Modeling a Secured Automated Hospital ManagementSystem

Ngonadi Ifeoma Vivian[1] andOsuagwu Eberechi Oliver[2]
Department of Computer Science and Information Technology
Petroleum Training Institute, Effurun. DeltaState
ifeomaojingonadi@gmail.com
[2]Department of Computer Science
ImoStateUniversity, Owerri
profoliverosuagwu@gmail.com

Abstract

Many hospitals employ the manual hospital management system in which all their processes are done manually. The manual system has a lot of challenges ranging from improper record keeping, waste of time and effort and inadequate care of the patients. The automated Hospital Management System is a computer based system which facilitates managing the functioning of the hospital and makes the whole functioning paperless while overcoming the challenges faced by the manual system. This paper develops a complete and secured automated Hospital Management System which automates all the processes in the system ranging from the registration of the patient, lab investigation, admission and drug dispensary. This was achieved by developing an application in java called the IntegratedHospital Information Management System which automates the entire hospital processes. In order to prevent unauthorized access to patients' records, the records are encrypted when they are generated and only decrypted by authorized personnel. A second application called the Simulated Patient Data Software was developed for the encryption and decryption of patients' medical records. The result of this research work is a complete automation of a Hospital Management System which is secured.

1. Introduction

Most hospitals employ the manual hospital management system in which all their processes starting from registration to admission and dispensary of drugs are done manually. The manual method of hospital management faces a lot of challenges ranging from improper record keeping, waste of time, effort and money, lack of job satisfaction on the part of the nurses and attendants and even doctors and inadequate care of the patients.It is alarming that in a paper-based environment, medical mistakes frequently approach 40%, of which 39% are made at the point of prescription, 12% by transcription errors and 11% in dispensing Wicks et al (2006).

An automated Hospital Management System (HMS) is a computer or web based system whcih facilitates managing the functioning of the hospital or any medical set up. This system or software will help in making the whole functioning paperless. It integrates all the information regarding patients, doctors, staff, hospital administrative details etc. into one software. It has sections for various professionals that make up a hospital.

Some hospitals now employ the automated hospital management system in which their processes are automated but they fail to secure the patients' medical records since they travel through networks. Communication of health related information has to be strictly private and confidential to protect patient privacy. The data sent over the network and wireless transmission is prone to different types of attack such as eavesdropping, sending false values or replay of previous data. Medical professionals have to be certain that the data are not tampered in transit or at a point of origin as proper diagnosis requires accurate data.

As the patients' medical records travel across networks, it is pertinent that they are protected against unauthorized access which might intercept the data along the way.In view of this, the patients' medical records are encrypted as they travel across the networks. This encryption is done by using the symmetric key encryption algorithm which encrypts the patients' medical records as they are generated and only decrypts them when they are needed. This ensures that unauthorized users are denied access.

2. Related Work

Ahmed et al presented a paper titled 'A Hospital Resource and Patient Management System Based on Real Time Data Capture and Intelligent Decision Making'. The paper highlights the shortcomings of existing hospital information systems and proposes a location, identification and communication system that is able to overcome some of the inadequacies of existing systems and improve the efficiency and throughput of the various units of the hospital setup. Their system was based on Radio Frequency Identification (RFID) and wireless sensor network-based location and information management framework and the technology that would allow hospital assets, personnel and patients to be tracked in real time for the purpose of optimizing operations in all aspects of the daily activities of hospitals. The RFID-based system is linked to a Wireless Body Area Network (WBAN) which collects physiological information about a patient and opens new possibilities for a remote patient care.

Ahmed et al identified their challenges to be assuring hardware fidelity, software functionality, data reliability and overall long-term system dependability. The major gap in this research work is the lack of provision of security architecture for the work. The patients' physiological data are collected and sent through networks but no security measure is put in place to safeguard the patients' medical records as they travel through the networks.

The work presented by Daiping et al on 'Study on Information system of Health Care Services Management in Hospital' reviews the Health Information Systems (HIS) which are widely used in many hospitals in China mainly to provide easier and faster way for daily medical tasks. The work proposes Hospital Services Management System (HSMS) which aims at improving quality of services, identifying cost reduction areas, analyses and evaluates health care services.

Christopher Bain's research work 'Developing Effective Hospital Management Information System' focuses more on the needs of hospital manager and the ecosystem in which it operates, the internal and external environment shaping factors which bears an impact or association on daily hospital activities and decision making process that the hospital manager has to go through in each situations. Some of the major challenges identified are high demand pressure, greater customer satisfaction level and low profit margins.

Hubner et al presented a paper on the 'Specification of a Reference Model for the Domain Layer of a Hospital Information System. The paper aimed at creating a reference data model that will serve as a generic starting point for any new Hospital Information System development projects so costs involved in studying and analyzing current state and coming up with gaps analysis and additional requirements can be significantly reduced. The paper again did not consider the security of patients' medical records.

Olusanya et al's work on 'Design and Implementation of Hospital Management System using Java' provides the benefits of streamlined operations, enhanced administration and control, superior patient care, strict cost control and improved profitability. The system used java as the front end software which is an Object Oriented Programming language and has connectivity with the back end software.

