Chapter 1

INTRODUCTION

1. INTRODUCTION

In recent years, personal health record (PHR) has emerged as a patient-centric model of health information exchange. A PHR service allows a patient to create, manage, and control her personal health data in one place through the web, which has made the storage, retrieval, and sharing of the medical information more efficient. Especially, each patient is promised the full control of their medical records and can share her health data with a wide range of users, including healthcare providers, family members or friends. Personal health records (PHRs) should remain the lifelong property of patients, who should be able to show them conveniently and securely to selected caregivers and institutions. We present a cloud-based PHR system taking a radically new architectural solution to health record portability.

While it is exciting to have convenient PHR services for everyone, there are many security and privacy risks, although the various table text styles are provided. The main concern is about whether the patients could actually control the sharing of their sensitive personal health information (PHI), especially when they are stored on a third-party server which people may not fully trust. To assure the patients' control over access to their own PHRs, it is a promising method to encrypt the PHRs before outsourcing. Different from the single data owner scenario considered in most of the existing works in a PHR system, there are multiple owners who may encrypt according to their own ways, possibly using different sets of cryptographic keys. Letting each user obtain keys from every owner who's PHR they want to read would limit the accessibility since patients are not always online. An alternative is to employ a central authority (CA) to do the key management on behalf of all PHR owners, but this requires too much trust on a single authority. Here, we use novel patient-centric framework and a suite of mechanisms for data access control to PHRs stored in semi-trusted servers. To achieve fine-grained and scalable data access control for PHRs we have used various encryption methods. This shifts the computational complexity of the involved parties to the cloud without compromising clients' privacy and service providers' intellectual property. Finally, our security and performance analysis demonstrates the effectiveness of our proposed design.

We will also deploy portable sensors to collect various physiological data, such as body temperature, heart beat rate .Such physiological data could then be sent wirelessly to the system.

Then, using the online portal we can save it to the central server, which could then run various web medical applications on these data to return timely advice to the client.

1.1. Motivation

Sharing our common interest on cloud computing, we decided to make our final year project on cloud security domain. We researched about various implementations of cloud and came across its implementation in storage and maintenance of PHRs. We found out that its storage was not secure and the access was confined only to the patients themselves or was public. So, we thought of using a method to make PHR storage more secure by giving the patients an authority to decide the access of his/her medical history.

Summarizing the key motivation to take up this project:

- The patient need not undergo the same examination, unless strictly required for formulating a diagnosis if he has been through the same in recent past.
- A patient is never able to show his entire medical history to the doctor because the maintenance and carrying of physical reports is cumbersome.
- In recent past, storage of PHR in cloud has evolved but it also poses a serious risk on clients' privacy.
- We propose a system which not only provides storage in Cloud but also protects the privacy of the involved parties and their data.

1.2. Objective of the work

The objective of our project is:

- To provide lifelong medical history of patients
- To provide secure storage of personal health records
- To provide timely health advice to patients

1.3. Problem Statement

Personal health records (PHRs) should remain the lifelong property of patients, who should be able to show them conveniently and securely to selected caregivers and institutions. Currently, the storage of the PHRs are not very secure and properly maintained. Also online medical consultation from a group of doctors to choose from is not available. So, we design a system where a patient's complete medical history is saved at one place so that he can show it easily to the doctor as and when required.

Chapter 2

LITERATURE SURVEY

2. LITERATURE SURVEY

We have surveyed various papers related to secure sharing and storage of Personal Health Records in Cloud and reading current health data using Wireless Sensor Networks. The various papers are as follows: -

[1] MyPHRMachines: Lifelong Personal Health Records in the Cloud
[International Symposium on Digital Object Identifier: 10.1109/CBMS.2012.6266378; Van Gorp, P ; Comuzzi, M.]

Abstract: Personal Health Records (PHRs) should remain the lifelong property of patients and should be showable conveniently and securely to selected caregivers. Regarding interoperability, current solutions for PHRs focus on standard data exchange formats and transformations to move data across health information systems. In this paper we propose MyPHRMachines, a patient-centric system that takes a radically new architectural solution to health record interoperability. We propose to deploy besides the medical data also the related software to the PHR system. After uploading their medical data to MyPHRMachines, patients can access them again from remote virtual machines that contain the right software to visualize and analyze them without any conversion. Patients can share their remote virtual machine session with a selected health provider, who will need only a Web browser to access the pre-loaded fragments of the lifelong PHR. We illustrate how a prototype of MyPHRMachines already supports the use case of a real-world patient and discuss the research agenda required to translate our prototypical implementation into a viable solution for the international healthcare industry.

[2] CAM: Cloud-Assisted Privacy Preserving Mobile Health Monitoring
[IEEE TRANASCTIONS ON IMAGE PROCESSING VOL:8 NO:6 YEAR 2013 Huang Lin, Jun Shaoy, Chi Zhangz, Yuguang Fang, Fellow, IEEE]

Abstract: Cloud-assisted mobile health (mHealth) monitoring, which applies the prevailing mobile communications and cloud computing technologies to provide feedback decision support, has been considered as a revolutionary approach to improving the quality of healthcare service while lowering the healthcare cost. Unfortunately, it also poses a serious

risk on both clients' privacy and intellectual property of monitoring service providers, which could deter the wide adoption of mHealth technology. This paper is to address this important problem and design a cloud assisted privacy preserving mobile health monitoring system to protect the privacy of the involved parties and their data. Moreover, the outsourcing decryption technique and a newly proposed key private proxy re-encryption are adapted to shift the computational complexity of the involved parties to the cloud without compromising clients' privacy and service providers' intellectual property. Finally, our security and performance analysis demonstrates the effectiveness of our proposed design.

[3] Scalable and Secure Sharing of Personal Health Records in Cloud Computing using Attribute-based Encryption *[Issue 1;Date Jan. 2013 ;Ming Li Member, IEEE, Shucheng Yu, Member, IEEE, Yao Zheng,Student Member, IEEE, Kui Ren, Senior Member, IEEE, and Wenjing Lou,Senior Member, IEEE]*

Abstract: Personal health record (PHR) is an emerging patient-centric model of health information exchange, which is often outsourced to be stored at a third party, such as cloud providers. However, there have been wide privacy concerns as personal health information could be exposed to those third party servers and to unauthorized parties. To assure the patients' control over access to their own PHRs, it is a promising method to encrypt the PHRs before outsourcing. Yet, issues such as risks of privacy exposure, scalability in key management, flexible access and efficient user revocation, have remained the most important challenges toward achieving fine-grained, cryptographically enforced data access control. In this paper, we propose a novel patient-centric framework and a suite of mechanisms for data access control to PHRs stored in semi-trusted servers. To achieve fine-grained and scalable data access control for PHRs, we leverage attribute based encryption (ABE) techniques to encrypt each patient's PHR file. Different from previous works in secure data outsourcing, we focus on the multiple data owner scenario, and divide the users in the PHR system into multiple security domains that greatly reduces the key management complexity for owners and users. A high degree of patient privacy is guaranteed simultaneously by exploiting multi-authority ABE. Our scheme also enables dynamic modification of access policies or file attributes, supports efficient on-demand user/attribute revocation and break-glass access under

emergency scenarios. Extensive analytical and experimental results are presented which show the security, scalability and efficiency of our proposed scheme.

[4] Secure PHR Access Control Scheme for Healthcare Application Clouds *[2013 42nd International Conference on Parallel Processing Chia-Hui Liu, Fong-Qi Lin, Dai-Lun Chiang, Tzer-Long Chen, Chin-Sheng Chen, Han-Yu Lin, Yu-Fang Chung, Tzer-Shyong Chen]*

Abstract: With the development of information technology and medical technology, developed countries have been establish organization to set standard for electronic medical records in response to new generation and information on the application, they gradually develop emerging medical information exchange mode, Personal Health Records(PHR). PHR can integrate different kind of personal health records. With the Internet or portable device, PHR offers the integrity and accuracy personal health and medical records. Through electronic medical records, we can evaluate the quality of medical care, provide continued care to patients, promote the medical efficiency and increase the accuracy of medical diagnosis.

