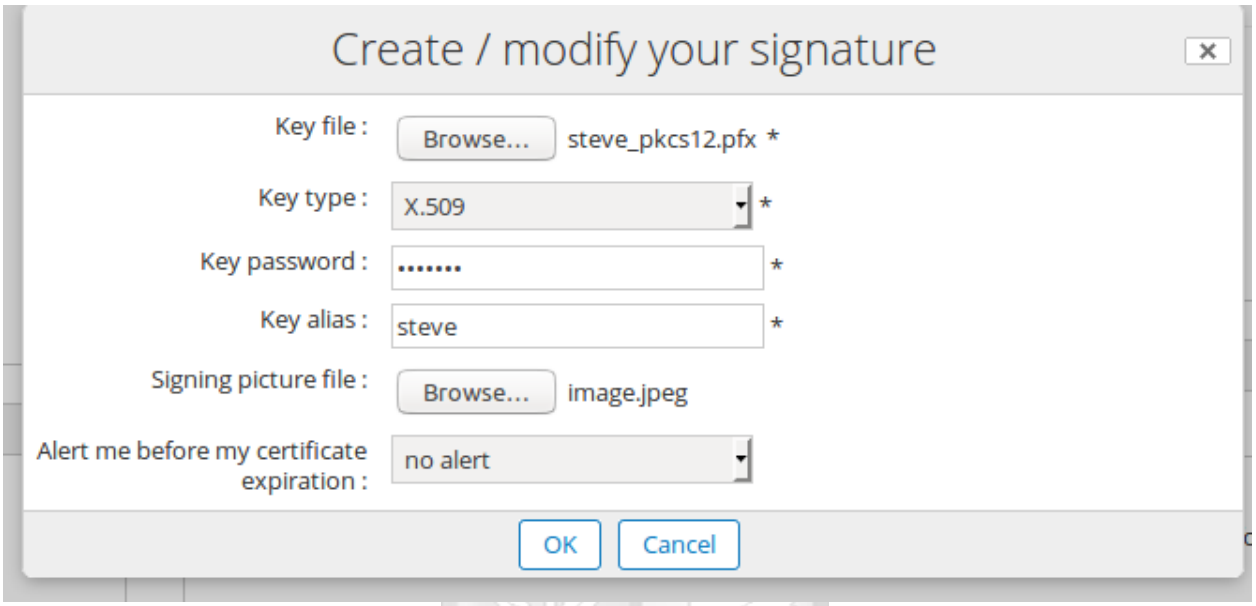

Figure 5-9 Upload of Digital Certificate

5.8.3 Digitally Signing Process

Figure 5-10 shows the process of digitally signing a document.