Digvijay et al's work on 'Hospital Management System' dealt with the computerization of the front office management of hospital to develop a software which is user friendly, simple, fast and cost effective. It deals with the collection of patient's information, diagnosis details etc. These functions were performed manually before but the function of the new system is to register and store patient and doctor details and retrieve these details when required and to manipulate these details meaningfully.

3.n Methodology and System Analysis

The existing system is a complete manual system where all the processes are done by hand. The system has a lot of paper work. When a patient visits the hospital, he obtains a card at which point a file is opened for him. He goes to see the doctor who then decides what type of treatment that is good for the patient. He can send the patient to the lab for further investigation after which he prescribes drugs or further investigation for the patient. All information is entered in the patient's

file and the file is kept in the file cabinet. When the patient comes to the hospital again, the file is retrieved for him to see the doctor.

The dataflow diagram of the existing system is shown in figure 1.

The proposed system is a completely automated system where all the patients' records are kept in the computer. It is a complete paperless system. When a sick patient visits the hospital, he goes to the registration center where his personal information is collected. The patient's biometrics are also taken and kept in the computer. A file is created for the patient and kept in the computer system. At this point, the patient goes to see the doctor who examines him and prescribes treatment for him. The doctor can send him to the lab for further investigation. The result of the lab investigation is sent to the server where the doctor can have access to it. All the computers in the hospital are connected through a network. A local area network is used to connect all the computers in the hospital and a wide area network is used to connect to the computers in the other branches of the hospital in different locations. The doctor sees the patient's medical records in his computer and proffers a solution to the patient's problem. When the patient visits the hospital again, he will go the registration centre where his file will be uploaded for the doctor to see irrespective of the computer he is using, this solves the problem of having to look for the patient's file in the manual method in the existing system.

Since the patients' medical records travel across networks, it is very important to safeguard the patients' medical records. This is done by encrypting the records as they travel through the networks and decrypting them at the other end.The dataflow diagram of the proposed system is shown in figure 2

.