To implement PHR system in Cloud computing environment, we can decrease the cost of managing the infrastructure, and speed up dynamic resource adjustment based on the situation of the changes in client's demand and other advantages. However, it also brings new challenges. It is important to users to assure information security in Cloud environment, so this paper provides efficient and safe access management mechanism to solve PHR implement on Cloud environment□s security problem. The information security being threatened in the Cloud may lead to the collapse of medical care, loss of personal data or other serious consequence.

Finally, this paper provides new PHR access control mechanism built in Cloud environment with Bilinear Pairing. It provides users with a safe and efficient dynamically access PHR information mechanism. As a result from research, through access control scheme, it can withstand equation attack, external attack and reverse attack perfectly in Cloud computing environment.

Chapter 3

SYSTEM ANALYSIS

3. SYSTEM ANALYSIS

3.1 Existing System

3.1.1 Paper PHR

It is worth remembering that the most common media for PHRs remains paper. (Unfortunately, it also remains common for health care organization's authoritative health records.) Paper-based PHRs can provide a great deal of value simply and cheaply. Thanks to the ubiquitous photocopier, paper records are easy to reproduce. A list of allergies, medications and a summary of pertinent past medical history may be all that is needed in emergency situations, and provide the preponderance of information value in many non-emergency ones.

3.1.2 PC-based PHR

Paper's defects as a recording media are well understood. In particular, paper is hard to update. That problem can be eliminated by maintaining a "paper" PHR as a free-form text document stored on a personal computer (PC). Personal Health Applications installed on the PC can manipulate this data if it is taken from free-format text to structured storage. This may be the second most common form of PHR in place today.

3.1.3 Portable storage PHR

PHRs are sometimes defined as "your health data on a stick." The stick in this case is a "flash drive" (the name derives from the use of "flash memory" solid state memory chips). Flash drives are a perfect exemplar of today's electronic data storage – small, portable, and very high capacity. Smartcard-based health records are a variation on this theme (chips embedded in a card form factor instead of a stick), which may be purely storage devices or include algorithmic capabilities to manipulate the data.

3.1.4 Backups for PHR

Card and stick storage platforms can be easily lost or stolen (they are not just portable for the legitimate owner), and though resilient compared to a hard drive or floppy can also be damaged relatively easily. Encryption based on password or fingerprint can protect a portable storage

device, but passwords can be lost (and fingerprints don't always scan appropriately). (For more, see the Primer on Authentication of Identity.)

Thus, as with paper, the copy of data on a portable electronic PHR device should never be the only copy, at least if it is data that has any significant value or replacement cost. There must be backups. Who (or what) will keep them?

3.1.4 Internet storage PHR

Since one may see a health care provider anywhere, particularly in an emergency, the backups must be accessible anywhere the data subject (a.k.a., patient) might be. The Internet (specifically the World Wide Web) is the only practical communications platform for this role.

Tools exist for remote communication to an individual's PC over the Web. The complexities of such tools tend to put them beyond the reach of the casual computer user, and the vulnerability of PCs makes it a risky platform in any case. It is thus unsurprising that the number of that institutional players offering (or planning to offer) Web-accessible repositories for PHRs, as well as applications for them, is growing fast

3.2. Proposed System

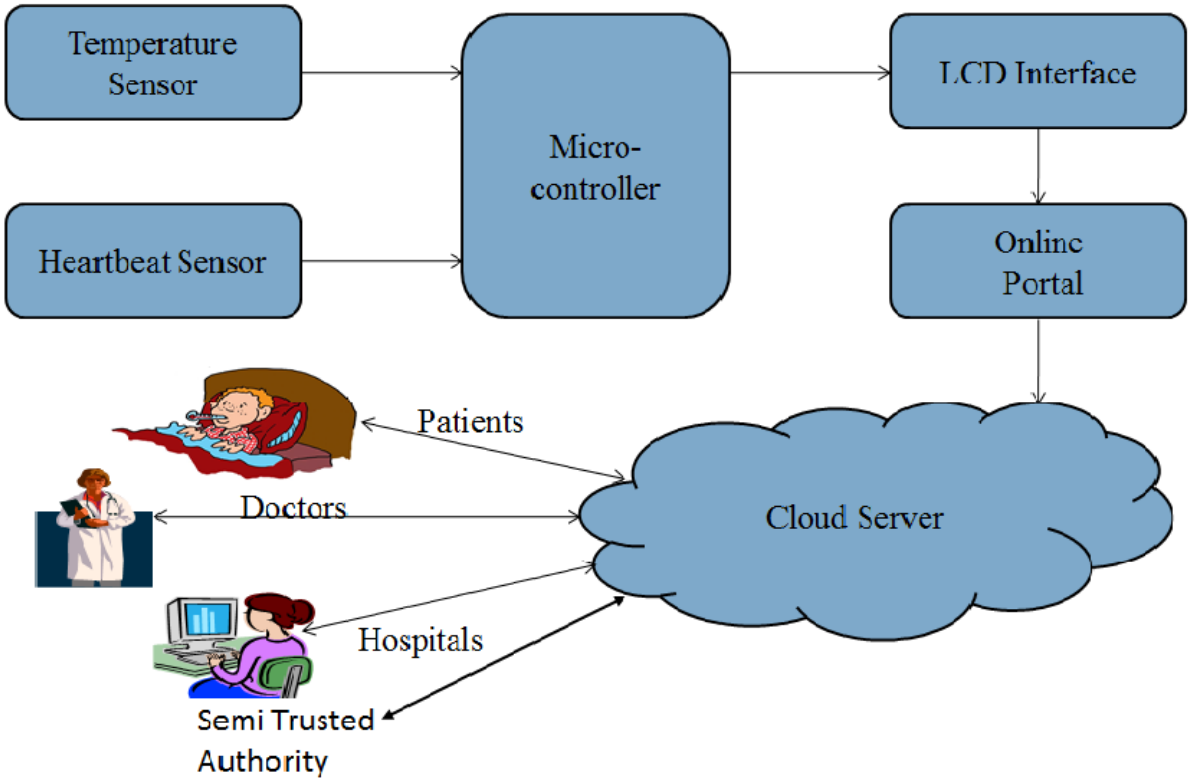


Figure :4.2. a Pictorial Representation of Proposed System.

Physiological data like the body temperature and the heartbeat rate is sensed using sensors which is then displayed on the LCD using an 8-bit microcontroller.

The collected data is then populated into the cloud server through an online portal. The data is transferred from the microcontroller using Zigbee technology. A log file is created in the system that can be viewed as and when required to upload the data. In cloud, we store the personal health records which is encrypted using the SHA1 technique where the token is generated by AES Algorithm. This ensures the security and integrity of the data stored.

The different stakeholders include the patients, doctors, hospitals, caregivers and the semi-trusted authority. The patients can be provided with the kit of physiological sensors at the time of registration in the hospitals involved. If the data entered by the patients are out of the normal range, the patients would be advised to act accordingly. The doctors can access the medical records of

the patient as required by the patients and also upload the report of the current diagnosis for future use. Also, the hospitals and medical institutions that generate reports based on various tests can upload them on cloud so that they can be accessed anywhere.