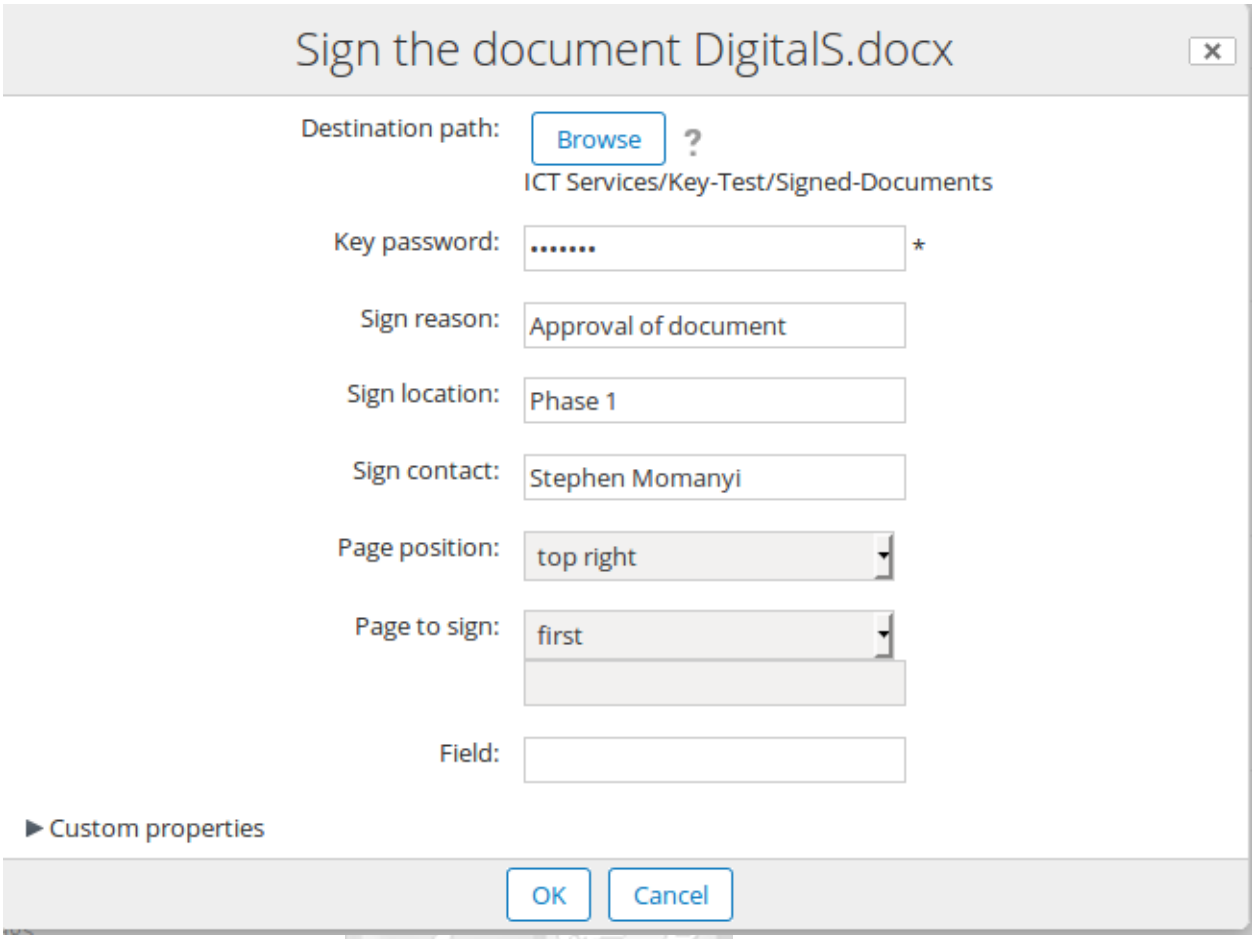

Figure 5-10 Document Signing Dialogue box

5.8.4 Digitally Signing Multiple Documents

Figure 5-11 shows the process of digitally signing multiple documents, this is convenient especially when a user has to sign many documents.

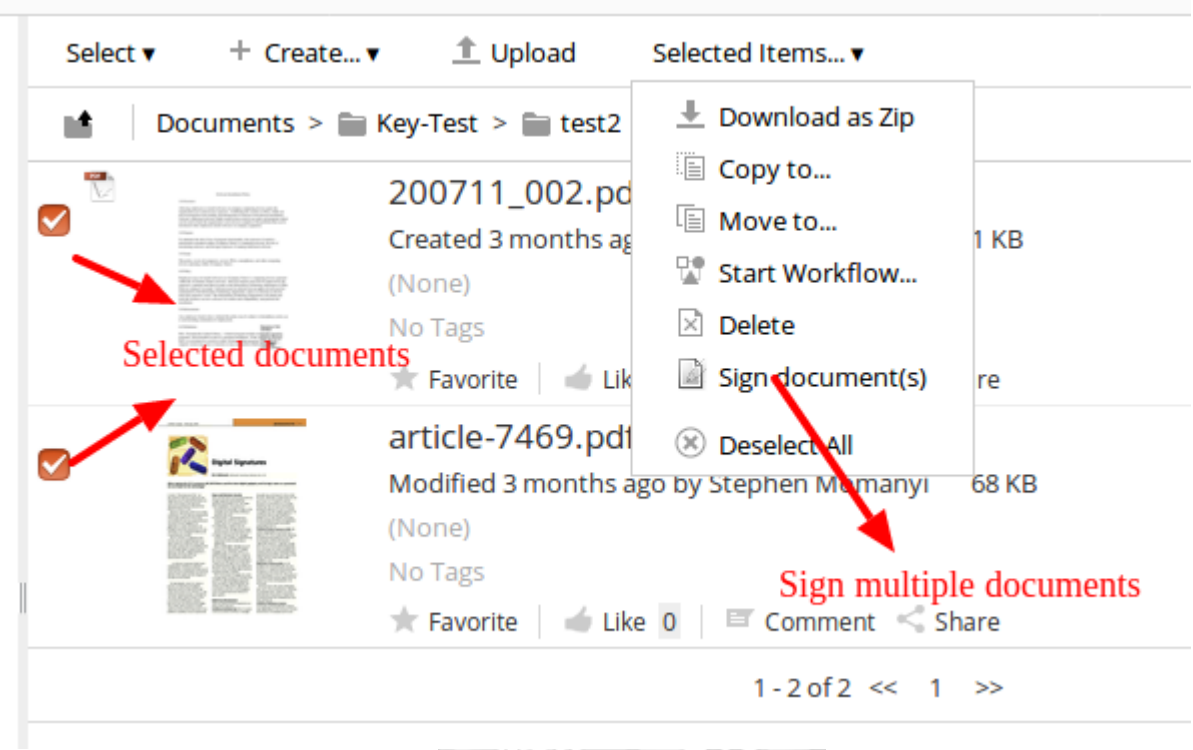

Figure 5-11 Digital Signature on Multiple Documents

5.8.5 Digitally Signed Document

Figure 5-12 below shows an example of a digitally signed document, with a timestamp imprinted on it and a watermark of the user's scanned signature.