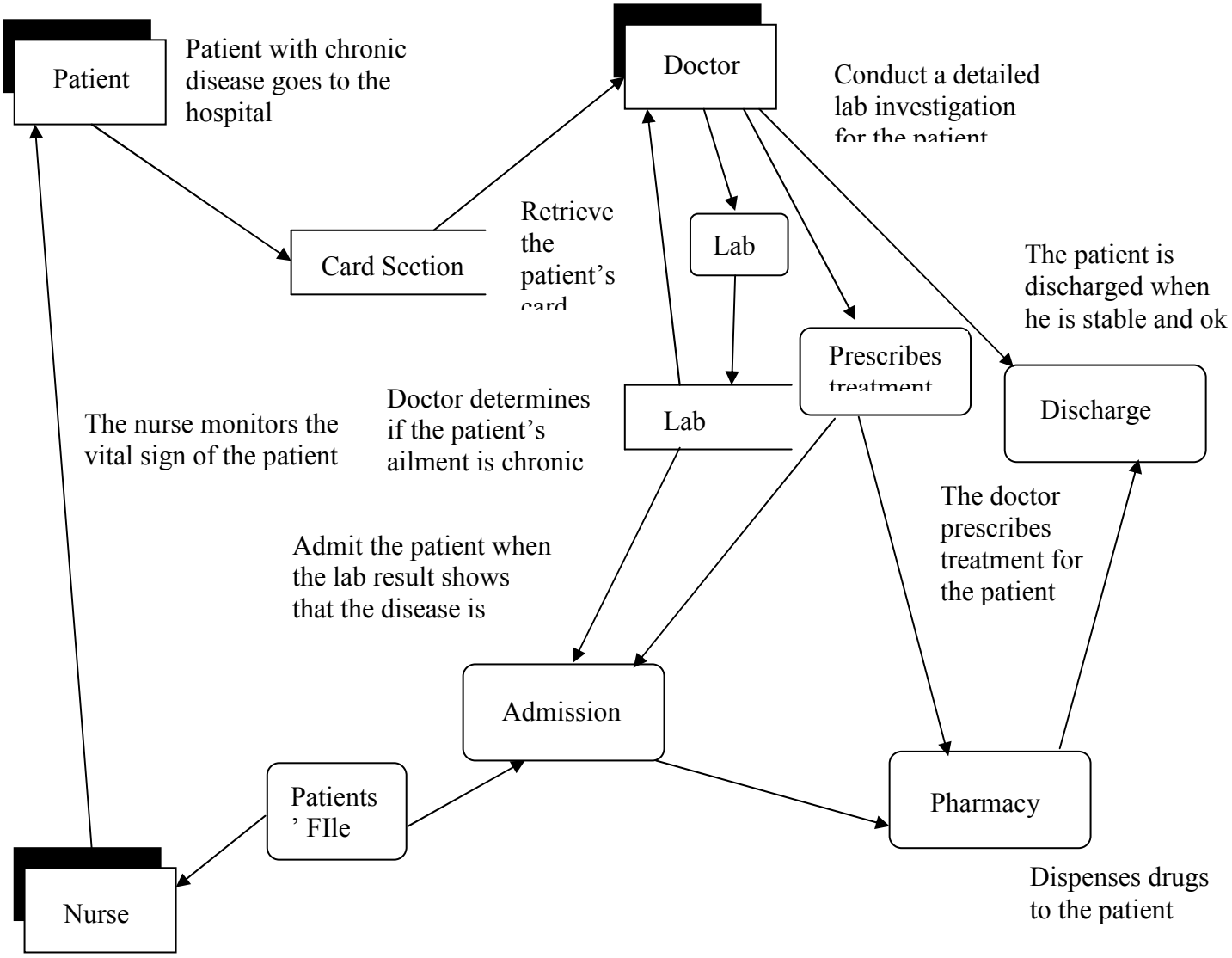

Figure 1: Data Flow Diagram of the Existing

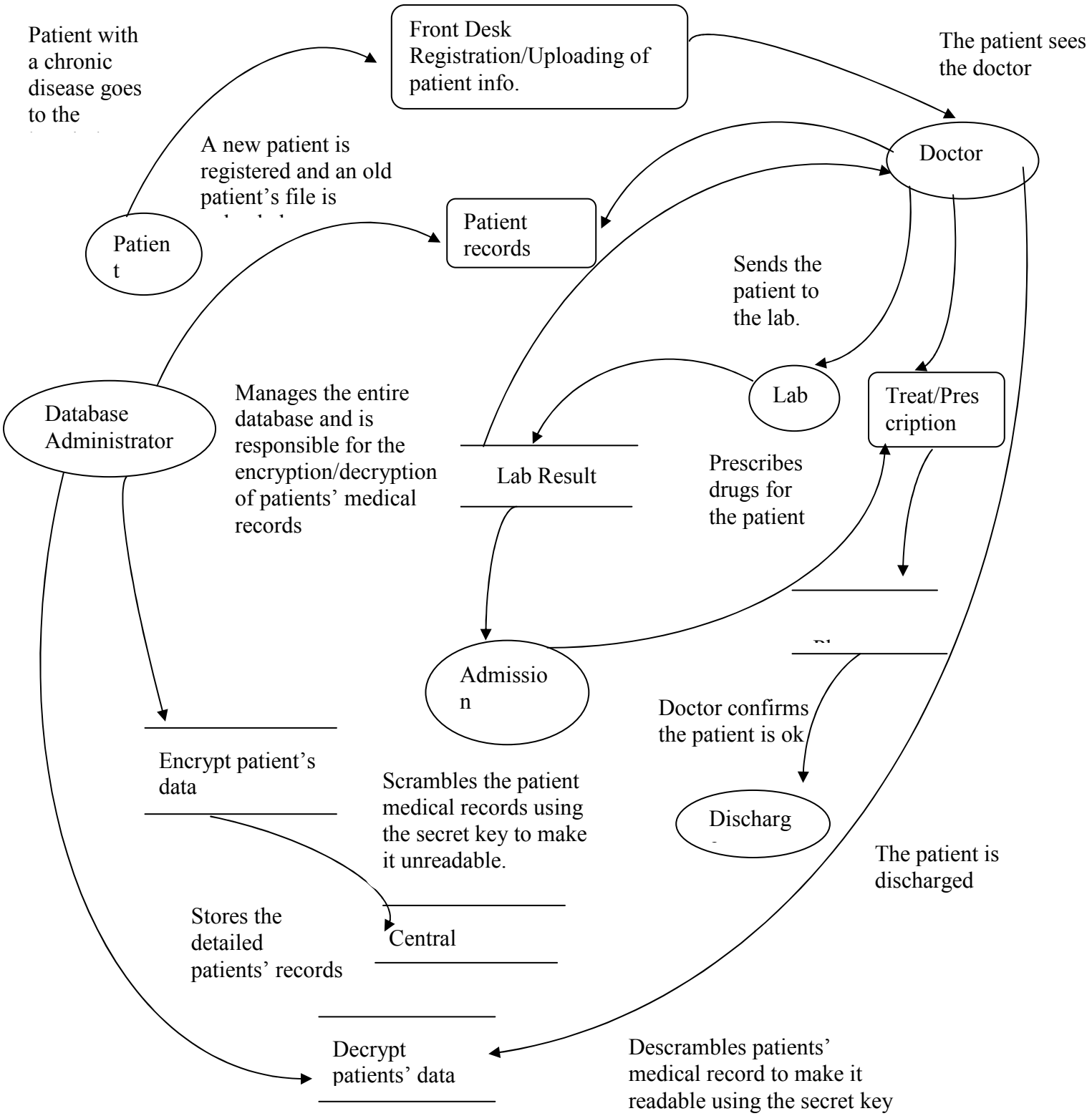

Figure 2: Dataflow diagram of the proposed

The automation of the hospital was done using the java language. The software was developed in Java because of its web based features and its portability The java language was used to develop an application called the Integrated Hospital Information Management System which performs several hospital functions ranging from registration of new patients, in-patient admission master list, out-patient admission master list, drug inventory, hospital discharge reports, medical staff information and experts information. The database was created using MY SQL where the records are stored. A second application was also developed in java called Simulated Patient Data Software (SPDS). This software is responsible for the encryption and decryption of the patients' medical records.

In the encryption, the original text called the plaintext is encrypted using an encryption algorithm, to generate the encrypted text called the cipher text that can only be read if decrypted.