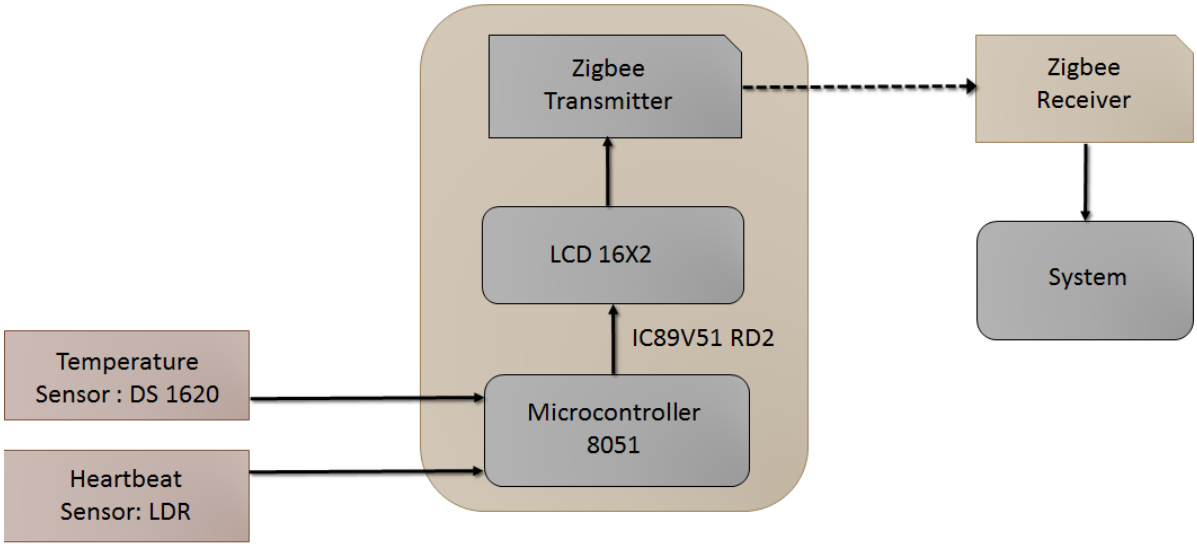


Figure: 4.2.b Pictorial Representation of Hardware system.

The hardware components comprise of a temperature sensor (ds1620) used for measuring body temperature, a heartbeat sensing device that uses a light dependent resistor (LDR) for measuring the patient's pulse. The resistance of a light dependent resistor is inversely proportional to the intensity of light received by the LDR. The greater the intensity of light received, the less will be the resistance.

A Light Emitting Diode which is part of the heartbeat sensing device, blinks each time a variation in resistance in the LDR is observed. The amount of blood flowing through our blood vessels varies whenever the heart pumps blood the amount of light absorbed by the blood vessels is more, and the remaining reflected light is detected by the LDR. During each pulse the LDR receives less light. Subsequently, the resistance of the LDR varies. This variation in resistance is observed and amplified using an op amp circuit so that it can be measured easily.

The heartbeat sensor as well as the temperature sensor work in coordination with the micro-controller 8051 which contains the IC 89v51 into which the programming code associated with the sensors is burnt. A LCD device is connected to this arrangement to display the temperature as well as the heartbeat as per the user's requirement. A Zigbee transmitter which is a part of the hardware arrangement is used to transmit the information appearing on the LCD to a Zigbee receiver. This zigbee receiver is connected to the user's system using UART (A universal asynchronous receiver/transmitter). The Universal Asynchronous Receiver/Transmitter (UART) takes bytes of data and transmits the individual bits in a sequential fashion.

Chapter 4

REQUIREMENT SPECIFICATION

4. REQUIREMENT SPECIFICATION

4.1 Hardware Requirements

Processor	-	Pentium –III
Speed	-	1.1 Ghz
RAM	-	256 MB (min)
Hard Disk	-	20 GB
Floppy Drive	-	1.44 MB
Key Board	-	Standard Windows Keyboard
Mouse	-	Two or Three Button Mouse
Monitor	-	SVGA
LCD	-	16X2
Temperature Sensor	-	DS1620
Heartbeat Sensor	-	LDR
Microcontroller	-	8051
Wireless Transmission	-	Zigbee Transmitter/Zigbee Receiver/Uhart

4.2 Software Requirements:

Operating System	:	Windows95/98/2000/XP
Application Server	:	Tomcat5.0/6.X
Front End	:	HTML, Java, JSP
Scripts	:	JavaScript.

Server side Script	:	Java Server Pages.
Database	:	MySql
Database Connectivity	:	JDBC.
Wireless Communication	:	Termite 2.9

Chapter 5

IMPLEMENTATION

5. IMPLEMENTATION

5.1 Algorithms

5.1.1 AES Algorithm

We have used an AES based encryption algorithm which uses key private proxy reencryption.We have also used Attribute Based encryption for referring to a particular doctor. The encryption algorithm is explained below.

1. KeyExpansion—round keys are derived from the cipher key using Irondale's key schedule. AES requires a separate 128-bit round key block for each round plus one more.

2. InitialRound

a. AddRoundKey—each byte of the state is combined with a block of the round key using bitwise xor.

3. Rounds

a. SubBytes—a non-linear substitution step where each byte is replaced with another according to a lookup table.
b. ShiftRows—a transposition step where the last three rows of the state are shifted cyclically a certain number of steps.
c. MixColumns—a mixing operation which operates on the columns of the state, combining the four bytes in each column.
d. AddRoundKey

4. Final Round (no MixColumns)

a. SubBytes
b. ShiftRows
c. AddRoundKey.