Figure 5-12 Digitally Signed Document

5.8.6 Digital Signing Test Case

Table 5-7 shows the test case of the digital signing process.

Test step	**Test step description**	**Expected rest**	**Actual result**	**Status**
Login	Process of user login	Login successful	Login successful	Pass
User access rights	User gets rights to appropriate sites	View sites	View sites	Pass
Upload document	Process of uploading document	Uploaded document	Uploaded document	Pass
Digital certificate upload	Process of uploading certificate	Certificate uploaded	Certificate uploaded	Pass
Digital signing	Process of signing documents	Document signed	Document signed	Pass
Digital signing of multiple documents	Process of signing multiple documents	Documents signed	Documents signed	Pass

Table 5-7 Digital Signing Test Case

Chapter 6: Discussions

6.1 Discussions

The system was deployed, and a sample of the population tested the system and feedback was elicited from them using a questionnaire.

6.2 Findings

6.2.1 User Friendliness

The sampled users rated the user interface as friendly and easy to use, that is, 69% of the users found the system usable while the other 23% were indifferent and 8% did not find the it user friendly, as shown in figure 6-1.

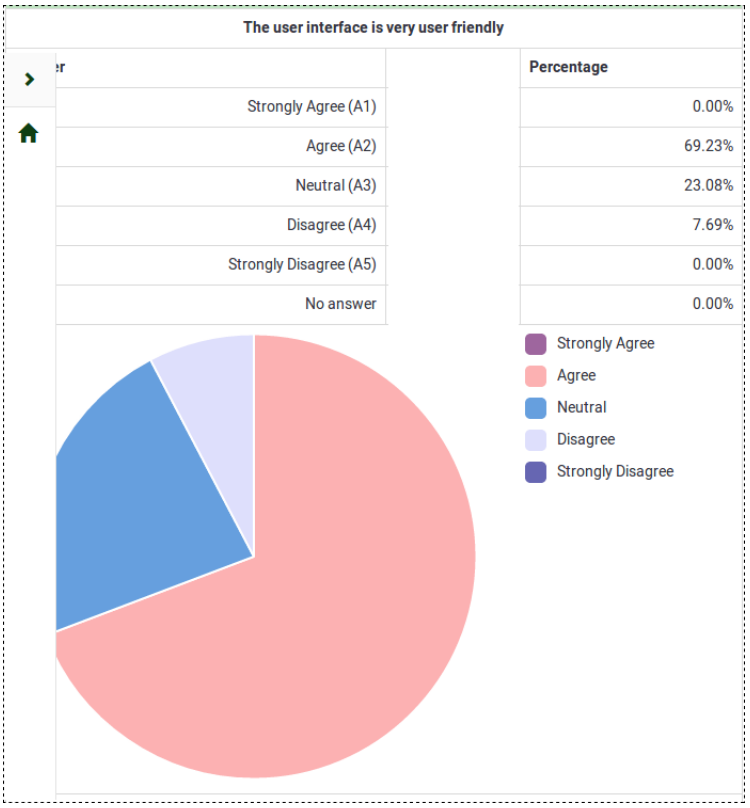

Figure 6-1 User Friendliness

6.2.2 Ease of uploading Documents

The system requires users to upload documents in readiness for digital signing, of the users sampled 38% felt that the process was very easy, while the other 62% thought that the process was relatively easy, as shown in figure 6-2.