The encryption scheme uses a pseudo-random encryption key generated by the algorithm. In the symmetric key scheme, the encryption and decryption keys are the same, thus communicating parties must have the same key before they can achieve secret communication.

The encrypted text which is the cipher text is decrypted in the server by only authorized users using the decryption key which is the same key as the encrypted key.

In the security architecture, each patient is assigned a unique ID called Patient ID (PID) which is generated using random number generation. The patient id generated is automatically encrypted. The patients' data are simulated with the Simulated Patient Data Software (SPDS) and sent to the Patient Data Acquisition Centre (PDAC) through wireless connection for consolidation and further transmission to the Central Hospital Monitoring System (CHMS) for processing and follow up by the medical care givers.

4. Results and Discussion

The Encryption/Decryption Algorithm

1.) Create the instance of javax.crypto.Cipher.
Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5Padding");

2.) Create the instance of sun.misc.BASE64Decoder.
BASE64Decoder d64 = new BASE64Decoder();

3.) Decode encryption key using decodeBuffer method of sun.misc.BASE64Decoder which will return a byte array.
byte[]b = d64.decodeBuffer(ENCRYPTION_KEY);

4.) Now create javax.crypto.spec.SecretKeySpec object using the key byte array and "AES".
SecretKeySpec key = new SecretKeySpec(strPassword.getBytes(), "AES");

5.) Now create javax.crypto.spec.IvyParameterSpec object using your IV spec.
AlgorithmParameterSpec paramSpec = new IvParameterSpec (strPassword.getBytes());

6.) Call the init method of Cipher Instance using encryption mode, key spec (created at step 4), IVParameter Spec(created at step 5).
cipher.init(Cipher.ENCRYPT_MODE, key, paramSpec);

7.) Call doFinal method of Cipher by passing the byte array of data which needs to be encrypted.
encrypted = cipher.doFinal(input.getBytes());

8.) Now encode the encrypted data using sun.misc.BASE64Encoder.
output = new BASE64Encoder().encode(encrypted)

Note About Key Size:

The AES algorithm requires that the key size must be 16bytes. So if you provide a key whose size is not equal to 16bytes, a java.security.InvalidKeyException will be thrown.

This algorithm uses cipher which is a sprat class in the javax.crypto package that is specially designed for encryption and decryption. It provides the functionality of a cryptographic cipher for encryption and decryption. To create a cipher object, the application calls the cipher's

getInstance method and passes the name of the requested transformation to it. The cipher getInstance method employs a random length padding which prevents an attacker from knowing the exact length of the plaintext message. This strengthens the algorithm because it prevents the cryptanalyst from using predictability to find known plaintext that can aid in breaking the encryption.

The base64Decoder converts the base64 encoded data into binary data and returns a byte array. The base64Decoder is responsible for displaying the encrypted files in the form of a ciphertext.

The algorithm uses a 16 bit key which is entered by the user and encrypted. A secret key spec and an algorithm parameter spec is created which validates the key input to make sure that it is the right key for the Advanced Encryption Standard algorithm to take place.

An init method which is found in the cipher class is called upon which ties the key to the encryption. The init method checks that the key supplied tallies with the key for the encryption.

The do final method is also a method inside the cipher class which is the final step of encryption. It gets the users input and converts it to byte which it stores in the byte array.

The Encode which is a method inside the base64Encoder displays the encrypted output.

This algorithm uses a 16-bit key and therefore the number of potential patterns to be created is quite large and this makes it difficult to break the code and descramble the contents.

This algorithm uses block ciphers which encrypts the data one fixed-size block at a time. The block cipher spreads a single bit of the plaintext across the entire encrypted block. Modes are employed in the encryption to combine the block cipher with the plaintext. The cipher block chaining (CBC) is the type of mode used in this encryption. CBC combines the previous block of cipher text with the next block of plaintext before encrypting it. A random value – an initialization vector is applied to the first block before it is encrypted. This chaining mechanism means the encryption of each block depends on the encryption of all previous blocks. CBC helps detect modifications to the cipher text.

The algorithm performs key distribution by using a public key to encrypt a randomly generated secret key. Then the recipient uses the corresponding private key to decrypt it.

Since the algorithm employs a random length padding (AES/CBC/PKCS5Padding), it prevents an attacker from knowing the exact length of the plaintext message. This strengthens the algorithm because it prevents the cryptanalyst from using predictability to find known plaintext that can aid in breaking an encryption.

Padding is used to prepare for the encryption using a specification such as PKCS5. This is achieved by appending random material to the message to make it long enough for the primitive. This padding scheme aims to ensure that the attacker cannot manipulate the plaintext to exploit the mathematical structure of the primitive that breaking the padding scheme is as hard as solving the hard problem underlying the primitive.