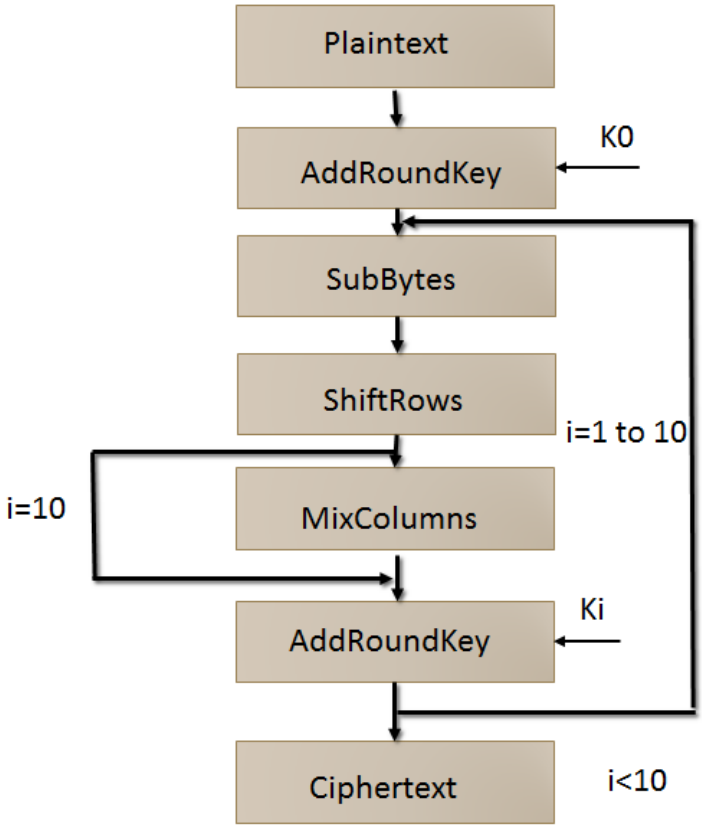


Figure: 5.1.1.a Pictorial Representation of AES Algorithm

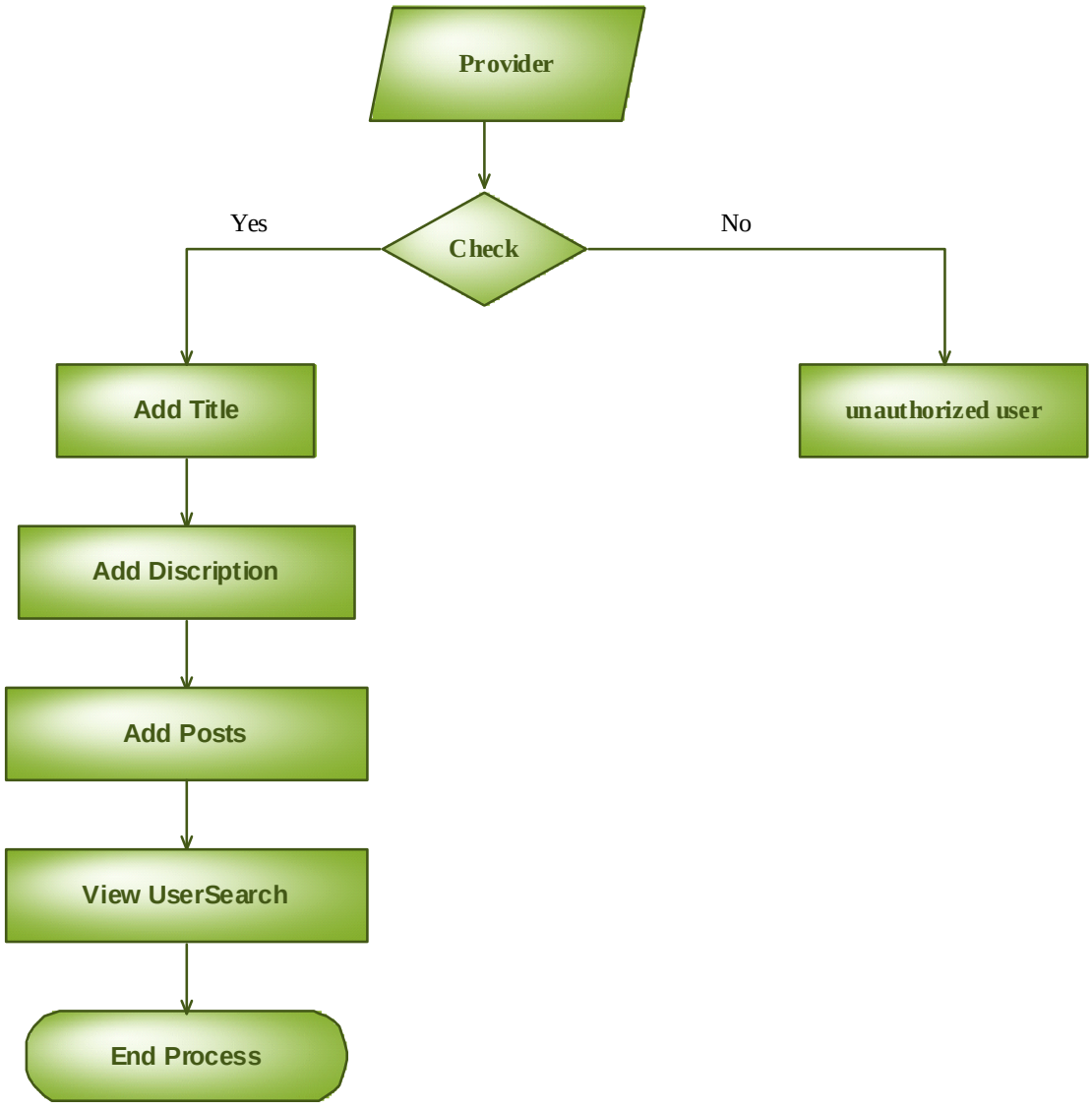
5.1.2 SHA1

In cryptography, SHA-1 is a cryptographic hash function designed by the United States National Security Agency and is a U.S. Federal Information Processing Standard published by the United States NIST.SHA-1 produces a 160-bit (20-byte) hash value. A SHA-1 hash value is typically rendered as a hexadecimal number, 40 digits long. SHA stands for "secure hash algorithm". The four SHA algorithms are structured differently and are named SHA-0, SHA-1, SHA-2, and SHA-3. SHA-0 is the original version of the 160-bit hash function published in 1993 under the name "SHA": it was not adopted by many applications. Published in 1995, SHA-1 is very similar to SHA-0, but alters the original SHA hash specification to correct alleged weaknesses. SHA-2, published in 2001, is significantly different from the SHA-1 hash function. SHA-1 is the most

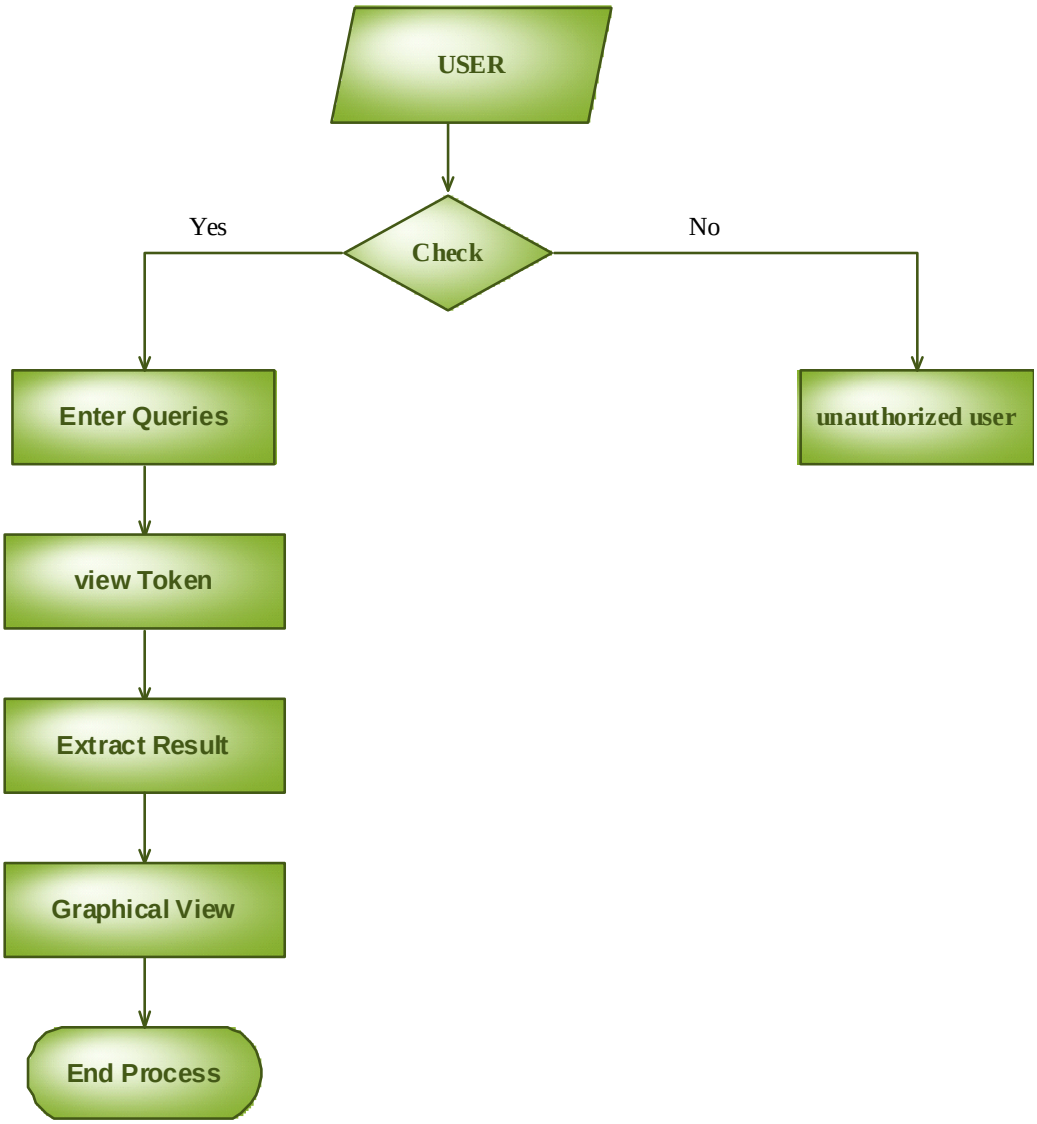
widely used of the existing SHA hash functions, and is employed in several widely used applications and protocols