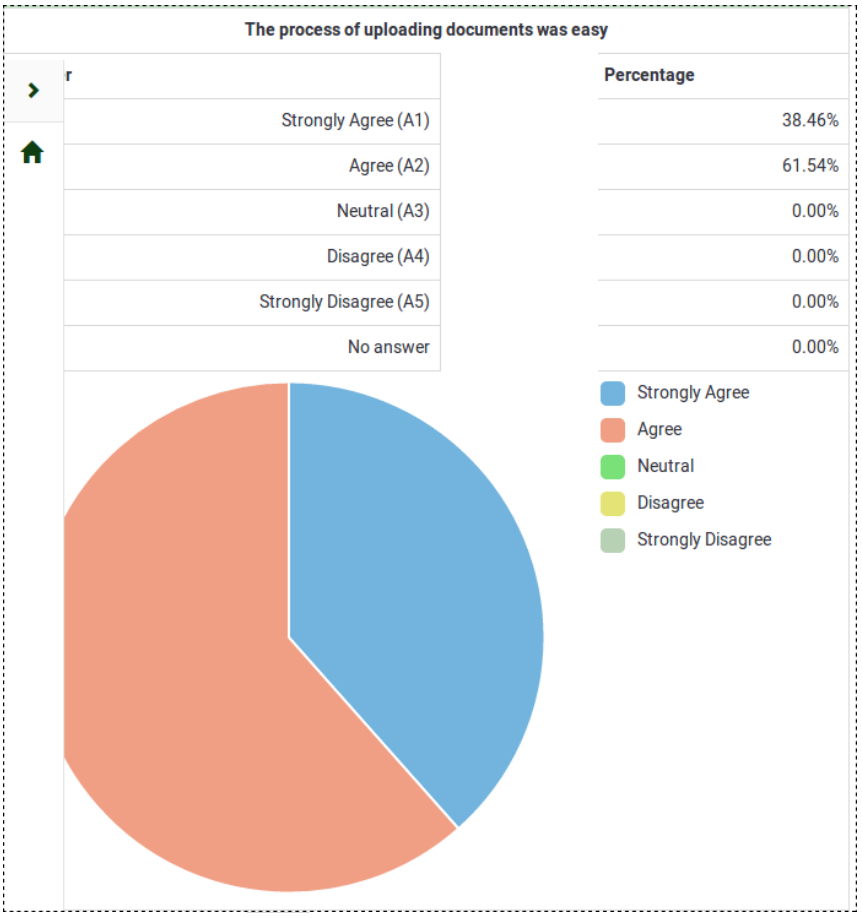

Figure 6-2 Ease of Uploading Documents

6.2.3 Search Facility

A search facility will aid the users to quickly find documents in the system. Figure 6-3 shows that, 15% of the users strongly felt that the search functionality exceeded their expectations, while 77% thought that the functionality meet their expectations. There other 8% were did not express any strong sentiments about the functionality.

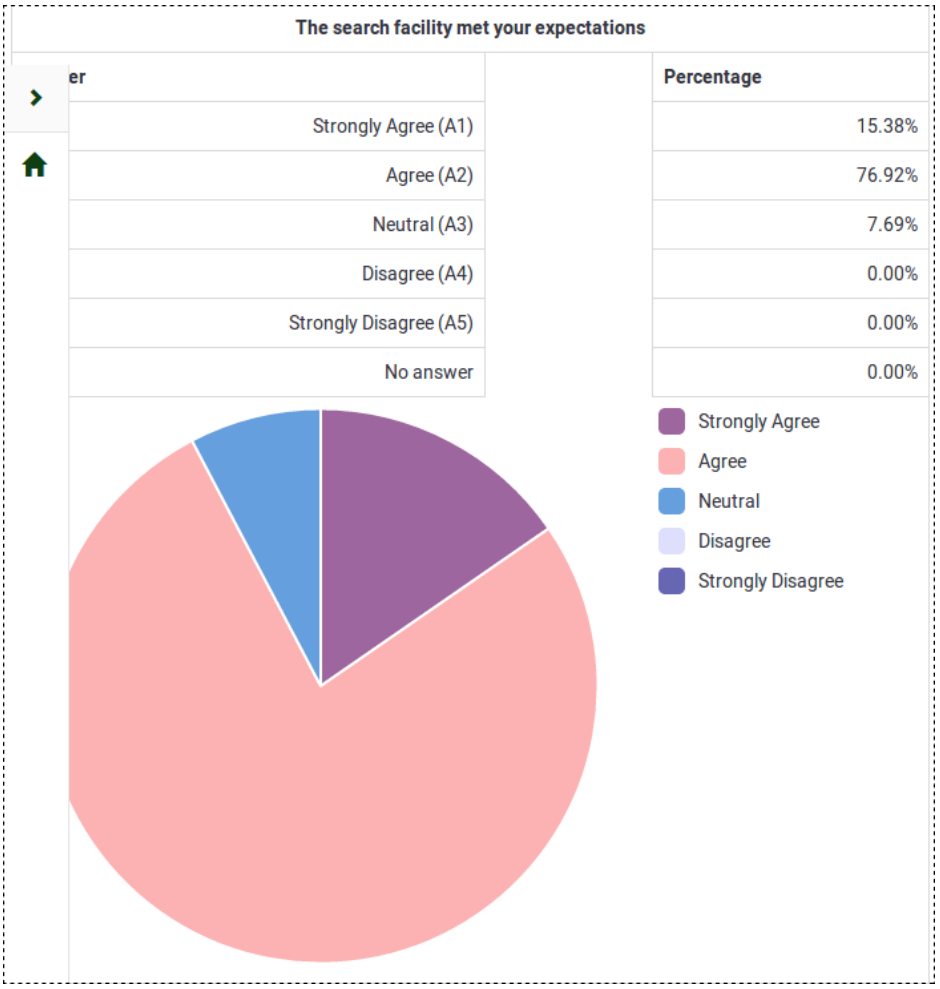

Figure 6-3 Search Facility

6.2.4 Convenient Document Sharing

The system provides a platform for collaboration within the workplace. 46% of the users sampled expressed the strongly agreed that the model provided a good platform for collaboration, while the remaining 54% agreed that it was convenient for document sharing. Figure 6-4 shows the results of the survey.