In the PKCS5 (Public Key Cryptographic Standards 5) padding used in this encryption, the last block is padded with integers, each of which is the number of bytes used to pad the message. The PKCS5 is used by padding with bytes all of the same value as the number of padding bytes. The input is padded with a padding string of between 1 and 8 bytes to make the total length an exact multiple of 8 bytes. The value of each byte of the padding string is set to the number of bytes added: i.e 8 bytes of value 0x08, 7 bytes of value 0x07, 6 bytes of value 0x06, 5 bytes of value 0x05, 4 bytes of value 0x04, 3 bytes of value 0x03, 2 bytes of value 0x02 or 1 byte of value 0x01

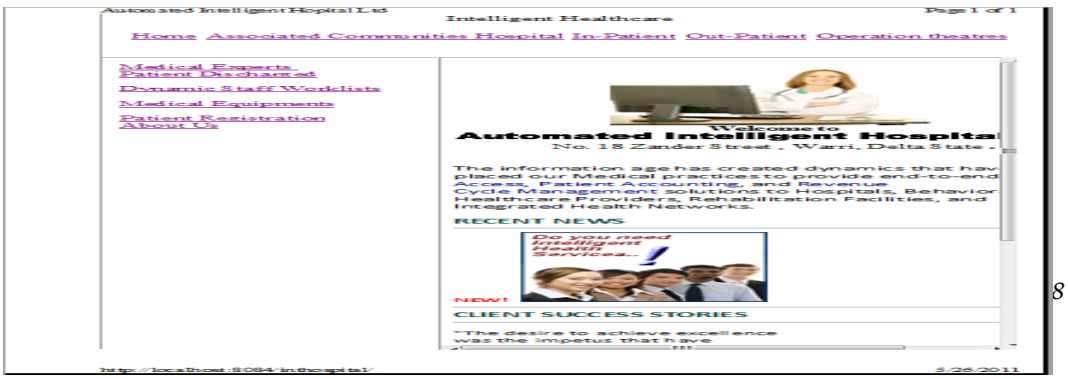

Figure 3: The Home Page

This is the first page that opens when the application is launched from where the user can navigate to other pages.

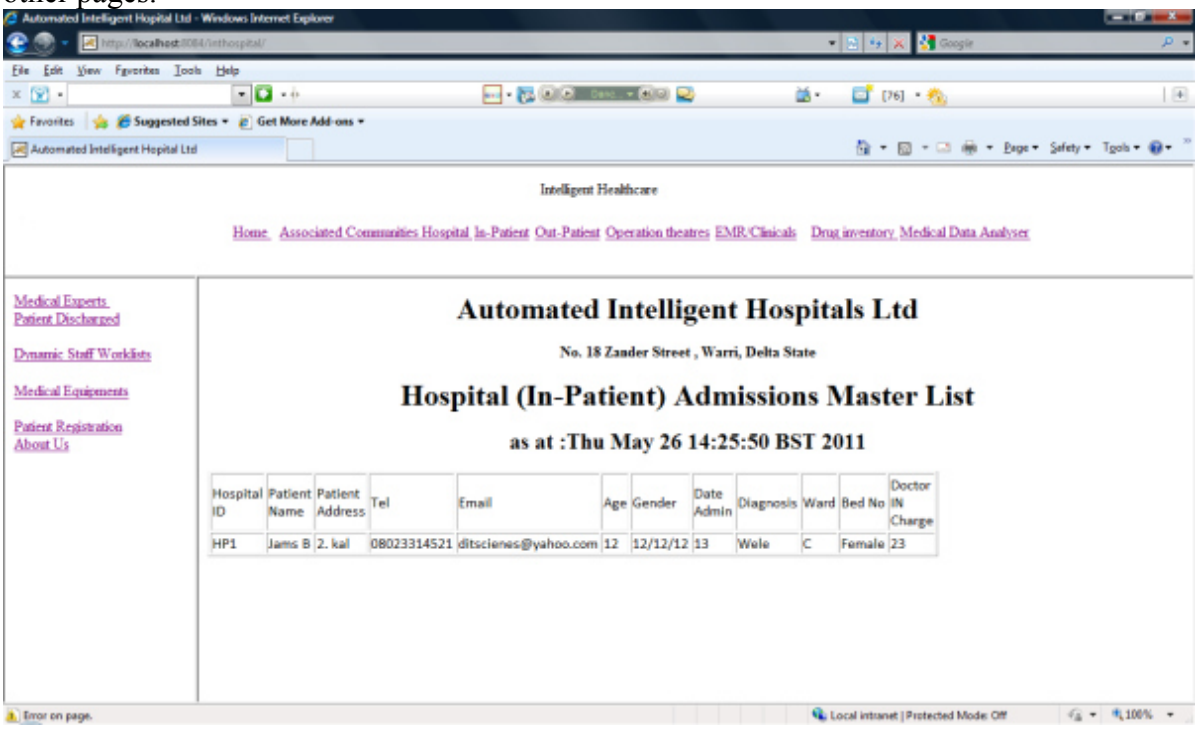

Figure 4: In – Patient Admission List

The in-patient admission list contains the detailed records of the patients on admission.