5.2 System Design

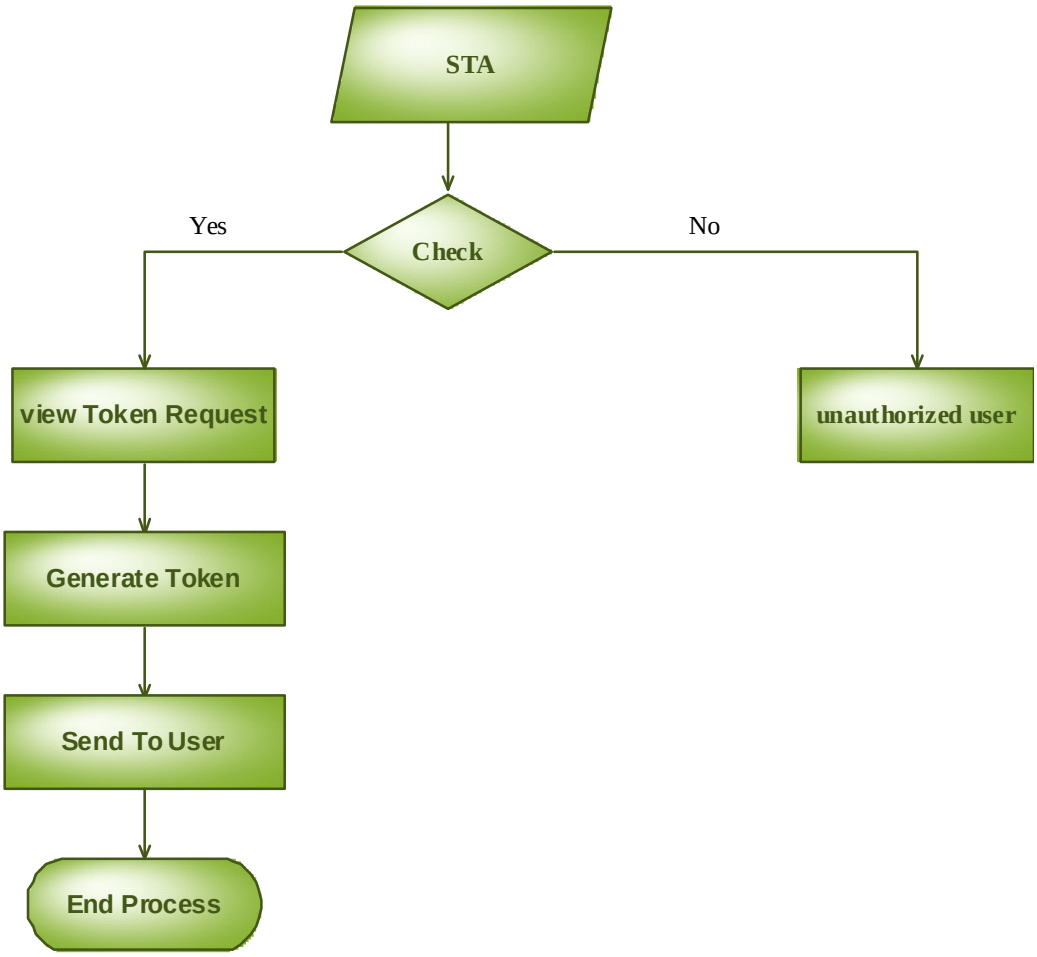
5.2.1 Provider:



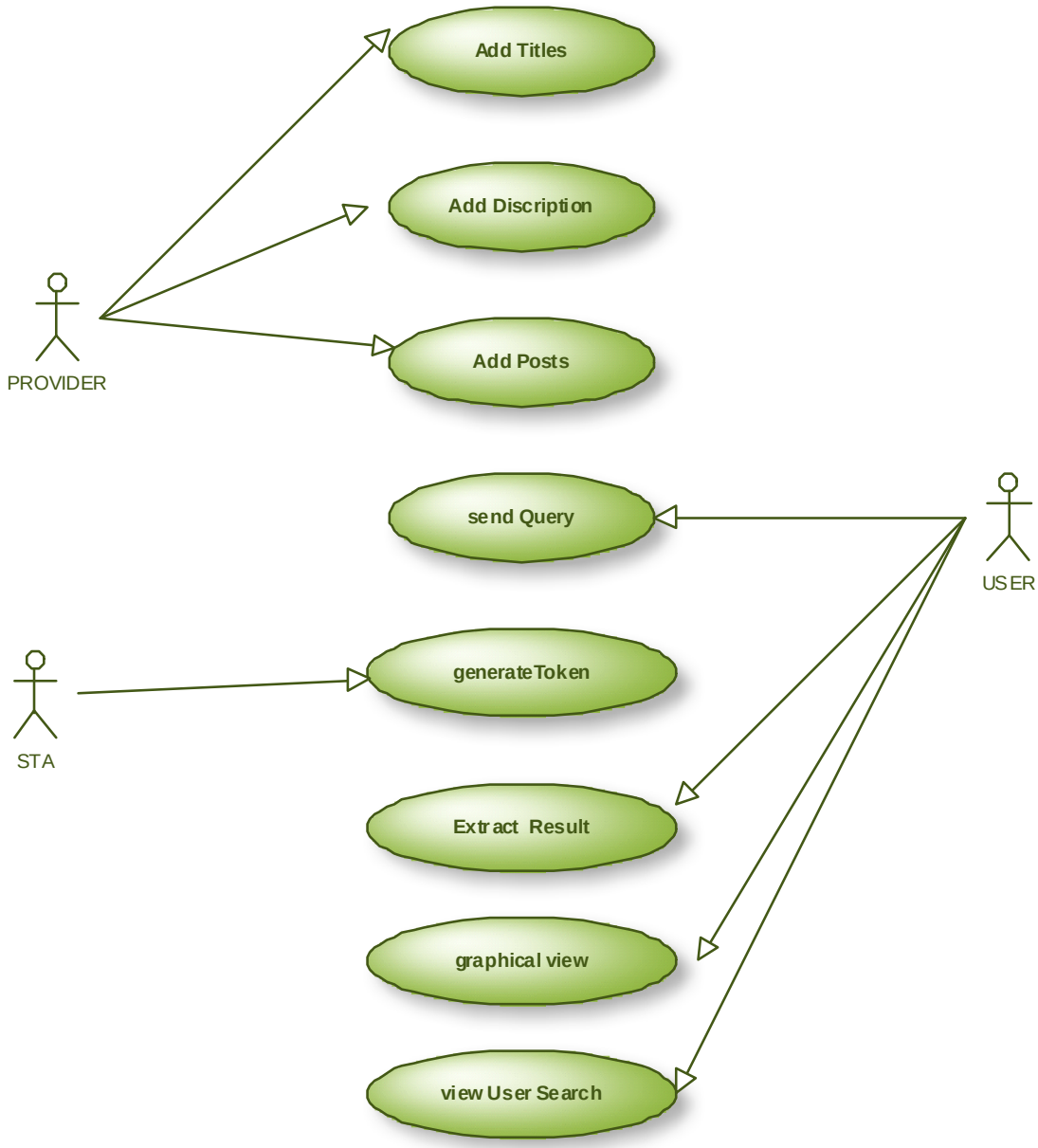
5.2.2 User:


USER
Check
Yes
No
Enter Queries
unauthorized user
view Token
Extract Result
Graphical View
End Process