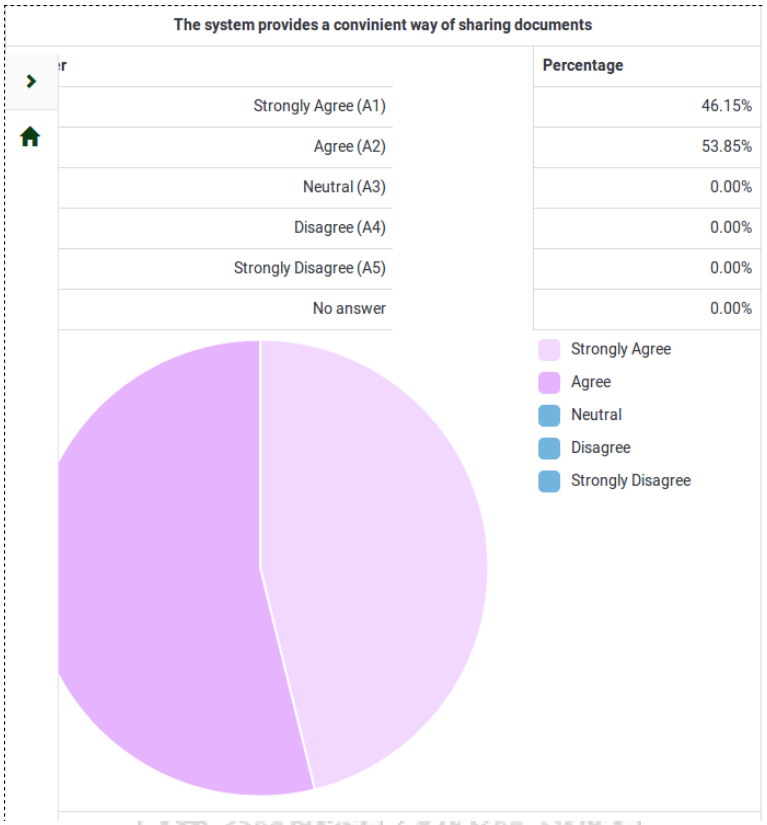

Figure 6-4 Documents Sharing

6.2.5 Other Documents Sharing Tools

There is need to get the users to store work related documents within the local university network rather than on the cloud, this will enhance control of the organisation over its data. Figure 6-5 shows that, 69% of the users used Google Drive, 8% used Email while 23% used DropBox.

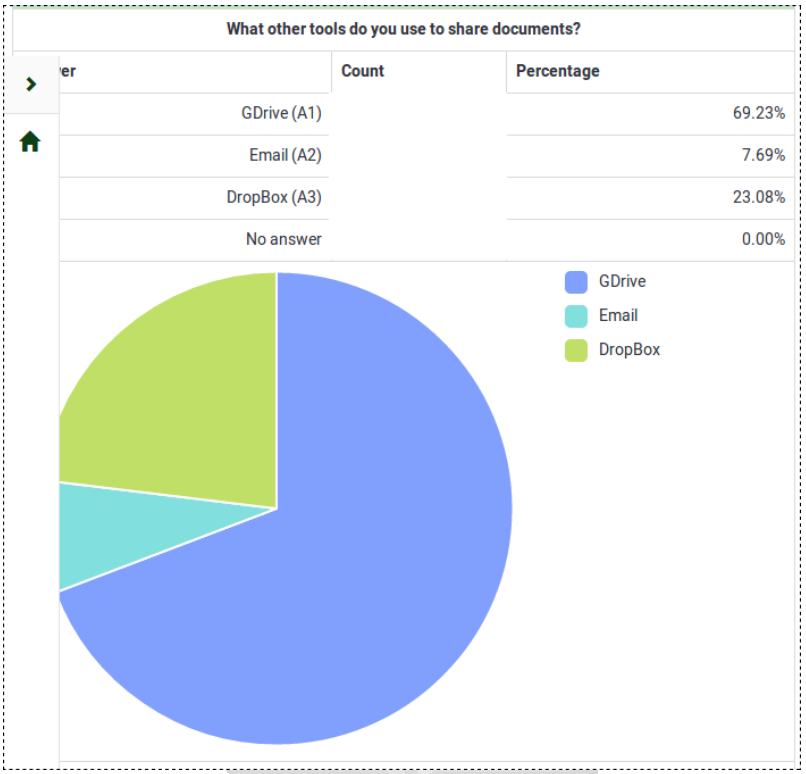

Figure 6-5 Other Documents Sharing Tools

6.2.6 Security of the Model

23% of the users thought of the system as being secure, 23% thought it very secure while 46% thought of it as moderately secure, as shown in figure 6-6. None of the users thought of the system as not being secure.