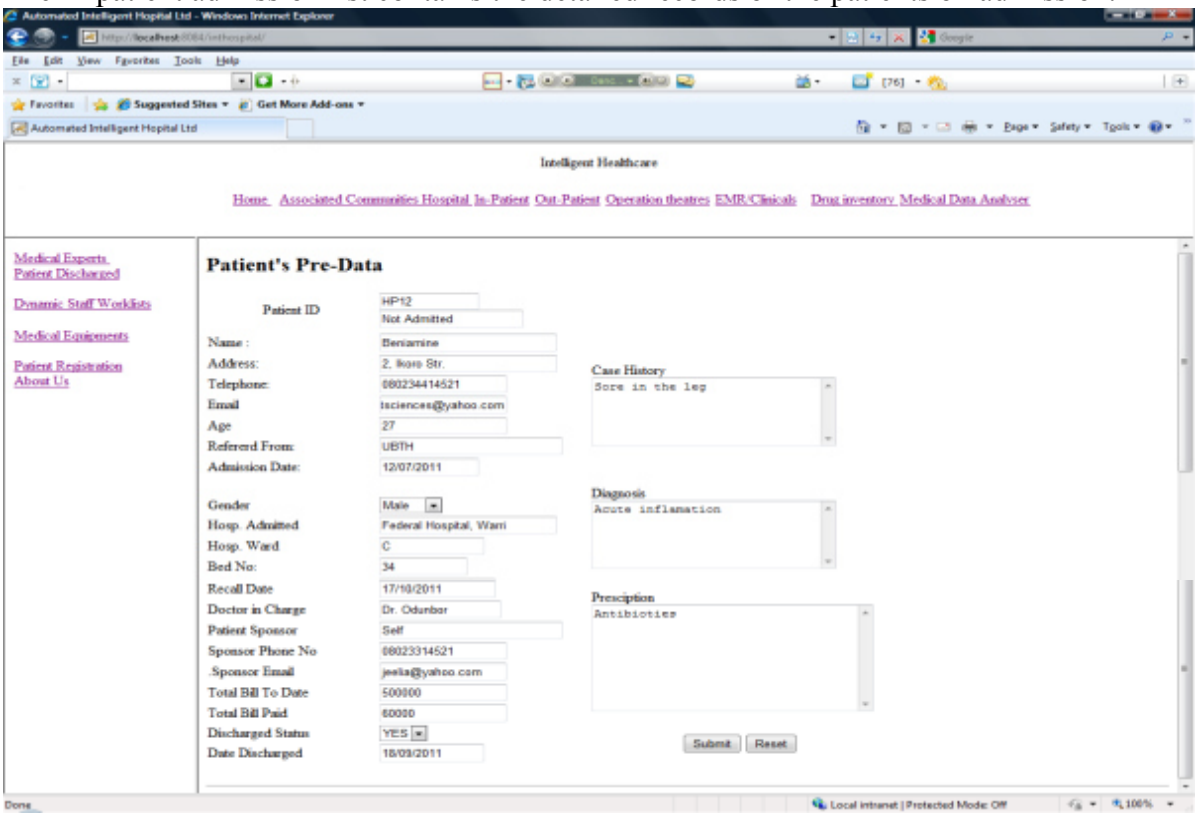

Figure 5: Patient Registration

A new patient is registered and all the details of the patient are entered here. If the patient is referred from another hospital, it is also entered.

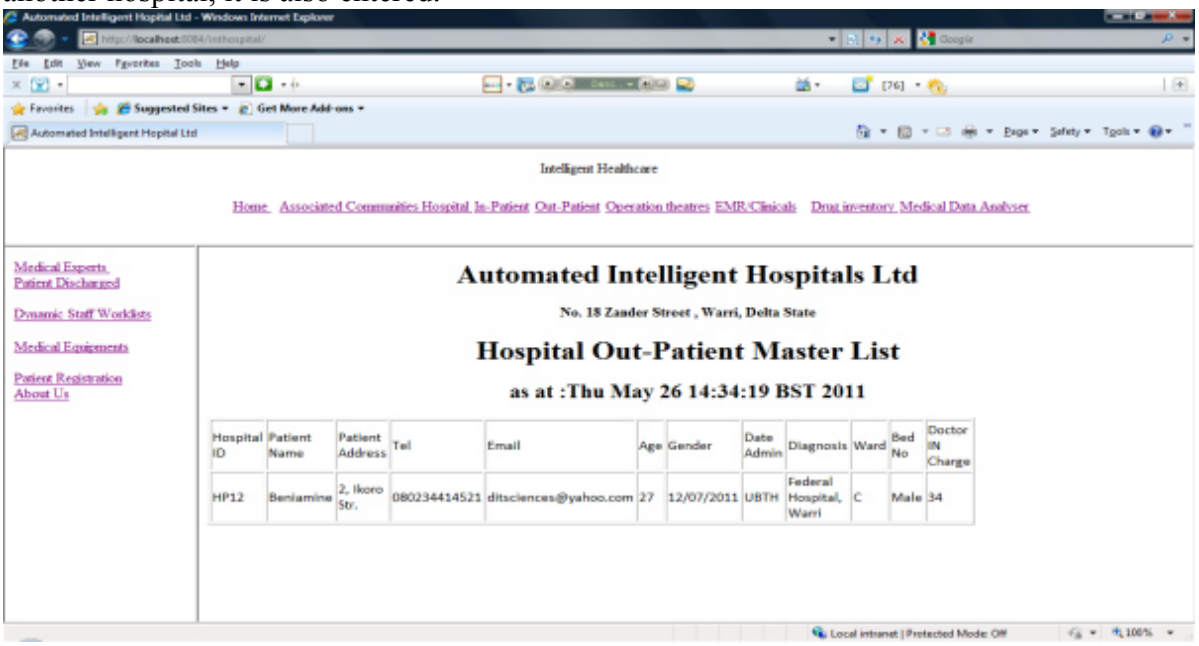

Figure 6: Out-Patient Master List

This is the master list of the out patients. All the records of the patients not on admission are entered here.