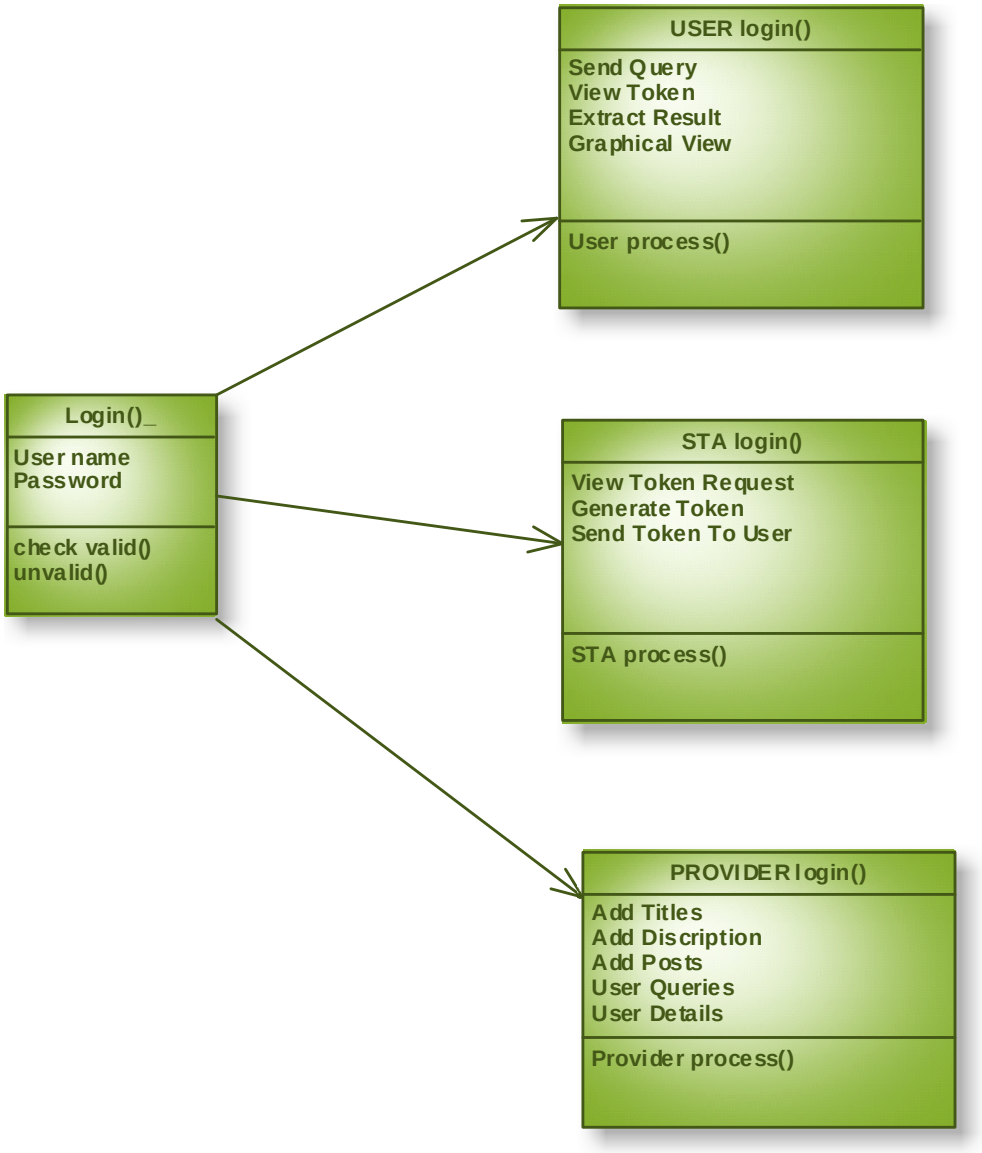
5.2.3 Semi-Trusted Authority(STA):