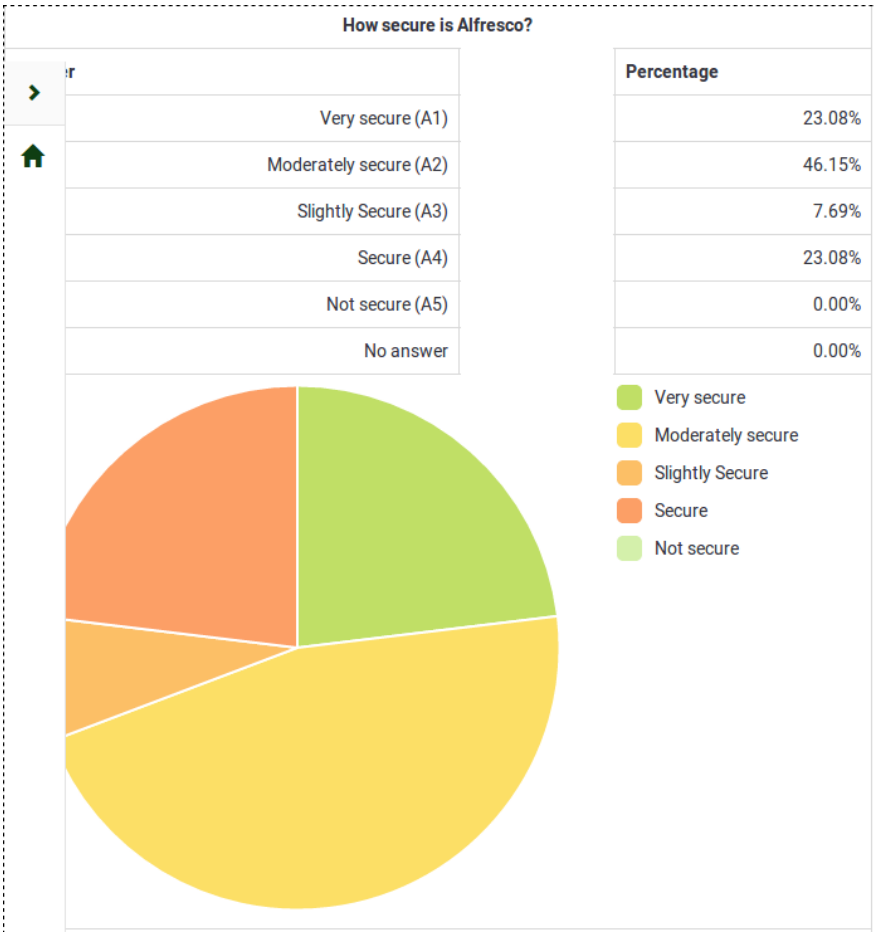

Figure 6-6 Security of Alfresco

6.2.7 User Training

User training is important to help users utilize the full potential of the system, of the sampled users 46% were trained will 54% were not trained due to time limitations. The untrained users however, managed to use the system with relative ease. Figure 6-7 shows the result for this question.

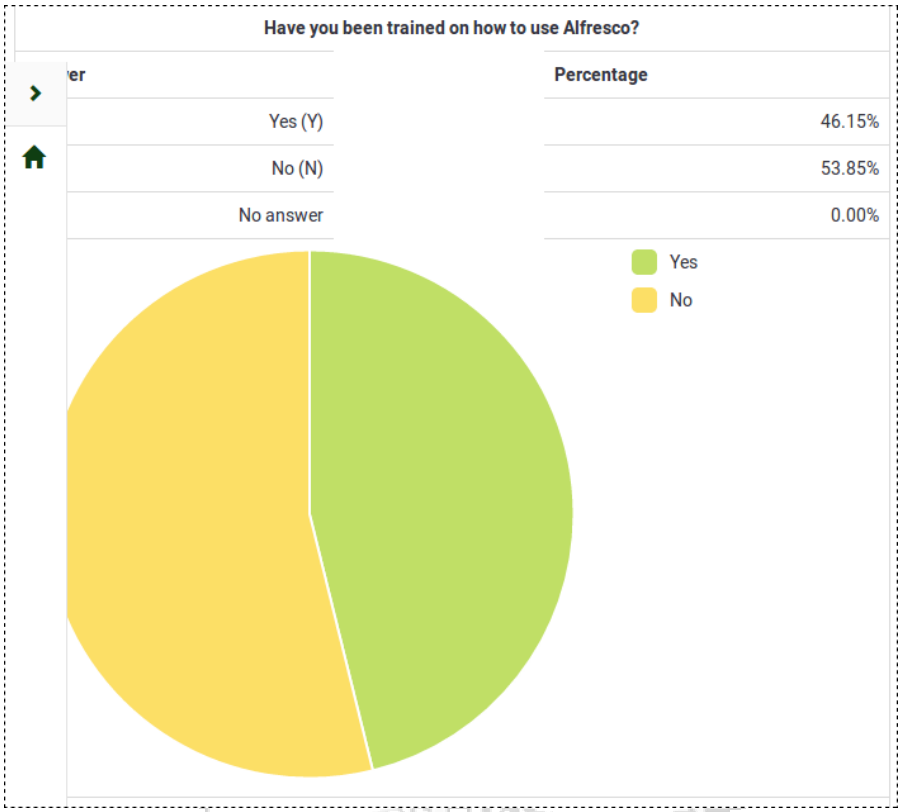

Figure 6-7 User Training

Chapter 7: Conclusions and Recommendations

7.1 Conclusions

Information security is a major aspect in any system, various techniques are employed to ensure confidentiality, integrity, authenticity and availability of a system. The use of digital signatures will guarantee the authenticity and non-repudiation of the signed documents.