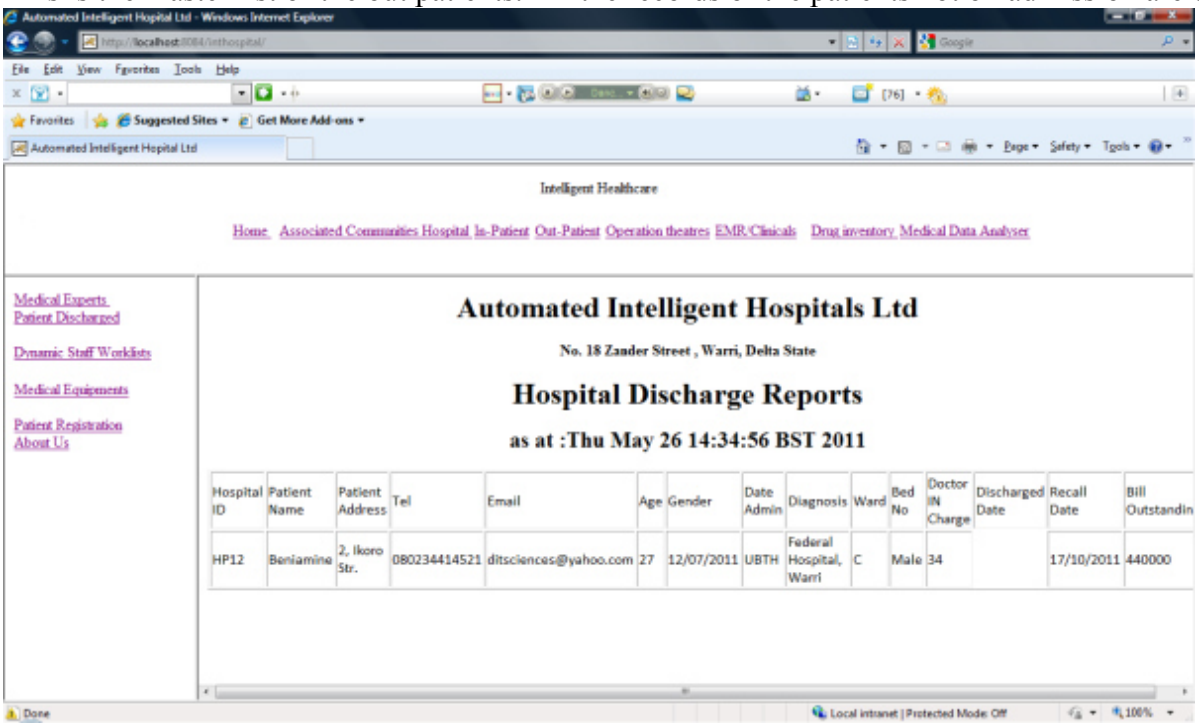

Figure 7: Discharge Reports

The details of all the patients discharged are stored in the Hospital Discharge Reports

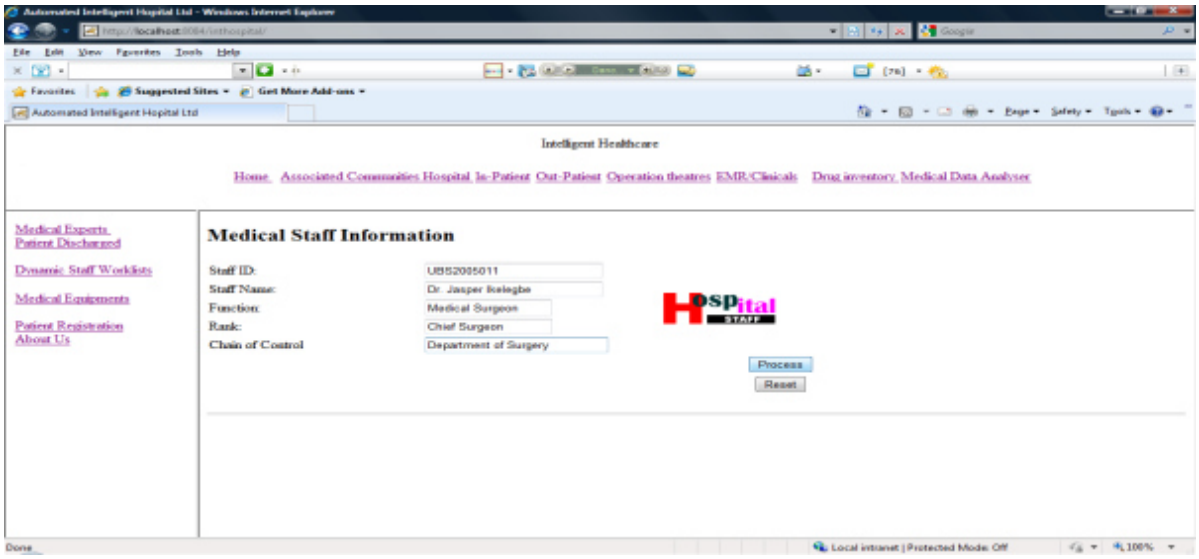

Figure 8: Medical Staff Information

The information of the medical staff are stored in the medical staff information.