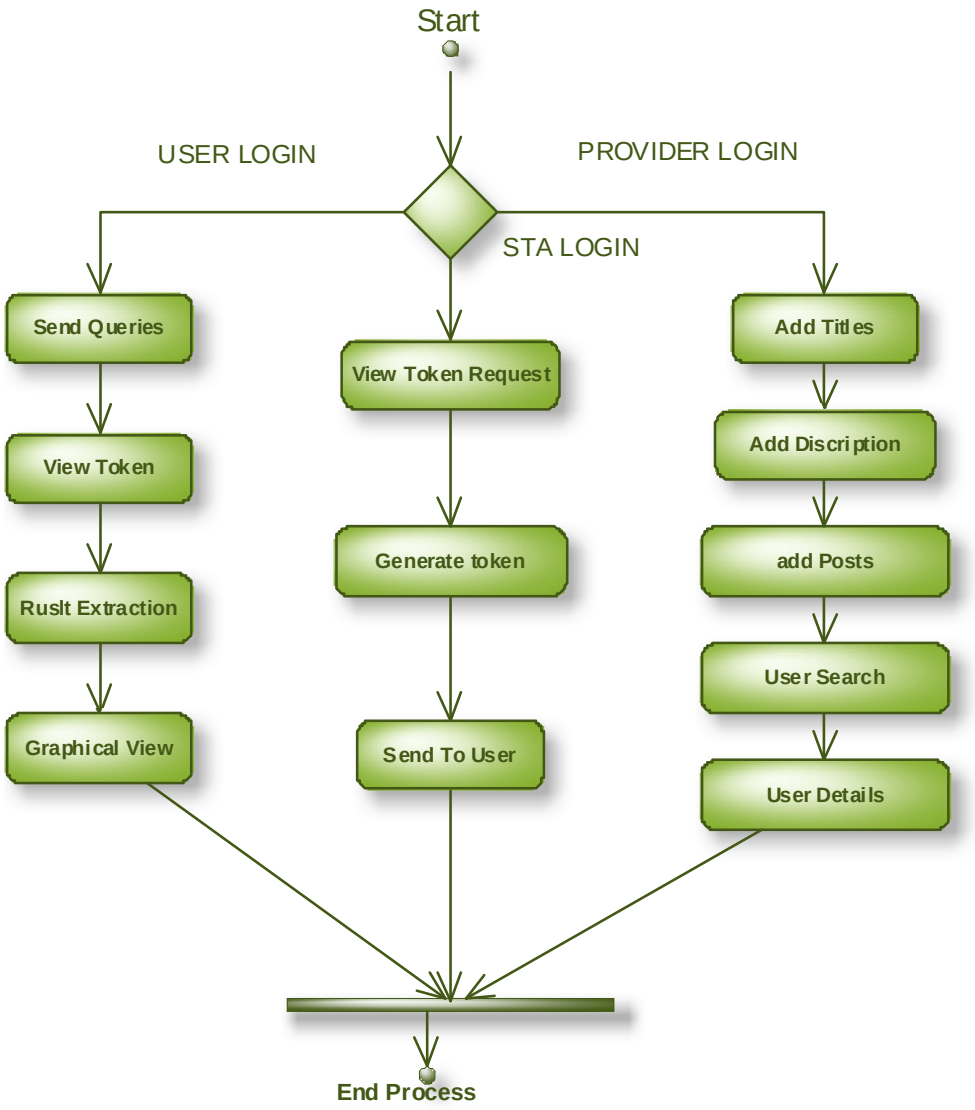
5.2.4 Use Case Diagram



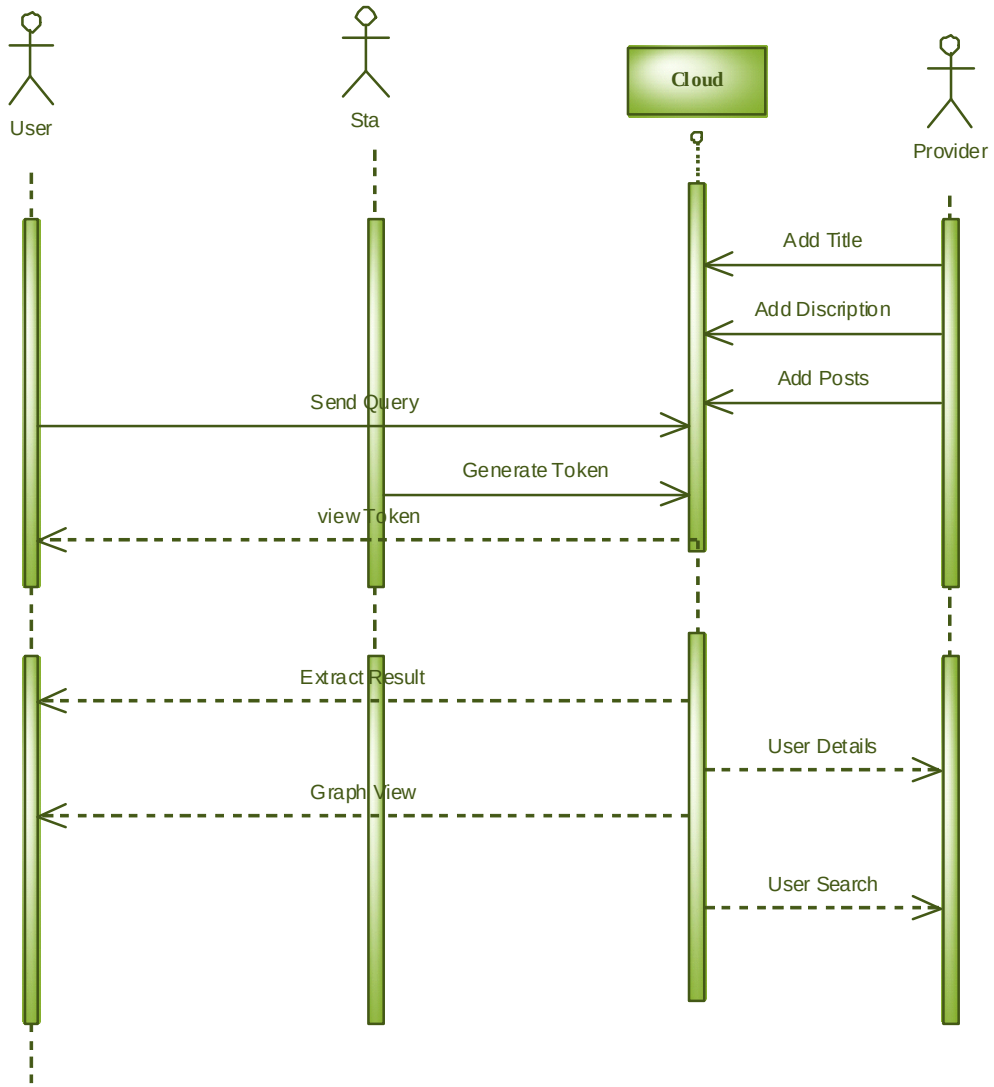
5.2.5 Class Diagram



5.2.6 Activity Diagram



5.2.7 Sequence Diagram



Chapter 6

VERIFICATION AND VALIDATION

6. VERFICATION AND VALIDATION

Verification and Validation (**V&V**) is the process of checking that a software system meets specifications and that it fulfills its intended purpose. It may also be referred to as software quality control. It is normally the responsibility of software testers as part of the software development lifecycle.

Validation checks that the product design satisfies or fits the intended use (high-level checking), i.e., the software meets the user requirements. This is done through dynamic testing and other forms of review.

Verification and validation are not the same thing, although they are often confused

- Validation: Are we building the right product?
- Verification: Are we building the product right?

6.1 Testing

Testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing can also provide an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs (errors or other defects).

6.1.1 White Box Testing

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

6.1.2 Black Box Testing

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is

treated, as a black box .you cannot "see" into it. The test provides inputs and responds to outputs without considering how the software works

6.1.3 Unit Testing:

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

6.1.4 Integration Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The ask of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