The digital signing model was implemented using cryptography standards that has been in existence for a long time and have stood the test of time. The RSA encryption technique was used in generation the key pair that was used to create the digital signature. To facilitate the of use of the digital signatures, Alfresco ECM was used to enable the end user to digitally sign documents, it was selected because it is being used within the university and it also has many useful features.

These security measures are to ensure that the digital documents stored in the system are free from any manipulation. The examination coordinators are assigned rights to upload the marksheets for their respective schools and the Deans of those schools approve the marksheets by appending their unique digital signatures. A digitally signed message or document cannot be altered without invalidating the signature. This is true whether the message is encrypted or not.

The move to store the marksheets in a digital format has reduced the amount of work required to physically sign the manual marksheets. Another benefit of the move is that it has removed the need the use filing cabinets which occupied a large space.

The research set out to identify the limitations of the process of physically storing consolidated marksheets. These limitations were: the amount of physical paper generated, the effort required to sign and store paper documents.

The process of generating a digital certificate was tested and successfully installed in the digital signing module as illustrated in chapter 5. Digital certificates generated with a large key size are more secure and virtually unbreakable using brute-force attacks as shown in Figure 2-6.

7.2 Recommendations

To improve on the digital signing model, the following can be implemented:

i. The digital certificate can be verified by a Certificate Authority to provide an extra layer of validation.

ii. Other encryption techniques such as Elliptic Curve Cryptography (ECC) which it appears to offer equal security for a far smaller bit size, thereby reducing processing overhead.

References

Stallings, W., & Brown, L. (UNSW C. at the A. D. F. A. (2015). *Computer Security Principles and Practice* (3rd ed.). Pearson.

Commission on Systemic Interoperability. (2005). Ending the Document Game: Connecting and Transforming Your Healthcare Through Information Technology.

Robiette, A. (2001). Digital certificates and public key infrastructure. *VINE*, *31*, 42–49.

WaveMaker. (2017). Rapid Application Development Model. Retrieved from https://www.wavemaker.com/rapid-application-development-model/

CUE. (2017). Commission for University Education - Mission and Vision. Retrieved from http://www.cue.or.ke/index.php/about-us/vision-and-mission

Digitary. (2017). Secure Online Academic Credentials. Retrieved from https://www.digitary.net/

EXT-NG. (2017). Verify Degrees and Certificates obtained in Nigeria. Retrieved from https://www.etx-ng.com/certverify-for-companies

HEDD. (2017). Higher Education Degree Datacheck. Retrieved from https://hedd.ac.uk/index.htm

iCertify. (2017). Online Degree Verification. Retrieved from http://icredify.com/index.html

Nation Media Group. (2012). Missing marks delay university students' dreams of degrees. Retrieved from http://www.nation.co.ke/news/Missing-marks-delay-students-graduation/1056-1634660-9c3wg/index.html

Whitten, J. L., & Bentley, L. D. (2007). Systems Analysis and Design Methods. McGraw-Hill (7th ed.).

Daud, N. M. N., Bakar, N. A. A. A., & Rusli, H. M. (2010). Implementing rapid application development (RAD) methodology in developing practical training application system. In *2010 International Symposium on Information Technology* (Vol. 3, pp. 1664–1667).

Rao, U. H., & Nayak, U. (2014). *The InfoSec Handbook. Apress Media, LLC*.

Hassler, V., & Biely, H. (1999). Digital signature management. *Internet Research*, *9*, 262–271.

Davies, J. (2011). Implementing SSL / TLS Using Cryptography and PKI. John Wiley and Sons.

Alfresco. (2017). Architecture | Alfresco Documentation. Retrieved from http://docs.alfresco.com/5.1/

Mishra, G. (2014). Test Case Vs Test Scenarios. Retrieved from http://www.360logica.com/blog/2014/05/test-case-vs-test-scenarios.html

AIIM. (2017). What is Enterprise Content Management (ECM)? Retrieved from http://www.aiim.org/What-is-ECM-Enterprise-Content-Management

Liddy, C., & Sturgeon, A. (1999). The evolution of certificate model architecture. *Information Management & Computer Security*, *7*, 95–100.

Oracle. (2013). The Java Virtual Machine Specification. Retrieved from http://docs.oracle.com/javase/specs/jvms/se7/jvms7.pdf

Schildt, H. (2002). *Java 2: The Complete Reference* (5th ed.). McGraw-Hill/Osborne.