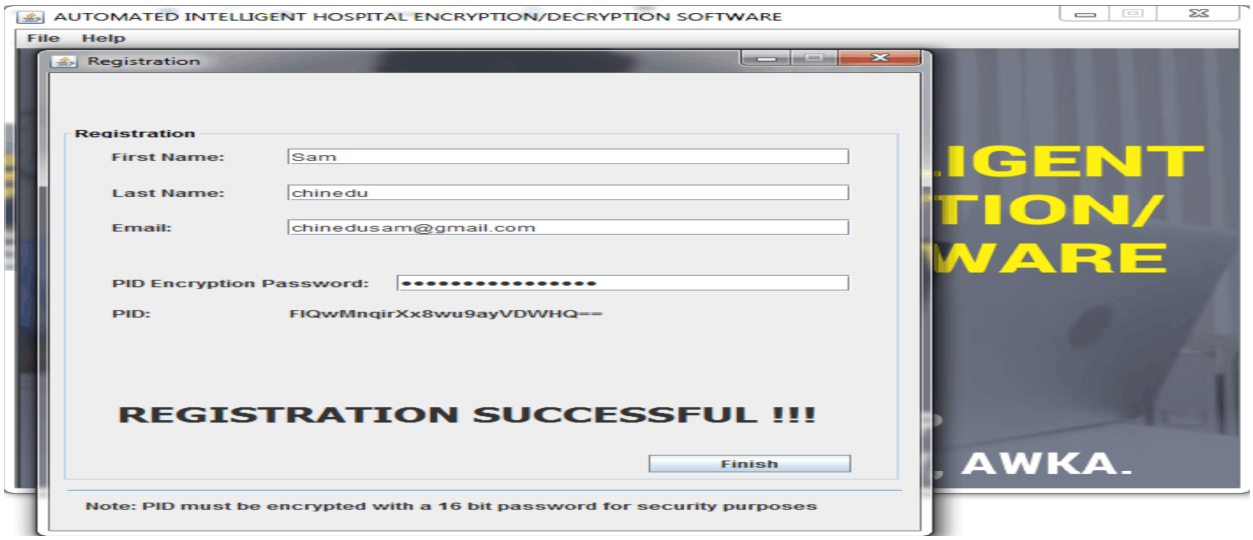

Figure 9: Patient Record Encryption

The patient's name are entered in the patient record encryption. A 16 bit password is entered in the pid encryption password. The patient id is generated and encrypted immediately. The patient's medical records are also hidden with the encryption.

Patient Record Decryption System is presented below in Figurue 10. In the pid decryption below, the administrator enters his email and his 16 bit password. This decrypts the patient id and displays it on the screen. The administrator then enters the pid and immediately the patient's medical records are displayed on the screen.

5.Conclusion

A secured automated Hospital Management System has been presented. It has been shown that at the manual hospital management system wastes a lot of time and effort, involves a lot of paper work and is prone to many errors. On the other hand, an automated Hospital Management System overcomes all the short comings of the manual Hospital Management System and is completely paperless. This is because all the patients' information are entered into the computer and all the computers are connected through a network and therefore the doctors can access patients' information from anywhere in the hospital.

The patients' medical records are secured by encrypting the records. This was achieved by using a 16 bit symmetric encryption algorithm. This encryption employs a 16 bit password to encrypt the patients' medical information. The encryption algorithm uses the same key for the encryption and the decryption. The patients' records are decrypted by entering the same key only by authorized personnel. In this way the patients' medical records are safeguarded from unauthorized access .

References

1. Ahmed M., Yahaya Y. and Matthias L. (2012), "*A Hospital Resource and Patient Management System Based on Real-Time Data Capture and Intelligent Decision Making"* International Conference on Systems and Informatics (ICSAI2012), pp 776 – 779.
2. Bain, C. (2014) "*Developing EffectiveHospital Management Information Systems: A Technology Ecosystem Perspective"*.
3. Diaping H., Weiguo X., Huizhang S. and Mengyu L. (2005) *"Study on Information System of Health Care Services Management in Hospital"* Proceedings of ICSSSM '05 International Conference on Services Systems and Services Management, 2005., vol. 2, 2005, pp. 1498–1501.
4. Digvijay H., Yadnyesh P. and Parineeta S. (2016). *"Hospital Management System"*. International Journal for Research in Engineering Application and Management (IJREAM) vol. 01, issue 11.
5. Hübner-Bloder, G., Ammenwerth E., Brigi B., Winter A. (2005). "*Specification of a Reference Model for the Domain Layer of a Hospital Information System."*Medical Informatics Europe, vol. 116, pp. 497–502.
6. Olusanya O., Elegbede A. and Ogunseye A. (2015) "*Design and Implementation of Hospital Management System Using Java*."
7. Wicks, A., Visich, J. and Li, S., *"Radio frequencyidentification applicationsin hospital environments"*. Hospital Topics, vol. 84, no. 3, pp. 3-9, 2006.