6.1.5 Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Chapter 7

SNAPSHOTS

7. SNAPSHOTS

Figure 7.a. Screenshot of the home page

Figure 7.b. Screenshot of the PHR home page

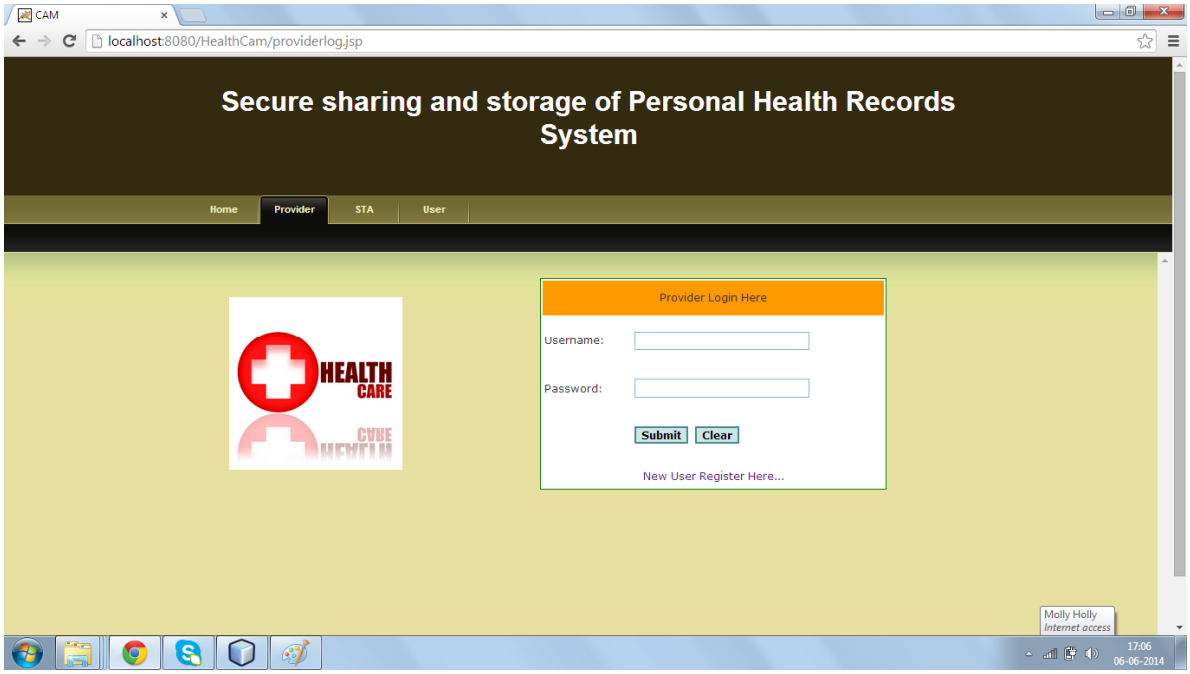


Figure 7.c. Screenshot of the Provider login

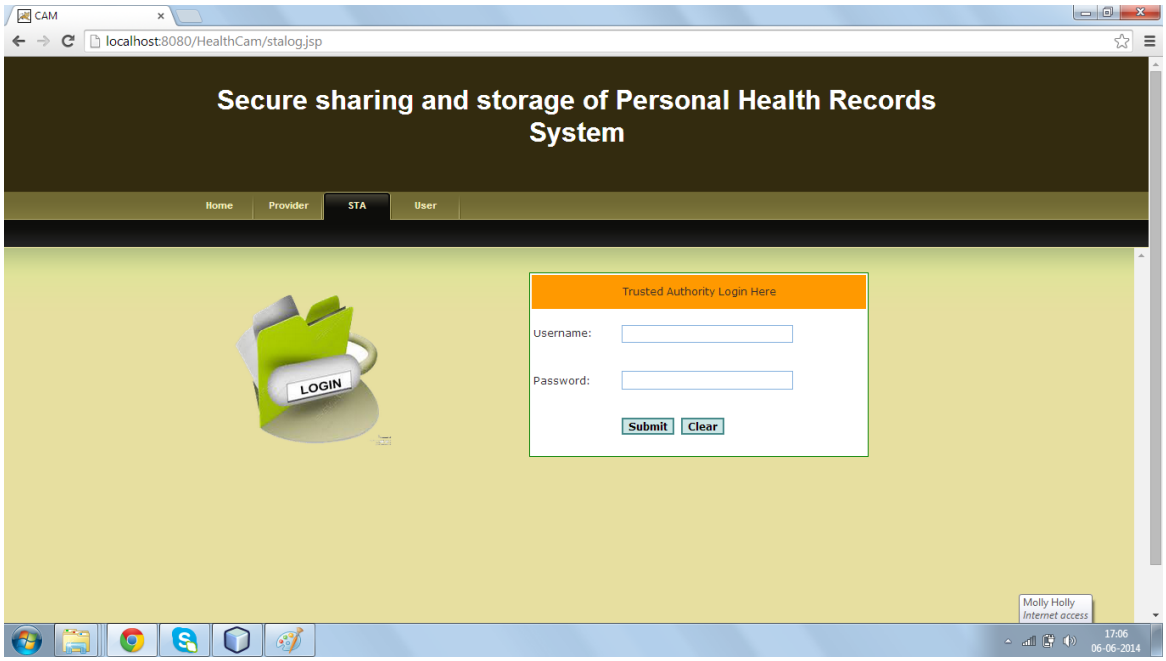


Figure 7.d. Screenshot of the STA login

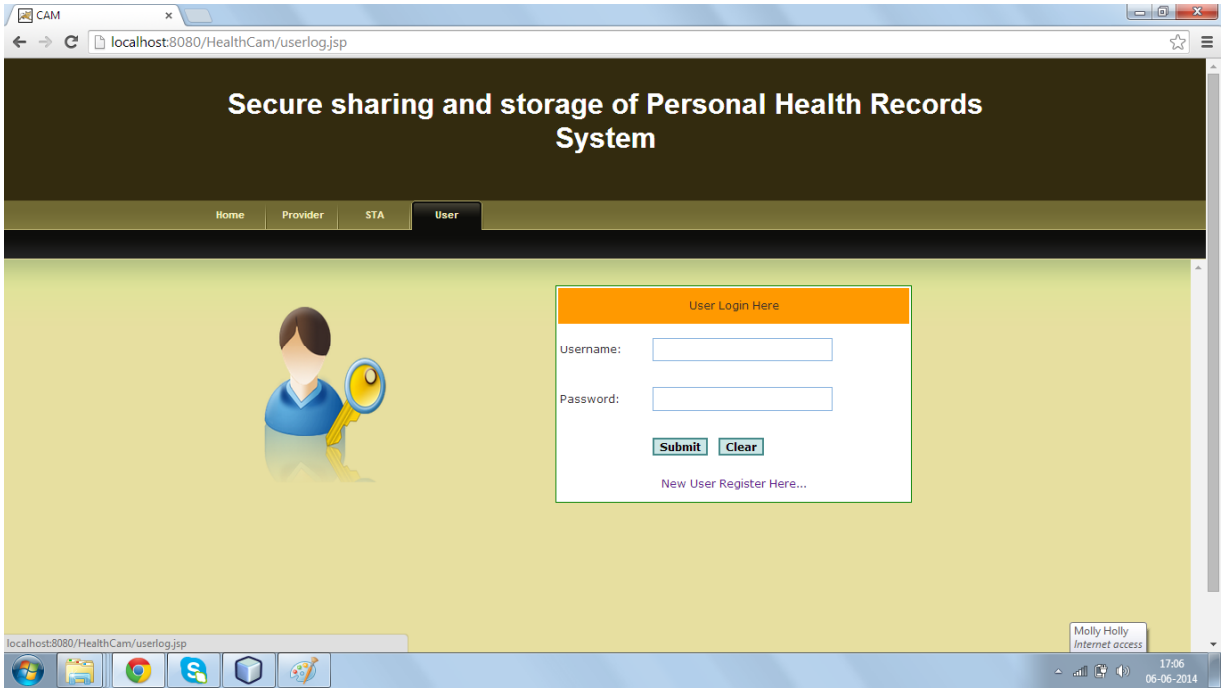


Figure 7.e. Screenshot of the User login

Figure 7.f. Screenshot of uploading the document

Figure 7.g. Screenshot of user details

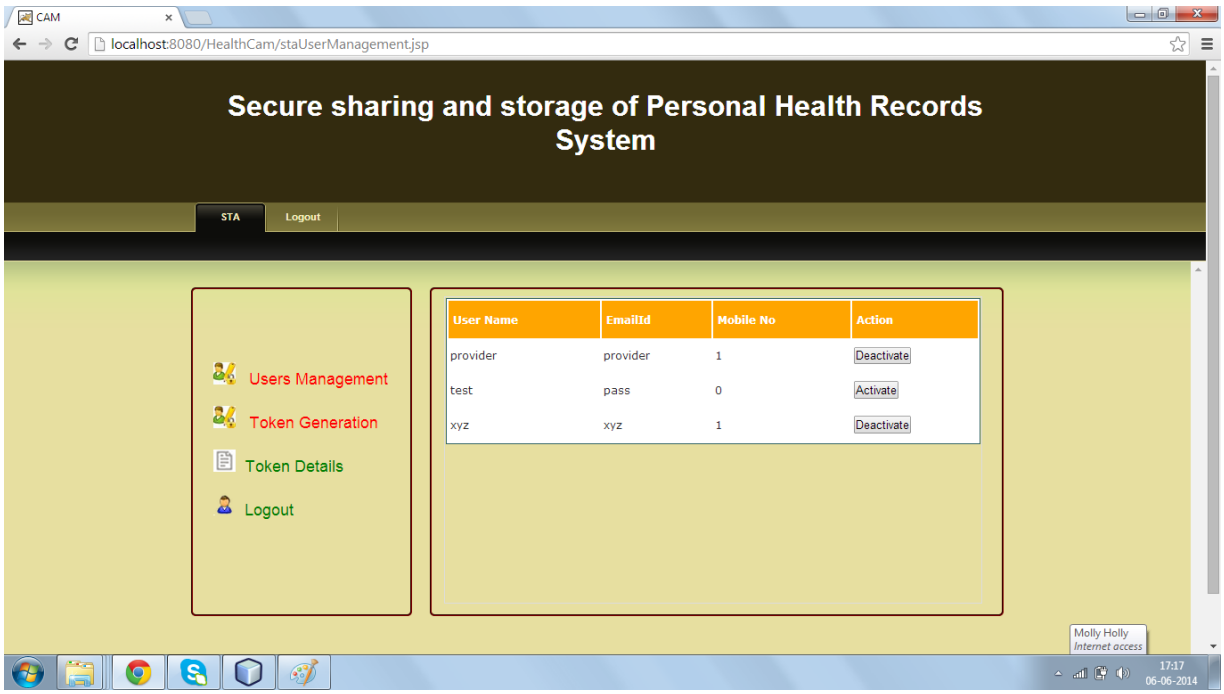


Figure 7.h. Screenshot of User management

Figure 7.i. Screenshot of STA generating the tokens

Figure 7.j. Screenshot of list of token generated

Figure 7.k. Screenshot of user home page

Figure 7.l. Screenshot of Personal health record of a user

Figure 7.m. Screenshot of token generated for different users

Figure 7.n. Screenshot of different queries requested by a user

Chapter 8

CONCLUSION AND FUTURE ENHANCEMENTS

8. CONCLUSION AND FUTURE ENHANCEMENTS

We have proposed a novel framework of secure sharing of personal health records in cloud computing. Considering partially trustworthy cloud servers, patients shall have complete control of their own privacy through encrypting their PHR files to allow fine-grained access. For PHR to develop well in the Cloud computing environment, we must make sure where the information come from and the integrity of content, also provide multi-user dynamic access mechanism.

In the short term, future work will concern the technical improvement of our prototype and focusing on various aspects:

- Portable sensors could be deployed in wireless body sensor networks to collect various physiological data, such as blood pressure (BP), breathing rate (BR), Electrocardiogram (ECG/EKG), peripheral oxygen saturation (SpO2) and blood Glucose to provide various functionalities ranging from sleep pattern analyzers, exercises, physical activity assistants, to cardiac analysis systems.
- Payment gateway can be added to make payments to doctor for online consultation.
- We can use latest technology to wirelessly transfer data from sensors to the system.
- We can deploy wireless sensor network to measure physiological data.,

REFERENCES

REFERENCES

1. MyPHRMachines: Lifelong Personal Health Records in the Cloud *International Symposium on Digital Object Identifier: 10.1109/CBMS.2012.6266378; Van Gorp, P ; Comuzzi, M.*
2. CAM: Cloud-Assisted Privacy Preserving Mobile Health Monitoring IEEE TRANASCTIONS ON IMAGE PROCESSING *VOL:8 NO:6 YEAR 2013 Huang Lin, Jun Shaoy, Chi Zhangz, Yuguang Fang, Fellow, IEEE*
3. Integrating Wireless Sensor Network into Cloud services for real-time data collection *ICT Convergence (ICTC), 2013 International Conference; Piyare, R. ; Dept. of Inf. Electron. Eng., Mokpo Nat. Univ., Mokpo, South Korea ; Sun Park ; Se Yeong Maeng;Sang Hyeok Park*
4. Scalable and Secure Sharing of Personal Health Records in Cloud Computing using Attribute-based Encryption *Issue 1;Date Jan. 2013 ;Ming Li Member, IEEE, Shucheng Yu, Member, IEEE, Yao Zheng,Student Member, IEEE, Kui Ren, Senior Member, IEEE, and Wenjing Lou,Senior Member, IEEE*
5. Secure PHR Access Control Scheme for Healthcare Application Clouds *2013 42nd International Conference on Parallel Processing Chia-Hui Liu, Fong-Qi Lin, Dai-Lun Chiang, Tzer-Long Chen, Chin-Sheng Chen, Han-Yu Lin, Yu-Fang Chung, Tzer-Shyong Chen*
6. Novel Cloud and SOA-Based Framework for E-Health Monitoring Using Wireless Biosensors *IEEE Journal of Biomedical and Health Informatics, Vol. 18, No. 1, January 2014; Abdelghani Benharref and Mohamed Adel Serhani*