Borroy, A. (2015). Electronic Signature addons. Retrieved from https://github.com/OrderOfTheBee/addons/wiki/Electronic-Signature-addons

Paganini, P. (2012). What is a digital signature? Fundamental principles. Retrieved from http://securityaffairs.co/wordpress/5223/digital-id/what-is-a-digital-signature-fundamental-principles.html

Mugenda, O., & Mugenda, A. (1999). *Research Methods Quantitative & Qualitative Approaches* (2nd ed., Vol. 1). Act Press.

Kothari, C. R. (2004). *Research Methods: Methods and Techniques* (2nd ed.).

Jervis, M., & Masoodian, M. (2014). How do people attempt to integrate the management of their paper and electronic documents? *Aslib Journal of Information Management*, *66*(2), 134–155.

Wilson, S. (1999). Digital signatures and the future of documentation. *Information Management & Computer Security*, *7*(2), 83–87.

Wilson, S. (1997). Certificates and trust in electronic commerce. *Information Management & Computer Security*, *5*(5), 175–181.

Johnston, G. P., & Bowen, D. V. (2005). The benefits of electronic records management systems: a general review of published and some unpublished cases. *Records Management Journal, 15*(3), 131–140.

Vonka, J. (2014). Deep Dive: Alfresco Core Repository. Retrieved from https://www.slideshare.net/jvonka/summit2014-core-repo

Haug, A. (2012). The implementation of enterprise content management systems in SMEs. *Journal of Enterprise Information Management, 25*(4), 349–372.

SSL. (2015). Create a .pfx/.p12 certificate file using OpenSSL. Retrieved from https://www.ssl.com/how-to/create-a-pfx-p12-certificate-file-using-openssl/

Appendices

Appendix A

Interview Guide for Examination of Staff

1. What challenges do you face with the current way of operations?

   …….

   …….

   …….

2. What features do you require for the digital storage of marksheets?

   …….

   …….

   …….

3. What level of security is needed?

   …….

   …….

   …….

4. Who will be the users of the system?

   …….

   …….

   …….

5. What operations are the users expected to carry out in the current system?

   …….

   …….

   …….

6. How often are the documents generated?

   …….

…….

…….

7. How much time is acceptable downtime in new system?

…….

…….

…….

Appendix B

Secure Electronic Document Management Survey

Dear Respondent,

This research is will be used for academic purpose only. Its main objective is to find out users' experience in using Alfresco. Kindly provide your honest opinion on the same. Please note that your responses will be treated as private and confidential.

Thank you

Which department do you work in? *

Choose one of the following answers

Please choose **only one** of the following:

- ○Administration Services Department
- ○School of Management and Commerce
- ○Information Communication Technology Department
- ○Faculty of Information Technology
- ○Admissions Services Department
- ○Strategy
- ○School of Humanities and Social Sciences
- ○Library
- ○Strathmore Institute of Mathematical Sciences
- ○Centre for Tourism and Hospitality
- ○Human Resources Department
- ○Student Mentoring Services
- ○Examination Services Department
- ○Office of the Registrar
- ○@iLabAfrica
- ○Legal and Governance Services
- ○Strathmore Law School

- ◯Office of the DVC (Planning & Development)
- ◯CTH-Cafeteria
- ◯Security & Safety Department
- ◯Strathmore Energy and Research Centre

Have you been trained on how to use Alfresco? *

Please choose **only one** of the following:

- ◯Yes
- ◯No

How often do you use Alfresco? *

Choose one of the following answers

Please choose **only one** of the following:

- ◯Daily
- ◯Weekly
- ◯Monthly

The user interface is very user friendly *

Choose one of the following answers

Please choose **only one** of the following:

- ◯Strongly Agree
- ◯Agree
- ◯Neutral
- ◯Disagree
- ◯Strongly Disagree

The navigation system is intuitive and helpful *

Choose one of the following answers

Please choose **only one** of the following:

- ◯Strongly Agree
- ◯Agree
- ◯Neutral
- ◯Disagree
- ◯Strongly Disagree

The search facility met your expectations *

Choose one of the following answers

Please choose **only one** of the following:

- ○Strongly Agree
- ○Agree
- ○Neutral
- ○Disagree
- ○Strongly Disagree

How secure is Alfresco? *

Choose one of the following answers

Please choose **only one** of the following:

- ○Very secure
- ○Moderately secure
- ○Slightly Secure
- ○Secure
- ○Not secure

The system provides a convenient way of sharing documents *

Choose one of the following answers

Please choose **only one** of the following:

- ○Strongly Agree
- ○Agree
- ○Neutral
- ○Disagree
- ○Strongly Disagree

What other tools do you use to share documents? *

Choose one of the following answers

Please choose **only one** of the following:

- ○GDrive
- ○Email
- ○DropBox

The process of uploading documents was easy *

Choose one of the following answers

Please choose **only one** of the following:

- ○Strongly Agree
- ○Agree
- ○Neutral
- ○Disagree
- ○Strongly Disagree

Any other comments about Alfresco?

Please write your answer here:

……

……